Proceedings of the 3rd International Conference on Computing and Informatics, ICOCI 2011,8-9 June, 2011 Bandung, Indonesia | *Paper No. 100*

VALIDATION OF MGBL ENGINEERING MODEL USING GROUP TREATMENT EXPERIMENTAL STUDY

Syamsul Bahrin Zaibon[1] and Norshuhada Shiratuddin[2]

[1]*Universiti Utara Malaysia, Malaysia, syamsulbahrin@uum.edu.my*
[2]*Universiti Utara Malaysia, Malaysia, shuhada@uum.edu.my*

ABSTRACT. A mGBL engineering model is proposed intentionally for developing mGBL applications and is outlined in this paper to provide novice developers with an integrated model with which they can approach more systematically the design and development of mGBL. The engineering model combines a game life cycle based on iterative prototyping and learning model, with supporting activities drawn from sources of best practice in mobile game development. This paper describes an experimental study involving the implementation of the proposed model with a group of undergraduate students who are taking Game Application Development course. The results indicate that the proposed model was practical and workable in developing mGBL applications compared to other models.

Keywords: mGBL, experimental study, group treatment, mobile game based learning, engineering model.

INTRODUCTION

Mobile game-based learning (mGBL) can be defined as a game specifically utilized for learning which is played on mobile devices such as mobile phones. The main objective of mGBL is to use game play to enhance motivation in learning, engage in knowledge acquisition, and improve effectiveness of learning activities through mobile environment. Furthermore, mobile environment offers learning in a natural setting, everywhere, and anytime. For those prospective reasons, the key challenges for effective learning with mGBL are for the learners to be interested, motivated, engaged, and mobility accessed.

There is a global trend to incorporate mGBL into learning environment to increase the efficiency, cost effectiveness and quality of learning. However, the literature still lacks of the mGBL design and development guidelines. Also, due to the different nature of mGBL, the issues to further explore the design and development of mGBL to help developers make the learning context more valuable are needed. Therefore, a mGBL engineering model is proposed (as described in the next section) which incorporates learning models and structured processes which aims to provide the steps and stages on mGBL design and development. In validating the proposed model, a group treatment experimental study was conducted by comparing to other models. The result from the study will test the hypothesis whether the proposed model is applicable.

THE PROPOSED MGBL ENGINEERING MODEL

The proposed mGBL engineering model comprises phases, components, activities, and deliverables. This model is proposed intentionally for guiding developers to develop mGBL applications. A better mGBL application delivery is also expected by implementing the proposed model. It is divided into two layers, where the first inner layer is called as general phases; pre-production, production and post-production. In the second layer, there are components should be included for each respected phase as illustrated in Figure 1. The three phases are executed in a sequential manner which starting from pre-production phase followed by production phase and then post-production (the clockwise-direction arrows represent the flows of the phases). After completing the first phase, all designs are sent for review before second phase is taken place. Any amendments are made and corrected after review. If the design are approved and signed off, the production phase is carried on next. The similar review activity also should be conducted after completing the production phase. All errors and inaccuracy of technical aspects of the game are rectified before it continues to the final phase.

The engineering model also includes components which are numbered from 1 to 12. These components are flexible and iterative, which can be customized based upon developer preferences. These components are mapped to the AI four stages: i.e. discover, dream, design and delivery. In addition, the mGBL engineering model suggests the expanded guidelines by providing specific objectives, activities, and deliverables for each component.

Discover
1
2
3
12
Delivery
Dream
4
11
10
5
9
Design
6
8
7

In pre-production phase, four components are identified which are essential to be considered at the initial stage of mGBL development, namely Requirement Analysis and Planning; Mobile Interaction and Technical Analysis; Learning Content Design; and Game Features Design. This phase is initially about discovering the target audience, conceptualizing of idea, designing interaction, specifying learning domain and creating storyboard. At this stage, creating the mGBL concept is vital activities which will be referred to. All of the theories should be embedded in the mGBL learning content design in order to achieve the learning objective. The next phase is shifting to real development of the mGBL where it is coded and integrated with features as specified in the previous phase. Components should be included in this phase are Learning Content Development; Game Assets Development; Coding and Core Mechanics Development; and Game Features Integration. The most important component in this phase is the learning content development which focuses on the learning concept and contents. The learning contents development should be referred from the content experts. Finally, at the final phase, the core activity is the testing procedure to ensure its quality before releasing to the market. Game Porting and Deployment; Playability, Usability and Mobility Testing; Educational Testing; and Distribution are the main components in this phase. Deployment step is essential at this stage to cater problems of running on different platforms of mobile devices. The platforms vary in different types and categories such as Symbian, Windows Mobile, Java, and Palm OS. In addition, this engineering model also provides flow of documents and deliverables to be referred to for documentation and references (Zaibon & Shiratuddin, 2010). In validating the proposed model, a group treatment experimental study was set up as described in the next section.

MODEL VALIDATION USING GROUP TREATMENT EXPERIMENTAL STUDY

A group treatment experimental study was conducted involving the implementation of the proposed model with a group of undergraduate students who are taking Game Application Development course at Universiti Utara Malaysia. 70 students participated in this study and they were divided into four groups for comparison as illustrated in Table 1. Group A, B, and C were allocated as the control groups while group D was the experimental group. They were required to use the given model as a basis for designing and developing mGBL applications for their final project. This study was carried out concurrently for all groups and took a whole semester. Each group was given detail descriptions of their model and both course instructor and researcher helped students in terms of the technical aspects for developing mGBL.

Table 4. Experimental and control groups

Group	N	Types of Development Model
A (Control)	18	Analysis, Design, Development, Implementation, Evaluation (ADDIE)
B (Control)	20	Input-Process-Output (IPO)
C (Control)	14	Game Life Cycle (GLC)
D (Experimental)	18	mGBL engineering model (mGBL)
Total	**70**	

Table 2. Evaluation Dimensions

	Dimensions	Descriptions
1.	Visibility	The model is visible to the game developers, so that the developers can judge the relevance and completeness of the game development.
2.	Complexity	Complexity is the degree to which a model is perceived as being difficult to use. The more complex of the model, the more difficult to use. Learning about the model should be easy, clear and understandable.
3.	Compatibility	Compatibility refers to the degree to which a model is perceived as being consistent with the existing values, needs, and past experiences of game developers.
4.	Flexibility	The model provides flexible development process with minimal planning. The model is also adaptive and responsive to changing user needs. The model should be flexible and adaptable for future use.
5.	Clarity	The model as a whole is workable. The phases in the model are easily followed and steps or activities included in the model are easy to apply. The model also provides specific guide to mGBL development.
6.	Effectiveness	The model is perceived as being better than its precursor. By using the model, it will increase productivity, effectiveness and quality of mGBL development.
7.	Manageability	The processes and activities in the model to be capable of being managed or controlled. In general, the model also provides project management.
8.	Evolutionary	The model provides the dynamic process which evolves through continuous feedback from users. The model is capable of incremental change, to cope with new ideas or technological opportunities. The model provides developers to communicate and collaborate with users continuously to incorporate their evolving requirements.

In validating the proposed model, some evaluation dimensions were studied which can be used for the model assessment. A number of evaluation dimensions have been proposed by researchers to evaluate models and methodologies which come from different fields such as general software development, multimedia applications, and project management. These are from Veryard (1985), Platts (1990), Henderson-Sellers (1995), Lang and Barry (2001), Riemenschneider (2002), Yu and Cysneiros (2002), Ciconte (2003), Hecksel (2004), Bonner (2008), and Kerzner (2006). These dimensions were composed as illustrated in Table 2. Table 3 presents the mean of all models based on the 8 dimensions denoted by students. It shows that mGBL engineering model scored mean above 7.0 (out of 10) of all dimensions compared to other models. This suggested that the mGBL engineering model is highly accepted by the experimental groups.

Table 5. Means and standard deviations for four models and eight variables

	ADDIE		IPO		GLC		mGBL	
Dimension/ Variable	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**
Visibility	6.722	1.300	6.483	1.374	6.714	1.563	7.833	1.195
Complexity	6.300	1.224	6.320	1.640	6.886	1.836	7.022	1.768
Compatibility	6.611	0.981	6.720	1.245	6.471	1.599	7.467	1.552
Flexibility	6.847	1.269	6.488	0.985	6.607	1.675	7.750	0.928
Clarity	7.236	1.044	6.469	1.173	6.277	2.091	8.035	1.317
Effectiveness	7.011	1.103	6.640	1.203	6.271	1.746	7.922	1.336
Manageability	6.792	1.412	6.675	1.095	6.589	1.890	7.917	1.406
Evolutionary	7.233	1.152	6.580	1.131	6.357	1.681	8.222	1.127

In order to ensure there are significant different between all groups, one way ANOVA was run eight times for each dimension. The results show that there are significant differences ($p < .05$) between all groups in term of Visibility with $F (3, 66) = 3.666$, $p = .017$; Flexibility with $F (3, 66) = 3.996$, $p = .011$; and Manageability with $F (3, 66) = 3.278$, $p = 0.26$. For dimension Clarity and Effectiveness there are very significant differences between all groups with $F (3, 66) = 5.571$, $p = .002$ and $F (3, 66) = 4.717$, $p = .005$ respectively. The result also indicates that the Evolutionary dimension is highly significantly different of all groups with $F (3, 66) = 7.543$, $p = .000$. However, two dimensions are not significantly different ($p >.5$): Complexity $F (3, 66) = 0.956$, $p = .419$ and Compatibility $F (3, 66) = 1.869$, $p = .143$. The reasons could be due to that students felt the models were complex with many steps or activities to be followed, therefore not well-suited to them as novice developers. In order to detect differences among groups, a multiple comparison test using Tukey Least Significant Difference (LSD) is utilized. The test can be used to determine whether a significant mean difference exists between each pair of groups (Table 4).

Table 6. Multiple comparisons between models using Tukey LSD

Types of Model		Mean Different (I – J) for Each Dimension							
(I)	(J)	Visibility	Complexity	Compatibility	Flexibility	Clarity	Effectiveness	Manageability	Evolutionary
mGBL	ADDIE	1.111* p= 0.016	0.722 p= 0.186	0.856 p= 0.061	0.903* p= 0.029	0.799 p= 0.094	0.911* p= 0.045	1.125* p= 0.022	0.989* p= 0.022
	IPO	1.350* p= 0.003	0.702 p= 0.187	0.747 p= 0.093	1.263* p= 0.002	1.566* p= 0.001	1.282* p= 0.004	1.242* p= 0.010	1.642* p= 0.000
	GLC	1.119* p= 0.023	0.137 p= 0.814	0.995* p= 0.042	1.143* p= 0.010	1.758* p= 0.001	1.651* p= 0.001	1.327* p= 0.012	1.865* p= 0.000

*The mean difference is significant at the .05 level.

In Visibility dimension, comparing mGBL to ADDIE, IPO, and GLC, mGBL is seen more visible with the mean difference in visibility is large ($M > 1$). The Sig. values of ADDIE ($p = 0.016$), IPO ($p = 0.003$), and GLC ($p = 0.023$) show that this is statistically significant. However in term of Complexity, the mean differences between mGBL to ADDIE, IPO and

GLC are relatively small (M < 0.7) and non-significant ADDIE (p = 0.186), IPO (p = 0.187), GLC (p = 0.814) even though mGBL is less complex than the three models.

In Compatibility dimension, although mGBL scored higher than ADDIE and IPO, the mean difference in compatibility is relatively small (M < 0.9) and the Sig. values (p > .05) shows that this is statistically not significant. In contrast, comparing mGBL to GLC, although the mean difference in compatibility is less than 1 (M = 0.995), the Sig. value (p = 0.042) shows that this is statistically significant. mGBL also gained more score compared to ADDIE, IPO and GLC in Flexibility and statistically significant. To compare mGBL to ADDIE, the mean difference in Clarity is small (M = 0.799) even though the positive sign indicates that mGBL is clearer than ADDIE. The Sig. value (p = 0.094) shows that this is not significant. However, in comparing mGBL to IPO and GLC, the mean difference in clarity is large and the Sig. value (p < .05) shows that this significant. The other three dimensions (Effectiveness, Manageability, Evolutionary) mGBL also have higher scores compared to ADDIE, IPO, and GLC. The mean difference in effectiveness is quite big and the Sig. value (p < .05) shows that this is statistically significant. The positive sign also indicates that mGBL is more effective than GLC. Significantly higher mean scores in visibility, flexibility, clarity, effectiveness, manageability, and evolutionary exhibited by students for mGBL engineering model, indicated that they understand how to implement the model as guideline for their mobile game development project. They also have completed their project in a systematic manner without having difficulties in finding the game requirements especially related to mobile game for learning. Nevertheless some minor issues with the mGBL engineering model did become noticeable where a few students found some of them confusing. These issues are related in particular to some aspects of technical testing for their project. In answering the main research question, a hypothesis testing was conducted. The hypothesis null is as follows:

H_0: The proposed mGBL engineering model is not significantly applicable.

Based on results of the experimental study, it can be summarized that the 8 evaluation dimensions can be defined as a single term which is applicability. Therefore one-way ANOVA test was run another round (included all 8 dimensions) that test the applicability of the proposed model as indicated in Table 5. In comparison of the applicability of the proposed model with other models, the results show significant values p = .007 (p < .05) with the value of F (3, 66) = 4.341. Here, the result shows that the null hypothesis would not be accepted. Therefore, it can be concluded that the proposed mGBL engineering model is significantly applicable in designing and developing mGBL.

Table 5. One Way Analyses of Variance for Four Models on Applicability

Attributes	df	SS	Mean Square	F(3,66)	Sig.
Applicability					
Between Groups	3	25.329	6.045	4.341	.007 *
Within Groups	66	118.145	1.392		

*significant level at .05

CONCLUSION

This study has proposed the mGBL engineering model that can be helpful for the developers to follow through for developing mGBL applications which make the mGBL more effective for learning environment. This study also validated the proposed model based on 8 dimensions, namely: visibility, complexity, compatibility, flexibility, clarity, effectiveness, manageability, and evolutionary. In the group treatment experimental study, the results indicated that six dimensions (visibility, flexibility, clarity, effectiveness, manageability, and evolutionary) were significantly different to all models, however two dimensions (complexity and compatibility) were not significantly different. These results showed that the mGBL engineering model scored a high overall mean. Hypothesis testing was also conducted and the result shows that the null hypothesis rejected. This implied that the proposed model could be implemented by potential developers to develop mGBL applications. A number of future considerations can be suggested for this study, for example the experimental study should be extended to other group of students and to developers in commercial and industry environments. Validation, verification and testing along the whole process of the proposed mGBL engineering model also required to be focused on. Such findings can perhaps provide richer information and more discussions. In addition, the evaluation session can be conducted to other mGBL applications.

REFERENCES

Bonner, N. (2008). *Acceptance of systems development methodologies: Testing a theoretically integrated model*. PhD Thesis. The University of Texas, Arlington.

Ciconte, B., Devgan, M., Dunbar, S., Go, P. & Prem, J. (2003). J2EE Software Development Methodologies. InformIT. Retrieved January 15, 2010 from http://www.informit.com/articles/printerfriendly.aspx?p=102017

Hecksel, D. (2004). *Methodology Evaluation and Selection*. White Paper. Sun Microsystems. Retrieved January 15, 2010 from http://www.davidhecksel.com/projectcontext/whitepaper.html

Henderson-Sellers, B. (1995). Who needs an object-oriented methodology anyway? *Journal of Object Oriented Programming, 8*(6), 6-8.

Kerzner, H. (2006). *Project management: best practices: achieving global excellence*. Wiley.

Lang, M. & Barry, C. (2001). *Techniques and Methodologies for Multimedia Systems Development: A Survey of Industrial Practice*. In Russo, N. L. et al. (eds), Realigning Research and Practice in Information Systems Development, Proceedings of IFIP WG 8.2 Conference, Boise, Idaho, USA, July 2001, pp. 77-86. Boston: Kluwer

Naismith, L., Lonsdale, P., Vavoula, G., & Sharples, M. (2006). *Report 11: Literature review in mobile technologies and learning*. Retrieved August 20, 2008 from http://www.futurelab.org.uk/resources/documents/lit_reviews/Mobile_Review.pdf

Pachler, N. (Ed.). (2007). *Mobile learning: towards a research agenda*. WLE Centre Occasional Papers in Work-based Learning 1. London: Institute of Education.

Platts, K. (1990). *Manufacturing Audit in the Process Strategy Formulation*. PhD dissertation, University of Cambridge, United Kingdom.

Riemenschneider, C. K., Hardgrave, B. C. & Davis F. D. (2002). Explaining Software Developer Acceptance of Methodologies: A comparison of Five Theoretical Models, *IEEE Transactions on Software Engineering, 28*(12), 1135-1145.

Zaibon, S. B. & Shiratuddin, N. (2010). *Mobile Game-Based Learning (mGBL) Engineering Model as a Systematic Development Approach*. In Z. Abas et al. (Eds.), Proceedings of Global Learn Asia Pacific 2010. pp. 1862-1871. AACE.

Veryard, R. (1985).*What Are Methodologies Good For Data Processing, 27*(6), 9-12.

Yu, E. & Cysneiros, L. M. (2002). *Agent-Oriented Methodologies-Towards a Challenge Exemplar*. Proceedings of the 4th International Workshop on Agent-Oriented Information Systems (AOIS'02). Toronto.