IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 12, 2018, accepted August 22, 2018, date of publication September 10, 2018, date of current version September 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2867559

Factors Affecting the Metamodelling Acceptance: A Case Study From Software Development Companies in Malaysia

VITALIY MEZHUYEV[1], MOSTAFA AL-EMRAN[1,2], MURNI FATEHAH[1], AND NG CHIN HONG[1]
[1]Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Gambang 26300, Malaysia
[2]Information Technology Department, Al Buraimi University College, Al Buraimi 512, Oman

Corresponding author: Mostafa Al-Emran (malemran@buc.edu.om)

This work was supported by Universiti Malaysia Pahang, Malaysia, under Grant RDU1703107.

ABSTRACT Metamodeling has become a crucial technique in the process of software development. However, the level of metamodeling acceptance is still very low in software development companies. To the best of our knowledge, this paper can be considered as the first that aims to examine the factors that affect the metamodeling acceptance by software engineers. To achieve this aim, the technology acceptance model was adopted and extended by adding additional factors, which are the organizational and team-based factor, perceived maturity, and perceived effectiveness. Data were collected from 152 software engineers from different software development companies in Malaysia. SmartPLS was used to conduct the partial least squares-structural equation modeling (PLS-SEM) for assessing the measurement and structural models. The results indicated that perceived usefulness, perceived ease of use, organizational and team-based factor, perceived maturity, and perceived effectiveness have a significant impact on metamodeling acceptance. The R^2 was 0.887 for metamodeling acceptance, which indicates that the level of model prediction is relatively substantial. Limitations and prospects for future research are also discussed.

INDEX TERMS Metamodelling, TAM, PLS-SEM, acceptance, Malaysia.

I. INTRODUCTION

Model Driven Engineering (MDE) is a software engineering (SE) technique that conceives models as the first phase of the software development life cycle [1]. In MDE, before building a software system, the model that explains the systems' features and functionality should be developed [2]. Metamodelling is the MDE technique that describes the model's abstraction levels and their associations with one another [3]. Metamodelling also refers to the precise syntax in which the model should be identified through an explicit modelling language [4].

Currently, there are several metamodelling frameworks. The most popular one is the Meta Object Facility (MOF) that was initiated by the Object Management Group (OMG). The Eclipse Modelling Framework (EMF) gives the implementation of an essential MOF (eMOF). Based on the MOF, there are four layers of the metamodelling architecture, which are M3, M2, M1, and M0. M3 is the highest level, which describes a meta-metamodel as the infrastructure for the metamodelling architecture. M2 is the metamodelling layer, which defines the concepts of class, attribute, association, etc. for software systems modelling. M1 is the model layer, which describes the instances of the metamodel and the information domain. The lowest layer M0 contains user-defined objects, which are the instances of the model and describe domain-specific information.

Metamodelling is considered as a part of language engineering in general, and Domain Specific Languages (DSL) in particular [5]. A DSL represents a software system in the concepts of domain, but not universal concepts of algorithms and data. Thus, a DSL gives a notation to model the knowledge of a specific domain. DSL elements are the typical concepts and relationships of a particular domain. For example, Business Process Model and Notation (BPMN) is one of the DSLs used for a business process modelling. BPMN has modelling elements such as activities, events, gateways and connecting objects relationships like sequence flow, message flow and data association. In addition, the DSL has been successfully integrated in many SE applications such as automation of a measurement system [6], safety-critical software development [7], and composition and verification of multidisciplinary models [8].

Although metamodelling technology is important, the level of its industrial acceptance is still very low, and this problem has been overlooked by researchers. The issue may refer to the lack of educational background of software engineers as well as the inertia of the software development industry. To the best of our knowledge, there is no research devoted to exploring the factors that influence the metamodelling acceptance. This research conducts an empirical study to evaluate the software engineers' attitudes towards the use of metamodelling in their software development companies. The theoretical basis for the evaluation is the two main constructs of the Technology Acceptance Model (TAM), which include perceived usefulness and perceived ease of use. Thus, the TAM has been modified by adding other factors, which were derived from the existing literature.

II. LITERATURE REVIEW

Metamodelling is the core of MDE, where it is used for language engineering and domain modelling. Thus, the acceptance of metamodelling can be considered in the broader context of MDE adoption.

A study by [2] proposed the use of MDE for automating data engineering tasks such as schema integration or conversion which can be implemented in an information system. Besides, a study by [9] addressed the use of MDE in an information system by focusing on multilevel modelling to tackle the interoperability in heterogeneous ecosystem in the oil and Gas industry. In terms of metamodelling, a study by [10] used the metamodelling approach in a security-critical system for the purpose of integrating other models in multi-paradigm modelling. Furthermore, a study by [11] used the metamodelling technique in a multi-agent system for supporting real-time requirement constraint during the early stage of the software development process to reduce the risk of software failure.

A study by [12] argued that MDE technologies are incorporated in a large-scale commercial software development. In addition, a study by [13] concluded that while MDE has been widely accepted by academia, its industrial acceptance is very limited. It is argued that many of the challenges restricting industrial adoption are linked to the lack of mature MDE tools. The focus of researchers was on areas where existing tools and frameworks were not sufficient, and they especially emphasized on metamodel development. One of the few successful examples of metamodelling applications is the study of [14], in which the use of the MetaEdit+ tool for the development of Domain-Specific Modelling Languages (DSML) in industrial projects for more than 10 years was discussed. As indicated by [15], the MetaEdit+ tool remains one of the most mature tools used in the industrial domain. At the same time, a study by [16] argued that MDE is still in the early phases of being successfully adopted by industry. The reason for this could be that MDE must prove its advantages over other development paradigms and be supported by standardized tools. Another precondition for successful adoption of MDE is that MDE should not introduce more complexity than it removes.

It is argued by [17] that despite the wide interest in MDE, there are no mature tools that support the UML-based MDE approach. The study focused on addressing an industrial relevance for these types of tools. A study by [18] presented results from empirical research on MDE adoption. Based on the qualitative questionnaire and interviews, the scholars investigated a range of technical, organizational, and social factors that influenced an organizational response to MDE. One of the factors investigated was the MDE perception, which is regarded as a successful or unsuccessful organizational intervention. To understand the MDE adoption in industry, they also learned social, technical, and organizational factors. In the same context, another study by [19] described the practices of three commercial organizations that adopted an MDE approach to their software development. The scholars used in-depth semi-structured interviews to learn about the experiences of MDE practitioners. This allowed the researchers to identify the importance of complex organizational, managerial, and social factors as opposed to simple technical factors in the success or failure of the MDE adoption. The study argued that the effective deployment of MDE requires a progressive and iterative technique, transparent institutional commitment and motivation, integration with relevant organizational processes, and a well-defined business target.

According to [20], despite the availability of published evidence that MDE significantly improves the developer's productivity and product quality in industrial projects, MDE adoption is still surprisingly low. The technical and non-technical causes behind this issue were reviewed, and the requirements for MDE to become a reliable approach to software development were outlined. A study by [21] discussed how model-based techniques can enhance the daily practice of software specialists. It is anticipated that MDE adoption in the software industry will grow exponentially in the near future due to the convergence of software development and business analysis processes. A study by [22] discussed the impact of tools on MDE adoption. The authors of the study argued that the lack of adoption of MDE is due to the poor tool support within an organization. The data taken from interviews with MDE practitioners were analyzed in the context of the adoption of MDE tools.

According to [23], MDE has become an effective technique used in industry. Many companies are beginning to evaluate the possibilities for adopting it. The paper examined the current state of MDE use in the software industry, summarized the current obstacles for the adoption, and discussed MDE's advantages and limitations. Finally, the study outlined ideas for a smoother transition toward a wider industrial adoption of MDE. Recently, a study by [24] argued that while poor tool support is often blamed for the low adoption of MDE, the problem is also linked with social and organizational factors. They revisited the past data on MDE usage in industry to categorize the issues that users had with the tools they adopted. Based on the empirical data, they formulated a taxonomy of tool-related considerations, which reflects the MDE tooling.

Based on the given analysis, the level of MDE's industrial acceptance is still very low, and this problem is a topic in the existing literature. According to the literature, it has been observed that there is no research devoted to exploring the factors that influence the MDE acceptance. Although studies like [18], [19], and [24] presented the organizational factors impacting MDE adoption, the reasons for MDE acceptance need further investigation. The focus of this study is the metamodelling since it is the core of MDE. Based on the previous assumptions, the present study empirically evaluates the software engineers' attitudes towards the use of metamodelling in their software development companies. The present study adopts the two main constructs of the TAM [25] (i.e., "perceived usefulness" and "perceived ease of use") along with the other factors, which are the organizational and team-based factor, perceived maturity, and perceived effectiveness aiming to study the metamodelling acceptance. The selection of the TAM constructs could be the reason that the TAM is one of the most popular information systems (IS) models [26], and this popularity tends to explain the users' acceptance of any IS or technology. Furthermore, based on the existing literature, it can be seen that the TAM has not been used to examine the metamodelling acceptance.

III. RESEARCH HYPOTHESES

The literature review allows the authors to identify the relevant factors to evaluate the software engineers' acceptance of metamodelling technology. The proposed model was constructed based on the two main constructs of TAM: perceived usefulness and perceived ease of use, along with the other factors including the organizational and team-based factor, perceived maturity, and perceived effectiveness as additional factors from the existing literature. Fig. 1 shows the proposed model. It assumes that metamodelling technology acceptance is affected by five factors, namely perceived usefulness, perceived ease of use, organizational and team-based factor, perceived maturity, and perceived effectiveness. It also suggests that perceived usefulness is affected by perceived ease of use.

A. PERCEIVED USEFULNESS (PU)

Perceived usefulness (PU) refers to "the degree to which a person believes that using a particular system would enhance his or her job performance" [25]. In the software development companies' context, the higher the usefulness of metamodelling in the process of software development, the higher the software engineers' acceptance of such technology. According to [27], PU enhances the performance of software measures, which in turn leverages its acceptance. The PU was found to affect the acceptance of different technologies such as mobile learning [28], e-books [29], Google

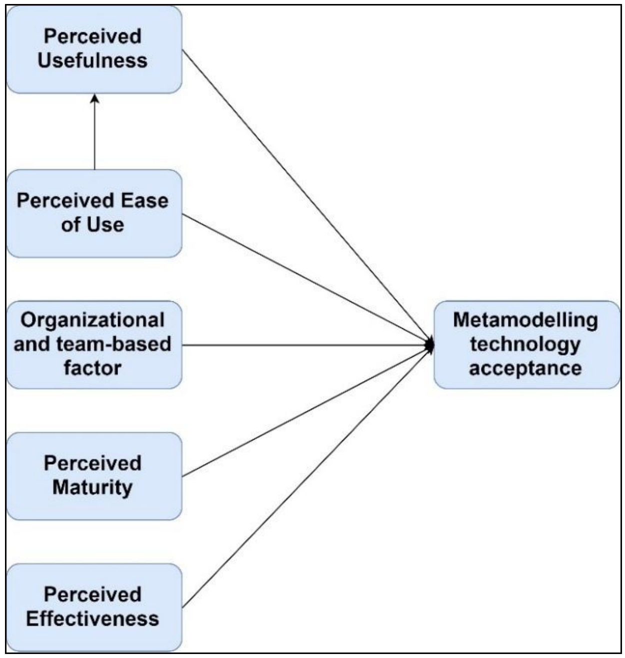

FIGURE 1. The proposed model.

classroom [30], corporate blogs [31], and social media [32], among many others. Thus, we hypothesized the following:

H1: There is a significant positive relationship between perceived usefulness and metamodelling technology acceptance.

B. PERCEIVED EASE OF USE (PEOU)

Perceived ease of use (PEOU) is defined as "the degree to which a person believes that using a particular system would be free from effort" [25]. In the software development companies' context, when metamodelling is perceived to be easy to use, it is more likely to be accepted by software engineers. According to [27], the PEOU makes it easy to increase the usage of software measures, which in turn increases its acceptance. Furthermore, the PEOU of the software measure can affect the PU of that measure. The PEOU was found to affect the acceptance and PU of various technologies, such as mobile learning [28], e-government services [33], cloud computing [34], and social media [35], among many others. Accordingly, we hypothesize the following:

H2: There is a significant positive relationship between perceived ease of use and metamodelling technology acceptance.

H3: There is a significant positive relationship between perceived ease of use and perceived usefulness.

C. ORGANIZATIONAL AND TEAM-BASED FACTOR (OTB)

The organizational and team-based factor is a key factor in predicting the acceptance of several technologies and systems. In the context of this study, several studies like [18], [19], and [24] studied the impact of an organizational and team-based factor on MDE adoption. However, further investigation is required in terms of MDE in general and metamodelling in particular. It is suggested that if there is sufficient training and support for metamodelling from the organization, the software engineers' acceptance of metamodelling will be enhanced. Therefore, we hypothesize the following:

H4: There is a significant positive relationship between the organizational and team-based factor and metamodelling technology acceptance.

D. PERCEIVED MATURITY (PM)

Perceived maturity (PM) is defined as "the degree to which tools are perceived as mature and suitable for the tasks in hand" [36]. PM can also refer to the function of technology uncertainty and inexperience [37]. PM is considered a critical factor in the assimilation of OOP languages [38]. In the context of SE, PM is regarded as an essential factor for determining the end-users' acceptance of a particular technology [38], [39]. In the context of this study, studies like [36] and [40] indicated that PM has a significant effect on MDE acceptance. Nevertheless, further examination is required in terms of MDE in general and metamodelling in specific. It is assumed that if metamodelling tools are reliable and mature enough, the software engineers' acceptance of metamodelling will be improved. Hence, we hypothesize the following:

H5: There is a significant positive relationship between perceived maturity and metamodelling technology acceptance.

E. PERCEIVED EFFECTIVENESS (PE)

Perceived effectiveness (PE) refers to "the degree to which using a technology will provide benefits to consumers in performing certain activities" [41]. According to the existing literature, PE was found to have a significant effect on the acceptance of many technologies like web-based learning systems [42], location-based advertising messaging [43], information security [44], and e-learning [45], among many others. In the context of SE, a study by [39] indicated that PE has a significant impact on computer-aided software engineering (CASE) technology. It is suggested that the higher the benefits of metamodelling in the process of software development, the higher the software engineers' acceptance of such technology. Therefore, we hypothesize the following:

H6: There is a significant positive relationship between perceived effectiveness and metamodelling technology acceptance.

IV. RESEARCH METHODOLOGY

A. CONTEXT AND SUBJECTS

The target population of this study is the software engineers working at different software development companies in Malaysia. All the companies use different software development methodologies in the process of developing multiple-purpose software. A total of 152 valid responses were received out of 1375 emails sent including the survey link, which indicates a response rate of 11%.

B. SURVEY INSTRUMENT

An online questionnaire survey was sent through email to all the software engineers for the purpose of data collection. The survey consisted of three different parts. The first part was a general information about the respondents. The second part was dedicated to evaluating the respondents' attitudes towards the use of metamodelling technology based on the proposed factors (perceived usefulness, perceived ease of use, organizational and team-based factor, perceived maturity, and perceived effectiveness). Those factors were used as determinants for users' acceptance of metamodelling technology in the proposed model. The third part included open-ended questions in which respondents were required to give their comments and recommendations to implement the metamodelling technology in their company. The items used for perceived usefulness and perceived ease of use constructs were adopted from [25] with some modifications to suit the purpose of the current study. In addition, the items used for the organizational and team-based factor were adopted from [27], while items used for the perceived maturity were adopted from [38], and perceived effectiveness from [42] and [46]. The list of the constructs and their corresponding items are detailed in the Appendix.

C. DATA ANALYSIS

In this study, we followed the suggested methodological guidelines for conducting IS adoption research [47]. In that, the Partial Least Squares-Structural Equation Modelling (PLS-SEM) method using SmartPLS 3 is employed for undertaking the statistical analysis [48]. Due to the reason that this study is exploratory in nature, the PLS-SEM is suggested to be the appropriate approach for this kind of studies [49]. In terms of the measurement model, [49] suggested that researchers should measure the outer loadings of the items and the average variance extracted (AVE) for establishing the convergent validity. Additionally, [49] recommended two measures for confirming a discriminant validity including cross loading and Fornell-Larcker criterion. Moreover, [50] proposed measuring the Heterotrait-Monotrait ratio (HTMT) as another measure for evaluating the discriminant validity. In terms of the structural model, the path coefficients and the coefficient of determination (R^2) will be measured [49]. Consequently, all the aforementioned criteria were followed in order to evaluate the measurement and structural models.

V. RESULTS AND DISCUSSION

A. DESCRIPTIVE ANALYSIS

The first part of the survey concerns about the demographic information of the respondents, such as age, gender, country, and educational level. It also covers the development methodology used by the company and the type of software

TABLE 1. Measurement model results.

Constructs	Items	Loadings	Cronbach's α	Composite Reliability	Average Variance Extracted
Metamodelling Technology Acceptance	MA1	0.988	0.975	0.988	0.976
	MA2	0.988			
Organizational and team-based factor	OTB1	0.953	0.969	0.976	0.890
	OTB2	0.934			
	OTB3	0.941			
	OTB4	0.931			
	OTB5	0.958			
Perceived Ease of Use	PEOU1	0.982	0.967	0.979	0.939
	PEOU2	0.948			
	PEOU3	0.976			
Perceived Usefulness	PU1	0.983	0.973	0.979	0.905
	PU2	0.931			
	PU3	0.974			
	PU4	0.920			
	PU5	0.947			
Perceived Maturity	PM1	0.959	0.887	0.930	0.817
	PM2	0.965			
	PM3	0.775			
Perceived Effectiveness	PE1	0.989	0.983	0.988	0.966
	PE2	0.971			
	PE3	0.990			

companies. Among the 152 respondents, 53.9% are males, and 46.1% are females. In addition, the respondents are aged between 20 to 27 years (52.6%), followed by 28 to 39 years (44.1%) and 40 to 54 years (2.6%), respectively. Additionally, the majority of the respondents hold bachelor's degrees (75%), while respondents who are still undergraduates cover the second largest portion in the survey (15.1%), followed by respondents with a master's level degree (6.6%). There are a few respondents who hold a doctorate degree, at only 3.3% of the total respondents. Furthermore, 14.5% of the respondents are working in companies with 250 employees and above, while only 1.3% of the respondents are working in companies with 1-5 employees. In terms of the software development methodology used, most of the respondents apply the agile software development in their company. The second most popular methodology is the Software Development Life Cycle (SDLC), which approximately comprises 53.3% of the companies use. The third most popular method is the Spiral method with 26.3%, followed by the Waterfall with 20.4%, the Scrum method with 17.8%, and the Rapid Application Development (RAD) with 17.1%, respectively. In terms of metamodelling technology awareness, 88.8% of the software engineers indicated that they do not have experience with this technology, while only 11.2% are experienced with this technology.

B. QUALITATIVE ANALYSIS

The third part of the survey analyses the reasons preventing the use of metamodelling technology and gathers suggestions from respondents to increase the level of industrial acceptance of the metamodelling. Most of the respondents indicated that the main factor that may prevent the use of metamodelling technology in their company is the increasing cost of software design and development. This is because experts are hard to find in this field, so hiring experts and purchasing tools for their company are usually expensive. Furthermore, training will be costly to the company since many employees are new to this technology, and sufficient training is required for them to become familiar with metamodelling. Software engineers also feel that they do not have adequate knowledge of the metamodelling technology. As such, they do not know what the technology is and how it can be implemented.

To increase the level of industrial acceptance of the metamodelling, respondents agreed that training will be an effective procedure to be performed and organized by the decision makers of the company. Additionally, some respondents suggested that instead of providing training in the company itself, this procedure can be organized as a course in a university so that they will have sufficient knowledge in metamodelling, which can then be implemented in their company when there is a need. Moreover, conferences, seminars, and real case studies are highly encouraged by respondents to increase the level of industrial acceptance of the metamodelling technology.

C. MEASUREMENT MODEL ASSESSMENT

To measure the reliability of each item, the factor loading should be measured. According to [49], a threshold value equal to or greater than 0.7 for each item's loading is considered as reliable. In addition, the Cronbach's Alpha and composite reliability values should be equal to or greater than 0.7. According to Table 1, it can be seen that all the items are reliable and meet the suggested criteria. Moreover, the AVE refers to the grand mean value of the squared loadings of the items related to the construct and is considered the main measure for confirming the convergent validity. A value equal to or greater than 0.5 for the AVE indicates that the variable explains more than half of the variance of its items [49]. Based on Table 1, the values of Cronbach's Alpha and composite reliability are all greater than 0.7, and the values of AVE are greater than 0.5. Hence, this shows that the convergent validity is confirmed.

To confirm the discriminant validity, the Fornell-Larcker criterion, cross-loadings, and the HTMT ratio should be measured. In terms of the Fornell-Larcker criterion, the square root of the AVE (diagonal value) for each construct should exceed the correlation of latent constructs, which is satisfied in the current study as shown in Table 2. In terms of the cross-loadings, the loading of each indicator should be higher than the loadings of its corresponding constructs' indicators. According to Table 3, it can be seen that the cross-loadings criterion is met. With respect to the HTMT ratio, a value of less than 0.85 for HTMT should be fulfilled. Based on Table 4, we can notice that the HTMT criterion is satisfied, and therefore, showing that the discriminant validity is confirmed.

TABLE 2. Fornell-larcker criterion results.

	MA	OTB	PEOU	PE	PM	PU
MA	**0.988**					
OTB	0.651	**0.943**				
PEOU	0.812	0.491	**0.969**			
PE	0.827	0.578	0.652	**0.983**		
PM	0.766	0.602	0.676	0.593	**0.904**	
PU	0.821	0.542	0.695	0.677	0.615	**0.951**

TABLE 3. Cross-loadings results.

	MA	OTB	PE	PEOU	PM	PU
MA1	**0.988**	0.632	0.779	0.806	0.780	0.812
MA2	**0.988**	0.654	0.854	0.797	0.733	0.810
OTB1	0.657	**0.953**	0.591	0.512	0.620	0.549
OTB2	0.582	**0.934**	0.495	0.453	0.559	0.478
OTB3	0.593	**0.941**	0.538	0.436	0.515	0.491
OTB4	0.603	**0.931**	0.514	0.471	0.579	0.506
OTB5	0.631	**0.958**	0.583	0.439	0.564	0.528
PE1	0.836	0.581	**0.989**	0.660	0.594	0.687
PE2	0.789	0.556	**0.971**	0.623	0.579	0.646
PE3	0.812	0.568	**0.990**	0.640	0.575	0.664
PEOU1	0.819	0.469	0.655	**0.982**	0.661	0.697
PEOU2	0.704	0.453	0.571	**0.948**	0.642	0.619
PEOU3	0.826	0.502	0.663	**0.976**	0.660	0.698
PM1	0.769	0.527	0.590	0.701	**0.959**	0.631
PM2	0.768	0.545	0.585	0.695	**0.965**	0.618
PM3	0.495	0.601	0.406	0.374	**0.775**	0.376
PU1	0.831	0.550	0.661	0.697	0.621	**0.983**
PU2	0.764	0.542	0.695	0.635	0.539	**0.931**
PU3	0.815	0.514	0.659	0.692	0.593	**0.974**
PU4	0.734	0.465	0.619	0.613	0.510	**0.920**
PU5	0.755	0.504	0.588	0.662	0.658	**0.947**

TABLE 4. Heterotrait-monotrait ratio (HTMT) results.

	MA	OTB	PEOU	PE	PM	PU
MA						
OTB	0.669					
PEOU	0.832	0.505				
PE	0.844	0.591	0.667			
PM	0.807	0.666	0.706	0.625		
PU	0.842	0.557	0.713	0.693	0.645	

D. STRUCTURAL MODEL ASSESSMENT

The explanatory power of the model is assessed by measuring the discrepancy amount in the dependent variables of the model. With regard to [49], the R^2 and path coefficients are considered the main measures for evaluating the structural model. As depicted in Fig. 2, the model has an R^2 value of 88.7% for metamodelling technology acceptance and 48.3% for perceived usefulness.

With respect to path analysis, Fig. 2 and Table 5 describe the path coefficients and p-values for each hypothesis. It is clearly shown that all the hypotheses are supported, which in turn reveals that all the paths are significant between the independent and dependent variables. $\mathbf{H_1}$ ($B = 0.270$, $p < 0.05$) describes the path between perceived usefulness and metamodelling technology acceptance, indicating that the perceived usefulness enhances the acceptance of metamodelling technology by software engineers. $\mathbf{H_2}$ ($B = 0.240$, $p < 0.05$) shows the path between the perceived ease of use and metamodelling technology acceptance, representing that the perceived ease of use leverages the software engineers' acceptance of metamodelling technology. $\mathbf{H_3}$ ($B = 0.695$, $p < 0.001$) demonstrates the path between the perceived ease of use and perceived usefulness, revealing that the perceived ease of use positively influences the perceived usefulness of metamodelling technology. $\mathbf{H_4}$ ($B = 0.082$, $p < 0.05$) describes the path between the organizational and team-based factor and metamodelling technology acceptance, indicating that the organizational and team-based factor significantly affects the software engineers' acceptance of the metamodelling technology. $\mathbf{H_5}$ ($B = 0.196$, $p < 0.05$) shows the path between the perceived maturity and metamodelling technology acceptance, indicating that the perceived maturity enhances the acceptance of metamodelling technology by software engineers. $\mathbf{H_6}$ ($B = 0.324$, $p < 0.01$) demonstrates

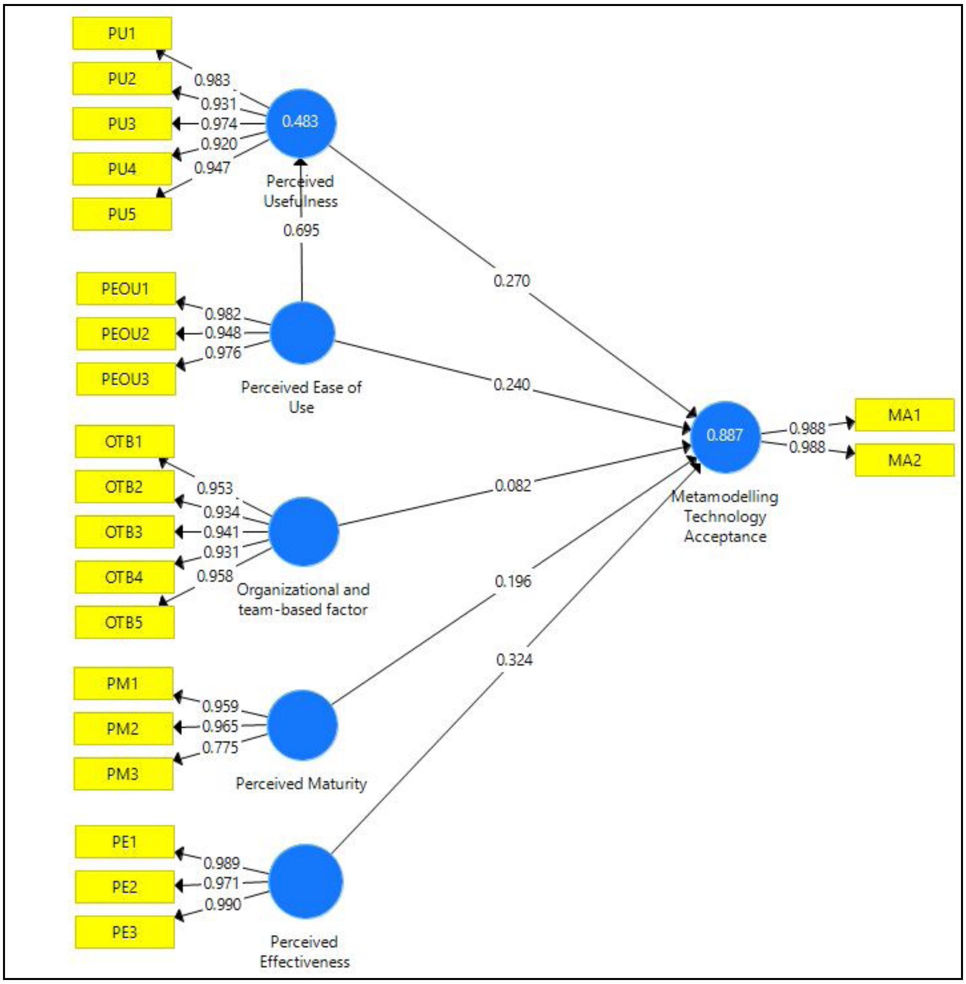

FIGURE 2. Path analysis results.

TABLE 5. Hypotheses test results.

Hypothesis	Path	Path Coefficient	Remarks
H1	PU → MA	0.270*	Supported
H2	PEOU → MA	0.240*	Supported
H3	PEOU → PU	0.695***	Supported
H4	OTB → MA	0.082*	Supported
H5	PM → MA	0.196*	Supported
H6	PE → MA	0.324**	Supported

Note: ***p < 0.001; **p < 0.01; *p < 0.05.

the path between the perceived effectiveness and metamodelling technology acceptance, revealing that the perceived effectiveness leverages the software engineers' acceptance of metamodelling technology. The results of this research study suggest that the PEOU, PU, OTB, PM, and PE positively affect the metamodelling technology acceptance by software engineers who perceive the use of metamodelling as easy and useful, and they are highly motivated towards the use of this technology in their company.

VI. LIMITATIONS AND FUTURE WORK

The response rate of the participants was relatively low. Only 152 responses were gathered out of 1375 emails sent. A small number of participants do not represent all the software engineers' groups and cannot give a reliable picture of the metamodelling technology acceptance in software development companies in Malaysia. In future work, the number of respondents needs to be increased to have a better understanding of the studied problem. Other than Malaysia, other countries need to be added in the survey to deduce a possible difference in responses. In addition, the TAM has been extended by adding three additional factors including the organizational and team-based factor, perceived maturity, and perceived effectiveness. In future work, additional research is required to examine the impact of other factors that may affect the metamodelling technology acceptance.

VII. CONCLUSION

The contribution of this paper is twofold. First, this is the first study that attempts to examine the metamodelling technology acceptance in general and to focus on software engineers in particular. Second, the TAM has been extended by adding three additional factors, which are the organizational and team-based factor, perceived maturity, and perceived effectiveness to examine the metamodelling technology acceptance. Our results indicated that perceived usefulness, perceived ease of use, organizational and team-based factor, perceived maturity, and perceived effectiveness have a significant positive effect on metamodelling technology acceptance. Additionally, perceived ease of use was found to affect the perceived usefulness of metamodelling technology. This is considered a new finding to the existing body of knowledge, as the literature has neglected to examine the impact of such factors on metamodelling acceptance.

These empirical results have managerial implications which are relevant for the policymakers and IT managers of the software development companies in the developing countries in general, and the Malaysian software companies in particular. In that, the decision makers of these companies need to schedule training sessions regarding metamodelling for their software engineers in order to enhance their awareness, and thereby, increase their industrial acceptance of metamodelling. Moreover, since perceived usefulness, perceived ease of use, organizational and team-based factor, perceived maturity, and perceived effectiveness were found to have a significant effect on metamodelling technology acceptance, the IT managers should take these factors into consideration while employing the metamodelling in their companies.

APPENDIX

Perceived Usefulness

PU1: Metamodelling technology can improve the software development process.

PU2: Metamodelling technology can accomplish software development tasks more quickly.

PU3: Metamodelling technology can save a lot of time.

PU4: Metamodelling technology can increase the productivity of software development.

PU5: Metamodelling technology can enhance the quality of software development.

Perceived Ease of Use

PEOU1: Metamodelling technology is easy to use.

PEOU2: Metamodelling technology requires no training.

PEOU3: Metamodelling technology is convenient.

Organizational and team-based factor

OTB1: My company has experts in the metamodelling technology.

OTB2: My company has strong competence in using the metamodelling technology.

OTB3: My company should provide training for the metamodelling technology.

OTB4: My company has projects that require the application of metamodelling technology.

OTB5: Metamodelling technology is supported in my company.

Perceived Maturity

PM1: Metamodelling technology has mature and reliable tools.

PM2: Experiences of metamodelling technology are accumulated in the IT industry.

PM3: Metamodelling technology has many real-world applications.

Perceived Effectiveness

PE1: Metamodelling technology provides effective solutions.

PE2: Metamodelling technology avoids spending excessive time on unimportant tasks.

PE3: Metamodelling technology enables hiding the complexity of large software systems.

Metamodelling technology acceptance

MA1: I like to use the metamodelling technology.

MA2: I am positive towards the metamodelling technology.

REFERENCES

[1] S. Martínez, S. Gérard, and J. Cabot, "On watermarking for collaborative model-driven engineering," *IEEE Access*, vol. 6, pp. 29715–29728, 2018.

[2] F. J. B. Ruiz, J. G. Molina, and O. D. García, "On the application of model-driven engineering in data reengineering," *Inf. Syst.*, vol. 72, pp. 136–160, Dec. 2017.

[3] C. Raibulet, F. A. Fontana, and M. Zanoni, "Model-driven reverse engineering approaches: A systematic literature review," *IEEE Access*, vol. 5, pp. 14516–14542, 2017.

[4] S. Jacome and J. de Lara, "Controlling meta-model extensibility in model-driven engineering," *IEEE Access*, vol. 6, pp. 19923–19939, 2018.

[5] S. Zschaler, D. S. Kolovos, N. Drivalos, R. F. Paige, and A. Rashid, "Domain-specific metamodelling languages for software language engineering," in *Software Language Engineering* (Lecture Notes in Computer Science) (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Berlin, Germany: International Conference on Software Language Engineering, 2009.

[6] T. Kos, M. Mernik, and T. Kosar, "Test automation of a measurement system using a domain-specific modelling language," *J. Syst. Softw.*, vol. 11, pp. 74–88, Jan. 2016.

[7] M. Voelter *et al.*, "Using language workbenches and domain-specific languages for safety-critical software development," *Softw. Syst. Model.*, pp. 1–24, 2018.

[8] P. O. Denno, A. N. Narayanan, A. Kulkarni, D. Balasubramanian, and G. Karsai, "A domain-specific language for model composition and verification of multidisciplinary models," in *Proc. Conf. Syst. Eng. Res.*, 2016, pp. 1–10.

[9] M. Selway, M. Stumptner, W. Mayer, A. Jordan, G. Grossmann, and M. Schrefl, "A conceptual framework for large-scale ecosystem interoperability and industrial product lifecycles," *Data Knowl. Eng.*, vol. 109, pp. 85–111, May 2017.

[10] C. De Faveri, A. Moreira, and V. Amaral, "Multi-paradigm deception modeling for cyber defense," *J. Syst. Softw.*, vol. 141, pp. 32–51, Jul. 2018.

[11] A. Ashamalla, G. Beydoun, and G. Low, "Model driven approach for real-time requirement analysis of multi-agent systems," *Comput. Lang. Syst. Struct.*, vol. 50, pp. 127–139, Dec. 2017.

[12] B. Hailpern and P. Tarr, "Model-driven development: The good, the bad, and the ugly," *IBM Syst. J.*, vol. 45, no. 3, pp. 451–461, 2006.

[13] A. Rutle and A. Rossini, "A tentative analysis of the factors affecting the industrial adoption of MDE," in *Proc. 1st Int. Workshop Challenges Model Driven Softw. Eng.*, 2008, pp. 57–61.

[14] J.-P. Tolvanen and S. Kelly, "MetaEdit+: Defining and using integrated domain-specific modeling languages," in *Proc. 24th ACM SIGPLAN Conf. Companion Object Oriented Program. Syst. Lang. Appl.*, 2009, pp. 819–820.

[15] J.-P. Tolvanen, "MetaEdit+ for collaborative language engineering and language use (tool demo)," in *Proc. ACM SIGPLAN Int. Conf. Softw. Lang. Eng. (SLE)*, 2016, pp. 41–45.

[16] P. Mohagheghi, M. A. Fernandez, J. A. Martell, M. Fritzsche, and W. Gilani, "MDE adoption in industry: Challenges and success criteria," in *Models in Software Engineering* (Lecture Notes in Computer Science) (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5421. Berlin, Germany: International Conference on Model Driven Engineering Languages and Systems, 2009, pp. 54–59.

[17] T. Vepsäläinen, S. Sierla, J. Peltola, and S. Kuikka, "Assessing the industrial applicability and adoption potential of the AUKOTON model driven control application engineering approach," in *Proc. IEEE Int. Conf. Ind. Informat. (INDIN)*, Jul. 2010, pp. 883–889.

[18] J. Hutchinson, J. Whittle, M. Rouncefield, and S. Kristoffersen, "Empirical assessment of MDE in industry," in *Proc. 33rd Int. Conf. Softw. Eng. (ICSE)*, 2011, p. 471.
[19] J. Hutchinson, M. Rouncefield, and J. Whittle, "Model-driven engineering practices in industry," in *Proc. 33rd Int. Conf. Softw. Eng.*, 2011, pp. 633–642.
[20] B. Selic, "What will it take? A view on adoption of model-based methods in practice," *Softw. Syst. Model.*, vol. 11, no. 4, pp. 513–526, 2012.
[21] M. Brambilla, J. Cabot, and M. Wimmer, "Model-driven software engineering in practice," *Synth. Lect. Softw. Eng.*, vol. 1, no. 1, pp. 1–182, 2012.
[22] J. Whittle, J. Hutchinson, M. Rouncefield, H. Burden, and R. Heldal, "Industrial adoption of model-driven engineering: Are the tools really the problem?" in *Model-Driven Engineering Languages and Systems* (Lecture Notes in Computer Science) (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 8107. Berlin, Germany: International Conference on Model Driven Engineering Languages and Systems, 2013, pp. 1–17.
[23] A. Vallecillo, "On the industrial adoption of model driven engineering. Is your company ready for MDE?" *Int. J. Inf. Syst. Softw. Eng. Big Companies*, vol. 1, no. 1, pp. 52–68, 2014.
[24] J. Whittle, J. Hutchinson, M. Rouncefield, H. Burden, and R. Heldal, "A taxonomy of tool-related issues affecting the adoption of model-driven engineering," *Softw. Syst. Model.*, vol. 16, no. 2, pp. 313–331, 2017.
[25] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quart.*, vol. 13, no. 3, pp. 319–340, 1989.
[26] M. Al-Emran, V. Mezhuyev, and A. Kamaludin, "Technology acceptance model in m-learning context: A systematic review," *Comput. Educ.*, vol. 125, pp. 389–412, Oct. 2018.
[27] L. G. Wallace and S. D. Sheetz, "The adoption of software measures: A technology acceptance model (TAM) perspective," *Inf. Manag.*, vol. 51, no. 2, pp. 249–259, 2014.
[28] M. A. Almaiah, M. A. Jalil, and M. Man, "Extending the TAM to examine the effects of quality features on mobile learning acceptance," *J. Comput. Educ.*, vol. 3, no. 4, pp. 453–485, 2016.
[29] S. Lee, "An integrated adoption model for e-books in a mobile environment: Evidence from South Korea," *Telemat. Informat.*, vol. 30, no. 2, pp. 165–176, 2013.
[30] R. A. S. Al-Maroof and M. Al-Emran, "Students acceptance of Google classroom: An exploratory study using PLS-SEM approach," *Int. J. Emerg. Technol. Learn.*, vol. 13, no. 6, pp. 112–123, 2018.
[31] N. Koenig and C. Schlaegel, "Effects of design characteristics on corporate blog acceptance," *Manag. Res. Rev.*, vol. 37, no. 4, pp. 409–440, 2014.
[32] S. J. Kwon, E. Park, and K. J. Kim, "What drives successful social networking services? A comparative analysis of user acceptance of Facebook and Twitter," *Social Sci. J.*, vol. 51, no. 4, pp. 534–544, 2014.
[33] J. G. Cegarra-Navarro, A. Garcia-Perez, and J. L. Moreno-Cegarra, "Technology knowledge and governance: Empowering citizen engagement and participation," *Government Inf. Quart.*, vol. 31, no. 4, pp. 660–668, 2014.
[34] I. Arpaci, "Antecedents and consequences of cloud computing adoption in education to achieve knowledge management," *Comput. Hum. Behav.*, vol. 70, pp. 382–390, May 2017.
[35] S. Fosso Wamba, M. Bhattacharya, L. Trinchera, and E. W. T. Ngai, "Role of intrinsic and extrinsic factors in user social media acceptance within workspace: Assessing unobserved heterogeneity," *Int. J. Inf. Manage.*, vol. 37, no. 2, pp. 1–13, 2017.
[36] P. Mohagheghi, W. Gilani, A. Stefanescu, and M. A. Fernandez, "An empirical study of the state of the practice and acceptance of model-driven engineering in four industrial cases," *Empirical Softw. Eng.*, vol. 18, no. 1, pp. 89–116, 2013.
[37] G.-H. Lee and Y.-G. Kim, "Implementing a client/server system in Korean organizations: Interrelated IT innovation perspective," *IEEE Trans. Eng. Manag.*, vol. 45, no. 3, pp. 287–295, Aug. 1998.
[38] I. Cho and Y. Kim, "Critical factors for assimilation of object-oriented programming languages," *J. Manag. Inf. Syst.*, vol. 18, no. 3, pp. 125–156, 2001.
[39] A. Rai, "External information source and channel effectiveness and the diffusion of CASE innovations: An empirical study," *Eur. J. Inf. Syst.*, vol. 4, no. 2, pp. 93–102, 1995.
[40] S. Walderhaug, M. Mikalsen, I. Benc, and S. Erlend, "Factors affecting developer's use of MDSD in the healthcare domain: Evaluation from the MPOWER project," in *Proc. Workshop Eur. Conf. Model-Driven Archit.*, 2008.
[41] V. Venkatesh, J. Thong, and X. Xu, "Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology," *MIS Quart.*, vol. 36, no. 1, pp. 157–178, 2012.
[42] Y. Cheng, "Antecedents and consequences of e-learning acceptance," *Inf. Syst. J.*, vol. 21, no. 3, pp. 269–299, 2011.
[43] H. Xu, L.-B. Oh, and H.-H. Teo, "Perceived effectiveness of text vs. multimedia location-based advertising messaging," *Int. J. Mobile Commun.*, vol. 7, no. 2, pp. 154–177, 2009.
[44] T. Herath and H. R. Rao, "Encouraging information security behaviors in organizations: Role of penalties, pressures and perceived effectiveness," *Decision Support Syst.*, vol. 47, no. 2, pp. 154–165, 2009.
[45] J. C. Roca, C.-M. Chiu, and F. J. Martínez, "Understanding e-learning continuance intention: An extension of the technology acceptance model," *Int. J. Hum. Comput. Stud.*, vol. 64, no. 8, pp. 683–696, 2006.
[46] H. Kearns and M. Gardiner, "Is it time well spent? The relationship between time management behaviours, perceived effectiveness and work-related morale and distress in a University context," *High. Educ. Res. Develop.*, vol. 26, no. 2, pp. 235–247, 2007.
[47] M. Al-Emran, V. Mezhuyev, and A. Kamaludin, "PLS-SEM in information systems research: A comprehensive methodological reference," in *Proc. 4th Int. Conf. Adv. Intell. Syst. Inform. (AISI)*, 2018, pp. 644–653.
[48] C. M. Ringle, S. Wende, and J.-M. Becker, "SmartPLS 3. Bönningstedt: SmartPLS," Tech. Rep., 2015.
[49] J. F. Hair, G. T. M. Hult, C. Ringle, and M. Sarstedt, *A Primer On Partial Least Squares Structural Equation Modeling (PLS-SEM)*. Newbury Park, CA, USA: Sage, 2014.
[50] J. Henseler, C. M. Ringle, and M. Sarstedt, "A new criterion for assessing discriminant validity in variance-based structural equation modeling," *J. Acad. Marketing Sci.*, vol. 43, no. 1, pp. 115–135, 2015.

VITALIY MEZHUYEV received the B.S. and M.S. degrees in physics and informatics from Berdyansk State Pedagogical University (BSPU), Ukraine, in 1997, the Ph.D. degree in physics instruction from Kiev National Pedagogical University in 2002, and the Sc.D. degree in information technologies from Odessa National Technical University, Ukraine, in 2012. From 2004 to 2014, he was the Head of the Department of Informatics and Software Engineering, BSPU, Ukraine. He is currently a Professor at the Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, and the Head of the Software Engineering Research Group. During his career, he participated in the multiple international scientific and industrial projects, devoted to the formal modeling, design, and the development of advanced software systems as a network-centric real-time operating system; IDEs for the automation of development of parallel real-time applications; tools for specification, verification and validation of software products; visual environment for metamaterials modeling and others. His current research interests include formal methods, metamodeling, safety modeling, and verification of hybrid software systems, and the design of cyber-physical systems.

MOSTAFA AL-EMRAN received the B.Sc. degree with honors in computer science from Al Buraimi University College and the M.Sc. degree in informatics from The British University in Dubai with a distinction level along with the top Academic Excellence Award. He is currently pursuing the Ph.D. degree in computer science at Universiti Malaysia Pahang. He has published over 35 research papers in highly reputed journals such as *Computers in Human Behavior*, *Computers & Education*, *International Journal of Information Management*, and *International Journal of Engineering Education*, among many others. Most of his publications were indexed under ISI Web of Science and Scopus. He is a certified recognized reviewer by several leading journals in Elsevier. His current research interests include mobile learning, knowledge management, technology adoption, and text mining.

MURNI FATEHAH received the B.Sc. and M.Sc. degrees in software engineering from Universiti Malaysia Pahang, where she is currently pursuing the Ph.D. degree in computer science. Her current research interests include software engineering, metamodeling, and technology adoption.

NG CHIN HONG received the B.Sc. degree in software engineering from Universiti Malaysia Pahang. His current research interests include metamodeling and technology adoption.