In Proceedings of the 2012 IEEE Symposium on Visual Languages and Human-Centric Computing, Innsbruk, Austria, Sep 30-Oct 3 2012.

Supporting Requirements Modelling in the Malay Language using Essential Use Cases

Massila Kamalrudin
Faculty of Information & Communication Technology
Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya, 76100 Durian Tunggal,
Melaka, Malaysia,
massila@utem.edu.my

John Grundy
Centre for Computing and Engineering Software Systems
Swinburne University of Technology
PO Box 218, Hawthorn, Victoria 3122,
Australia
jgrundy@swin.edu.au

John Hosking
College of Engineering & Computer Science
The Australian National University
Canberra ACT 0200
Australia
john.hosking@anu.edu.au

***Abstract*— Requirements are typically modelled in natural language, leading to inconsistencies, incompleteness and incorrectness due to inherent natural language ambiguities and lack of precise modelling rules. In previous work, we developed a technique and toolset to support extraction of requirements from English text and supporting semi-formal modelling and round-trip refinement using Essential use cases, helping to mitigate some of these problems. In this paper we describe new work applying this human-centric approach to requirements engineering to the Malay language. We describe an extension of our original Essential Use Cases toolset to support requirements modelling in the Malay language essential interaction modelling, and results of a preliminary experiment to gauge our tool's effectiveness in supporting Malay natural language extraction and round-trip requirements refinement.**

Keywords-Natural language requirements, Essential Use Cases, Requirements Engineering, Round-trip engineering, Human-centric modelling

I. INTRODUCTION

Most requirements in the software industry are written or described using natural language, commonly English. However, many other languages, such as Mandarin, Malay, Hindi, France, Dutch and many more, are also used to elaborate requirements. These natural language requirements are a very "human-centric" modelling approach, readily usable by clients, requirements engineers and developers alike. However, requirements specified purely in natural language often suffer serious problems of ambiguity, incompleteness and inaccuracy [1]. They can also be error-prone due to interpretation problems [2]. This is a particular problem when dealing with the interpretation of multi-lingual requirements, as two different languages might describe requirements for the same system in quite different ways [3].

Multi-lingual requirements are common in Malaysia, as it is a country comprised of multiple ethnic groups, regions, culture and language. Thus, Bahasa Malaysia or Malay language is the official language while English is the second language [4]. The Malay language is one of the core languages in South East Asia with about 200 million people communicate using this language [5]. As an official language of the country, most of the official functions and written communication are written in Malay [4] including requirements provided by/for government sectors. However, English is the preferred language by certain parties especially in the private sector and this leads to "code-switching" [4] between Malay and English communication. This kind of communication is also used in the requirements gathering and elicitation process between the requirements engineers and the clients. This can lead to inconsistency and incorrectness of the requirements.

In previous work we developed a technique and toolset to support the extraction of English-language requirements into the semi-formal Essential Use Cases modelling framework [6], [7]. Our toolset also assists round-trip refinement of requirements to improve their quality, accuracy and precision. We demonstrated that our approach successfully preserves the use of human-centric natural language requirements while significantly addressing the identified quality issues. We wanted to see if our approach would generalise to the Malay language, which has some grammatical differences to English, as a first step in demonstrating our approach generalises to and is effective in other languages. It is also a step on the way to supporting multi-lingual requirements engineering, allowing requirements engineers, end users and developers to work more effectively with requirements in multiple languages.

II. OUR APPROACH

Our original approach centred on a library of essential interaction patterns, a set of generalised English phrases that match to essential interactions, the building blocks of EUCs. Figure 1, from [7], shows the high level extraction process used. In our current work, we have extended our essential interactions pattern library to also cover Malay language based interactions. The requirements fragments, or patterns, kaning up this library extensions are derived from requirements engineering and software engineering books, published literature and published information from software developers' web pages, as well as requirements collected from requirements engineers and business analysts in Malaysia. From there, we identified various key phrases in the Malay language (essential interactions) for particular abstract interactions. Having extended the library, we then extended

our previous automated English language EUC extraction approach and supporting tool to enable requirements engineers to extract accurate EUC abstract interactions automatically from textual natural language requirements written in the Malay language. The main difference from [7] is thus the use of natural language requirements expressed in the Malay language.

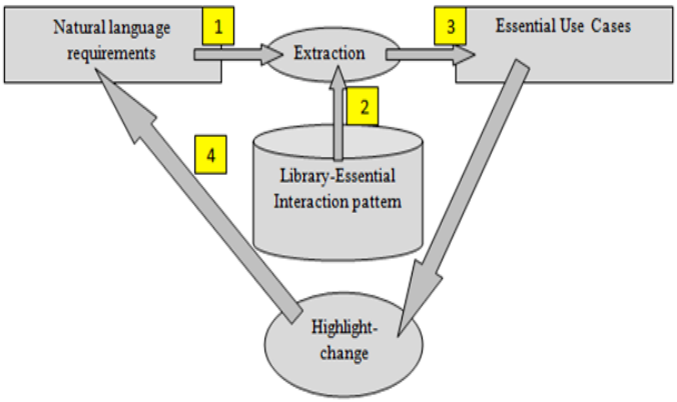

Figure 1 Our Extraction Approach

As seen in Figure 1, the Malay requirements (1) are fed through an extraction process (2) that uses a library of essential interaction phrases and expressions in Malay, producing a sequence of EUC essential requirements. The requirements engineer can then select items in the natural language requirements of EUC interactions (3) and see the corresponding items (4).

III. A New Essential Interaction Pattern library in the Malay language

As described in the previous section, we had to develop a new library of essential interactions patterns of requirements written in the Malay language. The library patterns are used to store all the essential interactions and abstract interactions based on a collection of phrases that illustrate the function or behaviour of a system. Each element of the collection of Malay requirements phrases making up a pattern is categorised based on its related or associated abstract interaction. Similar to our English abstract interaction patterns, a Malay abstract interaction pattern is associated with more than one essential interaction for various domains of application. For example; the abstract interaction "*mengesahkan pengguna*" (identify user) is associated to "*memasukkan kad pengenalan*" (insert identify card), "*membuat proses login*" (do login process) and "*memasukkan id dan kata laluan*" (enter id and password). This particular abstract interaction is associated with the sales system, registration system and election system domains. Table 1. shows examples of other abstract interactions and essential interactions in Malay. To date, we have collected almost 100 essential interactions and 30 abstract interaction patterns in the Malay language and 360 essential interactions and 80 abstract interaction patterns in English. On average, one abstract interaction is associated with three essential interactions.

Table 1. Example of Abstract Interaction and Essential Interactions in Malay language

Abstract interaction	Essential Interaction
Simpan data *(save data)*	Menyimpan rekod *(save the record)*
	Menyimpan maklumat *(save the information)*
	Menyimpan data *(save data)*
Semak status *(check status)*	Menyemak status permohonan *(check an application status)*
	Menyemak status tempahan *(check booking status)*
	Menyemak maklumat terkini *(check current information)*
Daftar ahli *(register member)*	Mendaftarkan kad pengenalan *(register an identity card)*
	Mendaftar nombor id *(register an id number)*
	Mendaftarkan *nama (register name)*

IV. Malay Language EUC Tool

Having extended our essential interaction patterns library for requirements written in the Malay language, our next step was to integrate the Malay EUC library patterns into our tool, MaramaAI, to provide automatic extraction and consistency management for Malay EUCs. MaramaAI is a tool to support consistency management and validation of requirements between natural language and EUCs. It helps to extract essential interactions from natural language requirements and then maps them to a candidate EUC model. It also helps requirements engineers to perform consistency checking using visual traceability support between three components of the requirements: natural language requirements, abstract interactions and EUC model [8].

Figure 2 (1) shows the results of extending MaramaAI with Malay capabilities. Here, requirements in both English and Malay languages are shown side-by-side at the same time. The tool allows the requirements engineer to insert the same natural language requirements, English and Malay at the same time and then compare the results of both requirements expressed as EUCs. With this juxtaposed view, the requirements engineer can easily cross- check and view the consistency and correctness of the abstract interaction and the EUC model for both languages.

Figure 2 (1A) shows an example of extracted abstract interactions and an EUC model for the given Malay requirements. It also shows that the association of the EUC component *"memaparkan maklumat*" (a) to the abstract interaction *"memaparkan maklumat"* (b) and also to the associated essential interactions (c) *"memaparkan kenderaan"* and *"memaparkan maklumat"*. Figure 2 (1B) shows the extraction of same requirements written in English. This shows that there are some differences in the essential interactions associated to the abstract interaction "view detail". "View detail" is similar to "*memaparkan maklumat*" in Malay language. The numbers of associated essential interactions are different compared to the Malay requirements in Figure 2(1A). Only one essential interaction is associated with the English abstract interaction "View detail" but there are three essential interactions associated to the abstract

Malay interaction ”*memaparkan maklumat*”. Thus, while both languages provide similar abstract interaction structures in the EUC model the actual natural language requirements differ somewhat. This highlights to the end user that there might exist inconsistencies in the description of the requirements due to language issues that need to be explored in more detail.

Figure 2(2) shows consistency checking of the Malay requirements. The tool detects inconsistency when any changes such as change of sequence, deletion or update occur. For example in Figure 2(2), the abstract interaction “*memaparkan maklumat*” is moved to the bottom and this leads to an inconsistency warning as well as problem markers (***) to appear to highlight the inconsistencies at both abstract and essential interaction levels.

Figure 2(3) shows consistency checking of Malay requirements for reserving a vehicle against a “best-practice” template in English for “reserve item”. From here, the requirements engineer can check for the correctness and completeness of their Malay language requirements compared to English “best-practice” requirements by identifying missing components or incorrect ordering in the EUC model.

A preliminary evaluation of our extended tool with 10 Malaysian requirements engineers was conducted. This showed that the tool is useful and easy to use for extracting essential interactions and mapping particular abstract interactions to an EUC element in natural language Malay requirements. Several commented that they thought it helped them to more easily identify correct abstract interactions and generate a correct EUC model. They also agree that the tool helps them to detect inconsistencies (if they exist) in requirements written in both languages.

V. Related Work

Some research has been done to handle natural language specifications written in the Malay language. For example Shukur et al. [5] employed formal approaches for translating a software specification in Malay natural language into formal Z statements using their tool called M2Z. This tool is limited to certain grammar, rules and data types. It also works for only small types of statements that represent requirements documents usually found in practice [5]. In contrast, in our work, we perform a semi-formal modelling approach using EUCs and realise it with a tool support and library patterns of essential interactions in the Malay (and English) languages. There are also approaches generating models from natural language requirements. Kof [9] developed an approach translating text to models by integrating NLP with a CASE tool called Autofocus, together with a machine learning technique. A simple discourse model in a LATEX like structure is used to input the requirements documents to the automated tool [9]. The concept of this work is similar with ours as they also transform text to models, but the approach and target models are quite different. Diaz et al. [10] developed a metamorphosis approach to establish the relationship between specification and functional requirements. This is done by using defined patterns and the concept of roles. This approach transforms use case models to interaction models to capture the abstraction of the linguistic text of use cases [10]. We differ by mapping essential interactions and abstract interaction patterns to EUC models.

VI. Conclusions

We have extended our English-language library patterns of essential interactions to support essential interactions in Malay. For this, we have collected a number of system requirements written in the Malay language from the industry and transcribed some requirements originally written in English. To help with the automation of extracting essential interactions and generating an EUC model in the Malay language, we have also extended our tool, MaramaAI, to support the extraction of requirements in the Malay language where it helps Malay speaking requirements engineers to extract EUCs and to trace forward and trace back between EUCs and Malay requirements text. It also provides a visual language and differencing support to help to ease the process of checking and managing consistency between EUCs and natural language requirements. We are adding translation of abstract interactions between both languages to ease the consistency checking of requirements for both languages, supporting multi-lingual development and global software development.

Acknowledgement

This research was funded by support from the Ministry of Higher Education Malaysia (MOHE), FRST Software Process and Product Improvement project, Universiti Teknikal Malaysia Melaka (UTeM), and Swinburne University of Technology.

References

[1] C. Denger and D.M.Berry., "Higher Quality Requirements Specifications through Natural Language Patterns,"IEEE International Conference on Software-Science, Technology & Engineering, IEEE Computer Society, 2003, pp. 80.

[2] I. Diaz, O. Pastor, and A. Matteo, "Modeling interactions using role-driven patterns," 13th IEEE International Conference on Requirements Engineering, 2005, pp. 209-218.

[3] F. Calefato, F. Lanubile, and R. Prikladnicki, "A Controlled Experiment on the Effects of Machine Translation in Multilingual Requirements Meetings," 6th IEEE International Conference on Global Software Engineering, 2011, pp. 94-102.

[4] H. Kassim and F. Ali, "English communicative events and skills needed at the workplace: Feedback from the industry", English for Specific Purposes”, vol. 29, pp. 168-182, 2010.

[5] Z.Shukur, A. M. Zin. A.Ban and S.C. Ping, "A Tool for Translating a Software Specification Written in Malay into a Formal Statement in Z”, Asian Journal of Information Technology vol. 5, pp. 272-279, 2006.

[6] M.Kamalrudin, J.Hosking. J.Grundy, "Improving Requirements Quality using Essential Use Case Interaction Patterns," 2011 International Conference Software Engineering, Honolulu, Hawaii, USA, 2011.

[7] M.Kamalrudin J. Grundy., J. Hosking, "Tool Support for Essential Use Cases to Better Capture Software Requirements", 25th IEEE/ACM International Conference on Automated Software Engineering, Antwerp, Belgium, 2010.

[8] M.Kamalrudin J. Grundy., J.Hosking, , "Managing consistency between textual requirements, abstract interactions and Essential Use Cases," 34th Annual IEEE International Computer Software & Applications, Seoul, Korea, 2010.

[9] L. Kof, "From Requirements Documents to System Models: A Tool for Interactive Semi-Automatic Translation," 18th IEEE Requirements Engineering Conference, 2010, pp. 391-392.

[10] I. Diaz, O. Pastor, and A. Matteo, "Modeling interactions using role-driven patterns," 13th IEEE International Conference on Requirements Engineering, 2005, pp. 209-218.

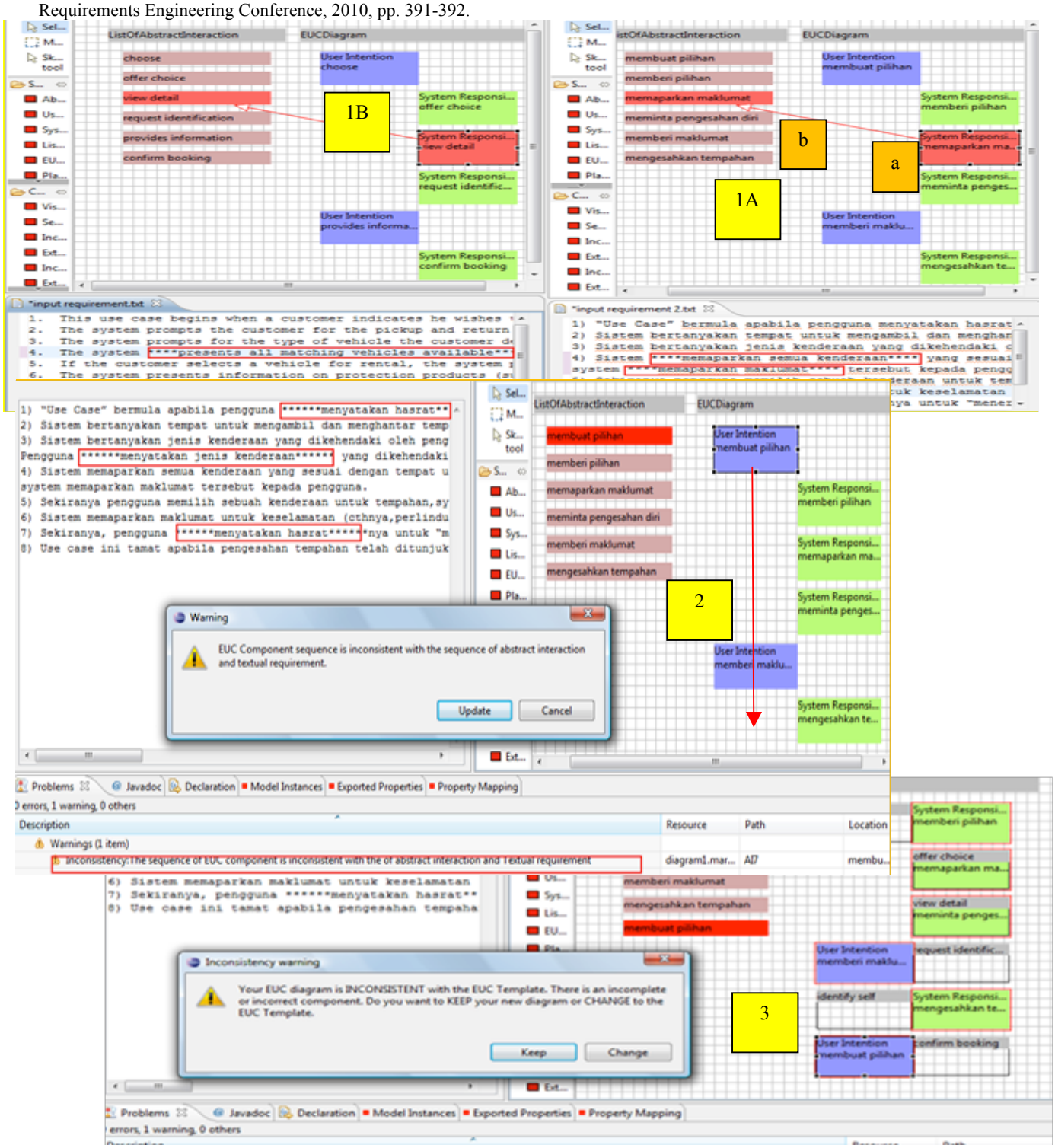

Figure 2 Malay language MaramaAI Tool Usage: (1) tracing extracted requirements to support consistency management; (2) equivalent English language requirements; and (3) indicating a likely inconsistency in the requirements captured.