Are Use Case Modeling Features Underutilized?
A Lightweight Survey that Raises Concerns

Mohamed El-Attar[1], Khaldoun Halawani[1], Mustafa Alsaleh[1] and Mahmood Niazi[1, 2, 3]
[1]*Department of Information and Computer Science,*
King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia
[2]*School of Computing and Mathematics, Keele University, ST5 5BG, Keele, U.K.*
[3]*Faculty of Computing, Riphah International University, Islamabad, Pakistan*

Keywords: Use Case Modeling, Notation Usage, Authoring Techniques, Reuse Mechanisms.

Abstract: Use case modeling is a very popular technique for eliciting, specifying and validating functional requirements. Use case modeling possesses a very rich notational set that allows its users to accurately specify a large variety of aspects about the underlying system's requirements. Many authoring techniques and templates were introduced to accurately describe a system's functional requirements. Although a relatively simple modeling technique, the literature has repeatedly reported on its misuse, leading to the development of end systems that do not satisfy the intended requirements. To this end, we have conducted a survey of use case models available online to shed light on the level of utilization of the use case modeling notation and how they are described, which can be symptomatic of how well do requirements engineers utilize the use case modeling technique and its modeling capabilities. In our survey we have collected and analysed 105 use case models. The results show an underutilization of the use case modeling notation and improper authoring techniques, which raises concern over the quality of the end systems.

1 INTRODUCTION

Modelers create use case models to accurately describe the functional requirements of a system (Booch, 2005; OMG, 2009). If modelers voluntarily restrict themselves from using the various use case modeling features then this will result in the development of use case models that do not accurately represent the underlying functional requirements. Mal-practice of use case modeling is particularly worrying in use case-driven development methodologies, where the quality of use case models has a significant impact on other development activities downstream.

The literature repeatedly reports on cases of use case modeling mal-practice (Anda and Sjøberg, 2002; Anda et al., 2001; Berenbach 2004; Bittner and Spence, 2004; Cockburn 2000; Lilly, 1999; Overgraad and Palmkvist 2005). Consequently, a great deal of research has been devoted to guide and improve use case modeling efforts (Anda and Sjøberg, 2002; Berenbach 2004; Bittner and Spence, 2004; 2001; Cockburn 2000). However, has the application of the use case modeling technique improved? To date, the overwhelming majority of use case models produced and being produced are of poor quality, which is evidenced by the significant subset of software development projects that fail due to requirements related issues, including low quality use case models. The *position* argued by the authors of this paper is that the continuing trend of poor quality use case models produced is due to a significant underutilization of use case modeling features. To support this *position*, a lightweight survey of 105 publicly available use case models was conducted. The collected use case modelled were analysed to determine their utilization of the complete set of use case modeling features.

2 USE CASE MODELING FEATURES

System boundary is an important notation since it explicitly indicates which entities are part of the end system and which are external to the system under development. Failure to include system boundary may lead to confusion as to which entities will need

El-Attar M., Halawani K., Alsaleh M. and Niazi M..
Are Use Case Modeling Features Underutilized? - A Lightweight Survey that Raises Concerns.
DOI: 10.5220/0004095102030206
In *Proceedings of the 7th International Conference on Evaluation of Novel Approaches to Software Engineering* (ENASE-2012), pages 203-206
ISBN: 978-989-8565-13-6

to be implemented (Lilly, 1999). As such, developers may implement functionality that is not required, which is a waste of resources.

Abstraction in use case models serves as a reuse mechanism in a similar was as in object-oriented programs (Bittner and Spence, 2004). For use cases, abstraction is used to state common yet incomplete behaviour. This incomplete behaviour should be completed by a *specializing* use case. For actors, abstraction can be used to specify a generalized actor role that is common amongst other actors, but one that itself is also incomplete and thus needs to be realized by a *specializing* actor. The extend, include, use case generalization and actor generalization relationships are reuse mechanisms in use case modeling (Bittner and Spence, 2002; Overgraad and Palmkvist, 2005).

The textual description of use cases is the heart of any use case model. A use case model that only includes a diagram would be quite vague about the details of the underlying functional requirements. Failure to include use case descriptions will require its readers to guess the details of the underlying functional requirements, which is likely to be incorrect or inaccurate (Anda and Sjøberg, 2002; Cockburn 2000). The use of templates to describe use cases greatly improves quality in use case models.

3 DATA COLLECTION, RESULTS AND DISCUSSION

The search excluded use case models created solely for educational purposes. Therefore, use case models that were used in tutorials and example use case diagrams in books, journals and conference proceedings were ignored. Use case models that were created using the old notational set of use case modeling were also ignored, for example use case diagrams that used the now outdated <<uses>> stereotype. The search also excluded use case models created by University students as part of a training exercise on use case modeling itself. The student use case models that were considered were those that were built as part of a development project whereby the use case models were created with the intention to develop an end product. Such use case models are usually developed as part of a student's senior graduation project. Upon executing this search, 105 use case models were elicited.

The data collected is categorized according to its source from the use case model, i.e. if it is a diagrammatic element or is it an element from the textual descriptions. Detailed categorization of the information collected is shown in Tables 1-3.

Links to access each use case models is available in an Excel sheet which can be located at (El-Attar, 2012) The data collected is shown in Tables 1-3. Table 1 shows the data collected for all 105 use case models. Table 2 shows the data collected only for the 65 industrial use case models. Finally, Table 3 shows the data collected only for the 40 student-developed use case models.

Table 1: All use case models.

All Use Case Models	Percentage Used
Diagrammatic Elements	
System boundary	45.76%
Actors	100.00%
Abstract actors	14.29%
Use cases	100.00%
Abstract use cases	6.67%
Extension points	2.86%
Relationships	
Extend	27.62%
Extend with a condition	1.90%
Include	38.10%
Use case generalization	12.38%
Actor generalization	13.33%
Reuse	54.29%
Association	96.19%
Bi-directional	65.71%
Directed	32.38%
Combination	1.90%
Cardinality	2.86%
Textual Descriptions	
Present	48.57%
General characteristics	
Bullet points	45.18%
Free-flow form	58.82%
Template	27.45%
Components	
Basic flow	100.00%
Alternative flow	47.06%
Preconditions	41.18%
Postconditions	33.33%
Special requirements	13.73%

Table 2: Real-world use case models.

Real-World Use Case Models	Percentage Used
Diagrammatic Elements	
System boundary	44.62%
Actors	100.00%
Abstract actors	13.85%
Use cases	100.00%
Abstract use cases	9.23%
Extension points	1.54%
Relationships	
Extend	26.15%
Extend with a condition	0.00%

Table 2: Real-world use case models (Cont.)

Include	44.62%
Use case generalization	13.85%
Actor generalization	12.31%
Reuse	60.00%
Association	98.46%
Bi-directional	65.71%
Directed	32.38%
Combination	1.54%
Cardinality	2.86%
Textual Descriptions	
Present	50.77%
General characteristics	
Bullet points	24.24%
Free-flow form	75.76%
Template	27.27%
Components	
Basic flow	100.00%
Alternative flow	51.52%
Preconditions	42.42%
Postconditions	30.30%
Special requirements	15.15%

Table 3: Student use case models.

Student Use Case Models	Percentage Used
Diagrammatic Elements	
System boundary	45.00%
Actors	100.00%
Abstract actors	15.00%
Use cases	100.00%
Abstract use cases	2.58%
Extension points	5.00%
Relationships	
Extend	30.00%
Extend with a condition	5.00%
Include	27.50%
Use case generalization	10.00%
Actor generalization	15.00%
Reuse	45.00%
Association	92.50%
Bi-directional	62.50%
Directed	32.50%
Combination	5.00%
Cardinality	2.50%
Textual Descriptions	
Present	45.00%
General characteristics	
Bullet points	44.44%
Free-flow form	55.56%
Template	27.78%
Components	
Basic flow	100.00%
Alternative flow	38.89%
Preconditions	38.89%
Postconditions	38.89%
Special requirements	11.11%

Upon analysing the data collected and shown in Tables 1-3, a number of concerns are raised about how use case modeling is practiced. Further concern is raised as some trends of use case modeling practice in industrial settings is found to match trends of use case modeling practice in academic settings. The following is a list of concerns raised after analysing the data collected:

- The system boundary was absent in more than half of the use case models. Further analysis shows that students are just as likely as practitioners to overlook the depiction of the system boundary.
- The level of use of *abstract* actors is low at approximately 14%. This percentage was found to be approximately the same between students and practitioners.
- The level of use of *abstract* use cases were found also to be low. However, practitioners were much more likely to introduce *abstract* use cases than students.
- The level of use case of *extension points* in use cases was found to be very low at approximately 3%, with students being much more likely to use it than practitioners.
- Only 5% of student use case diagrams used the *extend* relationship while specifying a condition. Meanwhile, no use case diagrams developed by practitioners were found to this notational feature.
- While the use of the *include* relationship was high, it was found the practitioners were twice more likely to use this relationship than students.
- The use of actor and use case *generalization* relationships were found to be equally low in use case diagrams developed by students and practitioners.
- Almost half of the use case models did not promote reuse using the various use case modeling relationships, with practitioners slightly more likely to use these relationships than students.
- The actor *generalization* relationship was found in few diagrams than *abstract* actors. This means that in some use case diagrams there were *abstract* actors that were not *specialized*.
- There are some use case diagrams created by practitioners and students that were found not to contain a single *association* relationship.
- Practitioners and students were twice as likely to use the bi-directional *association* relationship as they are to use the directed *association*

relationship. However, very few diagrams by practitioners and students were found to contain both types of *association* relationships, which might be indicative that the modelers may not know the difference between the two types of *association* relationships. Therefore, decisions of whether or not to add an arrow head to an *association* relationship notation were perhaps arbitrary.

- Specifying cardinality at an *association* relationship end was a very rare practice which occurred in less than 3% of all diagrams.
- Although Tables 1-3 show that almost half of the use case models did not include textual descriptions. This statistic is likely to be misleading since it may be possible that the textual descriptions were available at a different source which we were unable to access. However, there is also likelihood that a subset of the use case models actually does not include textual descriptions of the use cases.
- It was found that students are much more likely to describe their use cases in bullet-point form while practitioners are much more likely to describe their use cases in free-flow text form.
- The utilization of a template to describe use cases was equally low in practitioner and student use case models.
- More than half the use case descriptions did not specify alternative flows, preconditions and postconditions.
- Very few textual descriptions included the relative constraints imposed in non-functional requirements, i.e. very few included a 'Special Requirements' section.

4 CONCLUSIONS AND FUTURE WORK

Use case modeling is constantly increasing in popularity. The results of this survey show that the majority of use case modeling features is underutilized or misused. Given the current trends in use case modeling practice, there is great concern that software development teams will continually develop low quality systems. We argue that more care should be taken while teaching use case modeling in academic and industrial settings. Care should be given in the form of exposing students to the various use case modeling features and explaining how they should properly use them. In industry, current certifications by well-established organizations should conduct a more thorough examination of the use case modeling skills of analysts.

This study is considered preliminary since it assesses quality in use case models based on its utilization of use case modeling features. A more thorough study would certainly be desirable which will carefully analyse modeling decisions made in each use case models and while referring with the authors of each use case model. This comprehensive study is planned for future work.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support provided by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum & Minerals (KFUPM) for funding this work through project No. IN111028.

REFERENCES

Anda, B., and Sjøberg, D. I. K., " Towards an Inspection Technique for Use Case Models," in *Proc. 14th Int'l Conf. on Software Eng. and Knowledge Eng*., 2002, pp. 127-134.

Anda, B., Sjøberg, D. and Jørgensen, M. "Quality and Understandability in Use Case Models," in *Proc. 15th European Conf. Object-Oriented Programming,* J. Lindskov Knudsen, ed., 2001, pp. 402-428.

Berenbach, B., "The Evaluation of Large, Complex UML Analysis and Design Models," in Proc. 26th Int'l Conf. on Software Eng., pp. 2004, pp. 232-241.

Bittner, K. and Spence, I., Use Case Modeling. Addison-Wesley, 2002.

Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Second Edition. Addison-Wesley, 2005.

Cockburn, A., Writing Effective Use Cases. Addison-Wesley, 2000.

El-Attar, M., "Data Files Containing References to Use Case Models", [online] Available at: http://faculty.kfupm.edu.sa/ICS/melattar/UseCaseStats.html, [Accessed 6 April 2012].

Lilly, S., "Use Case Pitfalls: Top 10 Problems from Real Projects Using Use Cases," Proc. of Technology of Object-Oriented Languages and Systems, 1999.

Object Management Group (OMG), 2009. OMG Unified Modeling Language (OMG UML) Superstructure. <http://www.omg.org/spec/UML/2.2/Superstructure/PDF> [Accessed: 19 October 2011]

Overgraad, G. and Palmkvist, K., Use Cases Patterns and Blueprints. Addison-Wesley, 2005.