

Integration Issues on User Interface based Web Development

Gürkan Alpaslan ¹, Oya Kalıpsız ²

^{1,2} Department of Computer Engineering, Yildiz Technical University, Istanbul, Turkey

Email: ¹gurkana@yildiz.edu.tr, ²oya@ce.yildiz.edu.tr

Abstract—User interface prototypes are effective structure for describing as models for web application development thanks to their graphical and dynamic architecture. It provides the benefits such as language independent development, less error-prone and increased quality. The executable programming codes are created by code transformation feature of model-driven development tools as well. Nevertheless, this technique brings the problems along all its benefits. First of all, code transformation from user interfaces by development tools, creates complicated structure to analyze and modify. Another challenge is to deal with the processes which are triggering each other. In addition, data connected process adaptation is a problem as well. All these issues eventually require the integration of discrete modules. This study provides not only a proposal structure to deal with integration problem in web development, but also provides the evaluation criteria for classification of the integration processes.

Keywords—web application development, model-driven development, model integration

I. INTRODUCTION

In web development, user interface prototypes [1] can be used as web models which are called mockups or wireframes [2], [3]. The user interface prototypes can be transformed the generated web codes automatically, easily and reliably by code transformation feature of model development tools. Mockup development tools provide dynamic user interface elements, so, interactive web pages are created and transformed the executable codes. Nevertheless, the tools have limited capability [4], [5], therefore, the generated codes have to be assembled with the additional codes. So, a critical issue is emerged which is called integration.

In web applications, generated codes by model tools are so long in the result of transformation algorithms of tools. Model to code generation feature runs on Hypertext Markup Language (HTML) based structure. Code generation algorithms are mostly designed to describe all the static or dynamic features inside the HTML tags rather than distributing them into separate files. Graphical and interactivity commands are kept in a single file. This structure creates overmuch code in a single file which is hard to analyze. Therefore, analyzing the code to modify for integrating is a challenge for developers. This study aims to find a way to ease this challenge with looking at from developer's perspective.

Another issue is database connection with the mockup elements. We can imply that mockup tools do not provide a comprehensive database connection features. This study

provides the suggestions for implementing the relational database management systems or other data supporters with the mockups.

In the literature, integration on the model based development is studied as to incorporate the non-functional requirements into models [6], [7]. However, in this study, integration is considered not only for non-functional requirements, but also functional requirements.

The rest of the paper organized as follows: model-driven web development is explained in Section II. The proposed approach is described in Section III. After that, Section IV discuss the approach and finally, Section V concludes the paper.

II. MODEL-DRIVEN WEB DEVELOPMENT

Model-driven development approach (MDD) is based on developing the software on a higher abstraction level [8], [9]. In MDD, every process is described by multi-purposes models which are independent from programming structures. For web applications, programming structures are both server-side and client-side scripting languages as well as markup languages. MDD provides a life cycle beginning from Computational Independent Model, transformed to Platform Independent Model, and eventually serves a Platform Specific Model [10], [11]. The abstraction level decreases as the result of transformation to Platform Specific Model.

In web development based on models, there are few different techniques such as Unified Modeling Language (UML) based, Object-oriented based, Web modelling languages based and more, yet, key difference of them is the element which is used as model in the technique [12]. In this study, user interfaces prototypes are used for model description element. Hence, web domain model is defined as possible as simple.

III. THE PROPOSED APPROACH

A. Overview

The approach is based on priority analysis and vector representation of the integration processes for ordering them. Integration queue is designed, thus, not only processes proceed from more important ones to less, also correlated activities are considered as a group for avoiding the conflicts of the processes. This part is implemented in business process analyst

section which are the most time effort activity in our approach. Figure 1 is illustrated the software development life cycle.

The method proceeds iterative and incremental way, so it is an agile development method [13] supported life cycle. To combine the agile development method and model-driven method is a popular approach due to its flexible architecture [14], [15]. The life cycle begins with requirement analyst process and proceed with dividing the efforts into two parts as model coding and self coding activities. In the requirement analyst part, all requirements are matched with the user interface elements [16], [17]. The models are transforms to executable codes and integrates with the self codes orderly according to priority queue.

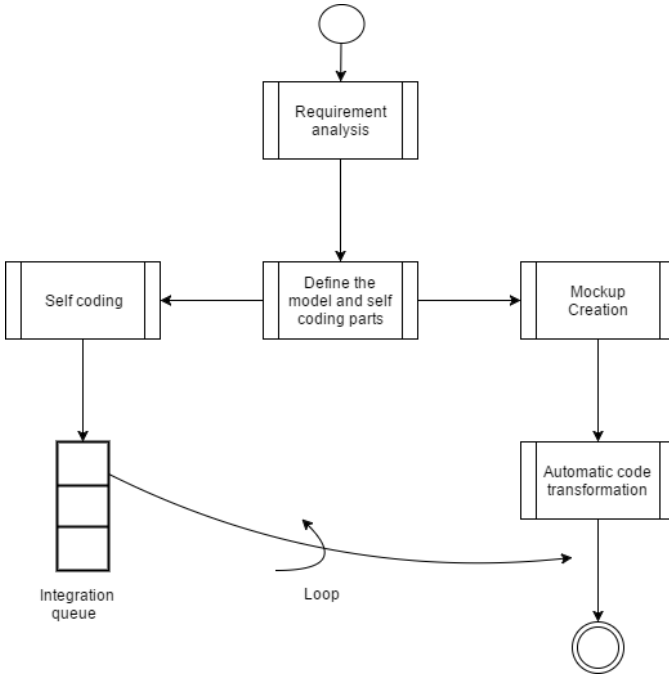


Fig. 1. Development life cycle

B. Integration queue

The integration parts should be placed into the iteration queue depending on their effects and scope. Additionally, coherence with the other parts should be considered. Therefore, the questions below must be answered:

- 1) Is the process a functional requirement or non-functional requirement?
- 2) Is the process an external data-connected activity or self activity?
- 3) How much area is affected by the integration?
- 4) Does the activity trigger any other?

Thus, by considering these questions, integration coefficient value is estimated and priority line is determined. Coefficient values are used for determining the complexity of the system as well.

- Is the process a functional requirement or non-functional requirement?

Generally, in MDD, all the functional requirements aim to be implemented via models. In our proposed approach, a parts of functional requirements are generated by hand-crafted codes, since, the efficient and fast development and as well as the results of tool constraints. Thus, it gains importance whether the generated ones have described as functional or non-functional in the business model.

- Is the process an external data-connected activity or self activity?

Activities which require a database connection or providing the data an external file have to be considered different than the self activities.

- How much area is affected by this integration?

Even they are being developed separately and iteratively, it is not possible to consider the units of the system independently. Therefore, almost every integration process affects one or more system units. In addition, there are the dependent processes which can not run unless prerequisite process is done. So, dependency matrix is proposed to track the process dependencies.

$$\vec{\gamma} = [x_1 \ x_2 \ \dots \ x_n]^T \quad (1)$$

where γ represents the processes while x stands for the integrations which is real numbers $x \in [0, 1]$. The value of x indicates the dependency degree between intended integration process and existing process.

- $x = 0$ means, there is no correlation
- $x = 1$ means the full correlation
- if x value is between the 0 and 1, number size represents the rating of dependency.
- Does the activity trigger any others?

The web pages which are triggered by another page can be connected to its parent source. Thus, they are considered as a group processes and run together. This activity prevents the run-time errors.

C. The Queue Ordering Algorithm

All process are transformed vector form and added into queue according to vector magnitude measure. They are firstly classified as whether functional or non-functional requirement, and data-connected or self process. After that, dependency matrix of classified processes are determined. Thus, every process can be defined with 3-tuple notation as:

$$P = (\alpha, \beta, \gamma) \quad (2)$$

where α denotes whether the process is functional requirement or not.

$$\alpha = \begin{cases} 1 & \text{process is functional} \\ 0 & \text{process is non-functional} \end{cases}$$

β denotes whether the process is data-connected or not.

$$\beta = \begin{cases} 1 & \text{process is data-connected} \\ 0 & \text{process is self process} \end{cases}$$

γ denotes the dependency matrix, identified before. Therefore, every process is defined as a vector

$$\vec{P} = [k_1\alpha \ k_2\beta]^T \odot \vec{\gamma}^T, \quad (3)$$

where \odot denotes concatenation in vertical dimensions and k is the coefficient value determines the importance evaluation particular for the developing software.

$$\vec{P} = [k_1\alpha \ k_2\beta \ x_1 \ x_2 \ \dots \ x_n]^T \quad (4)$$

Thus, process priority is determined by the vector magnitude measure, as well as, by the cosine similarity measure, similarity of the processes are calculated.

$$\|\vec{P}\| = \sqrt{k_1^2\alpha^2 + k_2^2\beta^2 + \sum_{i=1}^n x_i^2} \quad (5)$$

For two integration process P_1 and P_2 , cohesion of each other can be measured as:

$$Sim(\vec{P}_1, \vec{P}_2) = \frac{P_1 \cdot P_2}{\|\vec{P}_1\| \cdot \|\vec{P}_2\|} \quad (6)$$

Hence, high cohesion processes are grouped and considered as a single process and run sequentially.

D. Database connection with mockup generated codes

For the purpose of the database connection, utilization the embed server-side markup languages are proposed such as Razor [18]. For effective implementation, the codes which are created by tools with code generation have segmented into portions, related everyone with a specific requirement. So, system requirements to mockup elements pairing is proposed. Figure 2 is illustrated the data connection procedure.

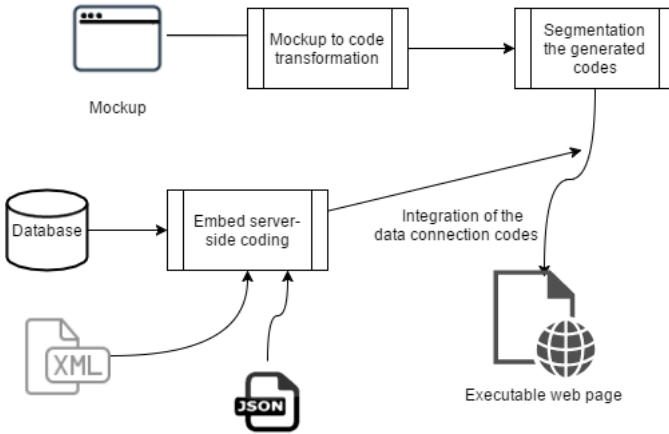


Fig. 2. Data connection procedure

Independently from which technique is used for providing data, such as relational database, JavaScript Object Notation (Json) ¹ or Extensible Markup Language (XML) ², data is processed by a convenient embed server-side coding method in order to utilize.

¹<http://www.json.org/>

²<https://www.w3schools.com/xml/>

IV. EVALUATION

A. In the terms of software development methodology

The process includes comprehensive analysis and design parts. After completion of these parts, it proceeds iteratively and incrementally. Most critical part is design part, due to the integration order is determined in that process. Thus, the whole process has the similarity features with both two main development methodologies: Waterfall and Agile [13], [14]. Process starts with like Waterfall methods, it includes time-consuming, strict analysis/design parts, but from the implementation to end, it proceeds with Agile methodology.

B. Change management

User interfaces are convenient structures to add, modify or remove. Till the automatic code transformation process, the modifications can easily be adapted. After the code transformation, it is more time-consuming process. To modify the integrated code directly can cause errors, therefore, it is more convenient to make the modifications on the model or non-integrated self code.

C. Web Application Security

The process utilizes embed server-side codes for database connection, thus, it is predictable structure. Predictable structures are open for SQL injection attacks, thus, SQL masking techniques are proposed.

D. Platform Flexibility

The approach is designed for web applications, yet, it is appropriate for adapting the other platforms with customizing it. Therefore, the criteria of the priority queue are identified according to adapted platform and the weights of them are determined.

E. Team Cohesion

The process starts with comprehensive business process by analyst team and after that proceed with parallel processes as model part and integrated code creation part. The processes are tight related with each other, thus, distinct team should work with high interaction.

F. Tool support

Even though, development mostly based on tool development, full adequate tools are so rare and provides limited features. It is an open area to develop a full customized tool for user interface based development.

V. CONCLUSION

User interface prototypes utilizing for web applications are recently popular area with its advantages. Generally, limited tool capacity, limited problem domain definition language are fundamental problems. A way to deal with limited features is to create the limited parts with self coding. In this study, it has been focused on integration issues between these type of distinct developed codes and provided an approach based on priority analysis.

In comparison with traditional method, the proposed approach in this study provides a baseline in order to sort and group the processes in life cycle. By this effort, all processes run as interconnected manner. Priority analysis criteria determined and integration queue ordering algorithm has been proposed based on defined measurements.

REFERENCES

- [1] T. G. Lewis, F. Handloser, S. Bose, and S. Yang, "Prototypes from standard user interface management systems," *Computer*, vol. 22, no. 5, pp. 51–60, 1989.
- [2] J. M. Rivero, J. Grigera, G. Rossi, E. R. Luna, F. Montero, and M. Gaedke, "Mockup-driven development: providing agile support for model-driven web engineering," *Information and Software Technology*, vol. 56, no. 6, pp. 670–687, 2014.
- [3] J. Zhang and J.-Y. Chung, "Mockup-driven fast-prototyping methodology for web application development," *Software Practice & Experience Journal*, 33 (13), 2003, pp., vol. 33, no. 13, 2003.
- [4] N. Koch, S. Meliá-Beigbider, N. Moreno-Vergara, V. Pelechano-Ferragud, F. Sánchez-Figueroa, and J. Vara-Mesa, "Model-driven web engineering," *Upgrade-Novática Journal (English and Spanish)*, Council of European Professional Informatics Societies (CEPIS) IX, vol. 2, pp. 40–45, 2008.
- [5] G. Rossi, Ó. Pastor, D. Schwabe, and L. Olsina, *Web engineering: modelling and implementing web applications*. Springer Science & Business Media, 2007.
- [6] K. Khatter and A. Kalia, "Integration of non-functional requirements in model-driven architecture," *Fifth International Conference on Advances in Recent Technologies in Communication and Computing*, 2013.
- [7] D. Ameller, X. Franch, and J. Cabot, "Dealing with non-functional requirements in model-driven development," in *Requirements Engineering Conference (RE)*, 2010 18th IEEE International. IEEE, 2010, pp. 189–198.
- [8] M. Völter, T. Stahl, J. Bettin, A. Haase, and S. Helsen, *Model-driven software development: technology, engineering, management*. John Wiley & Sons, 2013.
- [9] D. C. Schmidt, "Model-driven engineering," *COMPUTER-IEEE COMPUTER SOCIETY*, vol. 39, no. 2, p. 25, 2006.
- [10] J. L. H. Agustin, "Model-driven web applications," in *Science and Information Conference (SAI)*, 2015. IEEE, 2015, pp. 954–964.
- [11] W. Schwinger, W. Retschitzegger, A. Schauerhuber, G. Kappel, M. Wimmer, B. Pröll, C. Cachero Castro, S. Casteleyn, O. De Troyer, P. Fraternali *et al.*, "A survey on web modeling approaches for ubiquitous web applications," *International Journal of Web Information Systems*, vol. 4, no. 3, pp. 234–305, 2008.
- [12] J. A. Hincapié Londoño and J. F. Duitama, "Model-driven web engineering methods: a literature review," *Revista Facultad de Ingeniería Universidad de Antioquia*, no. 63, pp. 69–81, 2012.
- [13] R. C. Martin, *Agile software development: principles, patterns, and practices*. Prentice Hall PTR, 2003.
- [14] S. W. Ambler, *The object primer: Agile model-driven development with UML 2.0*. Cambridge University Press, 2004.
- [15] G. Alpaslan and O. Kalıpsız, "Model driven web application development with agile practices," *International Journal of Software Engineering & Applications*, Vol.7, No.5, 2016.
- [16] A. Homrighausen, H.-W. Six, and M. Winter, "Round-trip prototyping based on integrated functional and user interface requirements specifications," *Requirements Engineering*, vol. 7, no. 1, pp. 34–45, 2002.
- [17] N. Koch, A. Kraus, C. Cachero, and S. Meliá, "Integration of business processes in web application models," *J. Web Eng.*, vol. 3, no. 1, pp. 22–49, 2004.
- [18] P. Gupta, R. Mata-Toledo, and M. Monger, "Utilizing asp. net mvc in web development courses," *Journal of Computing Sciences in Colleges*, vol. 27, no. 3, pp. 10–14, 2012.