

METAMODELS EVALUATION OF WEB ENGINEERING METHODOLOGIES TO DEVELOP WEB APPLICATIONS

Karzan Wakil¹ and Dayang N.A. Jawawi²

¹ Fine Arts Institute in Sulaimaniyah-Iraq

²Software Engineering Department, Faculty of Computing, UTM-Malaysia

ABSTRACT

The purpose of a metamodel in web engineering methodologies is for the platform independent analysis as well as the design of the content, navigation, and presentation issues of web applications. In the previous years, numbers of methodologies for the development of web applications were proposed, and most of them defined their notation for creating metamodels. The increasing expansion and complexity of web applications are a new challenge for web software developers. This paper presents a comparison study between metamodel of the three methodologies which are; UML-Based Web Engineering (UWE), Web Modeling Language (WebML) and Object Oriented Hypermedia (OOH). The aim is to show the capability of the methodologies to address the challenges in developing the web applications. The evaluation results presented in this paper help the designer in providing initial knowledge of the strengths and weaknesses of the three methodologies for developing web applications.

KEYWORDS

Web Engineering Methodologies, Metamodel, UWE, WebML, OOH, Comparison, Evaluation.

1. INTRODUCTION

Web Engineering is rich in design methodologies such as OOH, OOHDM, UWE, W2000, WebML and so on, for supporting and developing the complex task of designing web applications. These methodologies propose the construction of different models, which comprises at least conceptual model, navigation model and presentation model. Each model consists of a set of modeling elements, for example; nodes and links for the navigation model or image and anchor for the presentation model. In addition, all these methodologies define or choose a notation for the constructs that they define [1].

Web Engineering Methodologies for improvement of web applications, use various notations and propose a little various development processes, it used a common metamodel as based methodologies for the web domain. Also metamodel is a best definition of the models of modeling. The relation between metamodels and their elements, with well-formedness rules, are requirements for creating a semantic web. Web engineering methods based on this common metamodels can only use part of the build provided by the metamodel. The common metamodels must be the unification of the modeling structure of popular web engineering methods permitting for their better comparison and integration [2].

Comparison is a reason to find weaknesses and strengths of the methods. We will compare elements of the metamodels between UWE, WebML and OOH, and then make an evaluation for the comparison, for finding the best methodology among them to developing web applications.

This paper is organized as follows: the section two shows the current works. The section three explains metamodels of web engineering methodologies (UWE, WebML and OOH). The section four compares the three methodologies and evaluates the comparison between UWE, WebML and OOH. The section five shows the design of the case study and implementation. The last section consists of the conclusion of the comparison, evaluation, and the future work.

2. RELATED WORK

In the research of web methods, there are relatively limited examples of it in the literature of studies [3-7]. Functioning as foundation for process enhancement and better product superiority, the majority of investigation in Web engineering has focused on the suggestion of methodologies and tools. An examination where certain characteristics of a web technique were measured as explained by Lee et al. [4]. Key modeling methods, phases, documentation, source of navigation, method to recognize users' opinion and semantic quality were the features utilized. However, the legitimacy of the outcomes may have been prejudiced because a majority of the outcomes that were demarcated by the authors are subjective.

The type of differentiation that Koch [8] matches the phases that are gone through by certain web methods may conceal certain essential facts. There is a clear variation in the penetration in which a method describes a phase. For instance, certain Web techniques only suggest a set of textual procedures for some growth stages while others furnish supporting tools for the similar stages.

Montero et al. [6] prepare a framework of requirements, to study web engineering methodologies for system's hypermedia. Some requirements are gotten from the fields of hypermedia engineering and software engineering. These requirements are used to design methodologies for identify their advantages and disadvantages.

To signify the features of web applications, and where the current techniques fall short in abiding by these requirements, Gu et. al [5] searches the requirements of web methods. In some key regions, current restrictions seem to be most essential, for instance, the connection between information and operational features want of support in its dynamic modeling. Gu et al.'s evaluation can be considered as less subjective and imprecise than Lee et al.'s, Montero et al.'s and Koch's work, although only a few web methods were presented, and some extensions were not taken into account.

Baresi and Morasca [7] carried on three empirical studies focusing the effort needed on designing web applications. The empirical studies were based on the use of W2000, but the hypotheses and results may apply to other methods.

By differentiating between web engineering methodologies in previous works, there were drawbacks and strengths. But certain investigators endeavoured to enhance web engineering techniques and metamodels, however until now is not ideal yet for the development of web application. In this paper, we will try to find the best method among UWE, WebML, and OOH to develop the web applications.

3. WEB ENGINEERING METHODS

Various web modelling methods have emerged in the past in the field of web engineering, which includes W2000, WebML, WSDM, OOH, UWE, OOWS, and OOHD. Each of these methods used for the development of web pages [8]. In this paper we use three famous methods that are consisting of UWE, WebML and OOH.

3.1. UML-Based Web Engineering (UWE) Method

UWE came up by 1998. The method was developed by the Web Engineering Group from the Ludwig-Maximilians-Universität München [9-10]. It is a software engineering approach based on UML [11]. It uses the UML standard notation as much as possible and defines a UML profile to specify the peculiarities that introduce web applications. The major benefit of being UML compliance is that any, CASE tool that supports the UML notation can be used to produce the UWE models [12].

3.1.1. UWE Metamodel

UWE metamodel is a design, which considered as the conservative extension of the UML metamodel, in other words, can say that the modelling elements of the UML metamodel are inherited from the UML metamodel, and they are not modified by adding new features or additions to the modelling element's class. The UWE metamodel can be customized on the basis of a profile by mapping it to a UML profile. UWE metamodel for web applications can be created by using generic UML case tools and UML profiles or their extension, objects those are tagged and OCL restrictions [13-14], Figure 1 shows UWE metamodels.

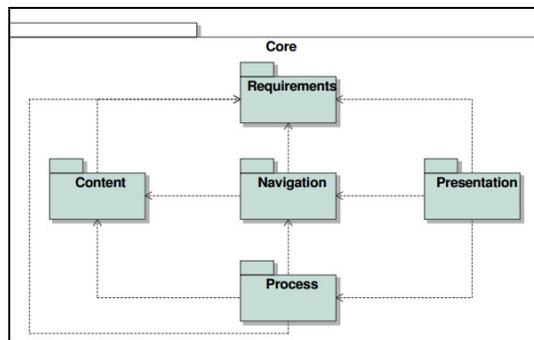


Figure 1. UWE Metamodel [1]

A. Conceptual Modelling

The conceptual design intends to generate a conceptual model which ignores aspects of interaction, presentation and routing paths as much as possible. These features are deferred to the steps of the course-plotting and arrangement modelling. The important UML modelling elements that used in it effectively are classes, their associations and packages. These elements are defined graphically using UML notation [15].

B. Navigation Modelling

Navigation design tells about the specification in which objects are defined to be accessed through some navigational structure of the web application and the ways about these object's accessibility in the access structure. The main navigation modelling elements are the navigation classes and their corresponding associated navigational links [16]. The elements that are used to design the UWE navigation model are [17-18]:

1.  : Navigation Class
2.  : Menu
3.  : External Node
4.  : Query
5.  : Guided Tour
6.  : Index
7.  : Navigation Link

C. Presentation Modelling

The presentation model is creating from navigation objects and their access primitives. This model tells about the access primitives and their corresponding objects, which are accessible by the user. Navigation structure is transformed by presentation model into a set of models [19]. UWE offers a number of modelling elements, which can explain the abstract user interfaces, and that is 'text,' 'form,' 'image,' 'audio,' 'video,' 'button,' 'anchor' and different collections [20]. The elements that are used to design presentation model are [17-18]:

1.  : Presentation Alternatives
2.  : Presentation Group
3.  : Iterated Presentation Group
4.  : Input Form
5.  : Presentation Page
6.  : Tab
7.  : Button
8.  : Anchor
9.  : Text
10.  : Image
11.  : Media Object
12.  : Selection
13.  : File Upload
14.  : Customs Component
15.  : Slider
16.  : Text Input
17.  : Image Input

3.2. Web Modeling Language (WebML) Method

In 1998, an Italian Politecnico di Milano research group brought WebML [21] into attention. In web application areas and execution platform, UML is a visual modelling language of structuring, specification and documentation of systems. A distinctive language for associating software systems, it is also broadly embraced by academia as well as industry.

3.2.1. WebML Metamodel

A Metamodel is the best way to model a continuously changing notation and maintain it in a homogeneous and comprehensive way [14]. There are different notations and metamodeling languages within them a set of Object Constraint Language (OCL), Meta Object Facility (MOF) and Object Management Group (OMG) for denoting metamodels [22]. A WebML is represented by four Metamodel packages as shown in the below diagram [23]: Common element, Data View, Hypertext View and Presentation View as shown in Figure 2.

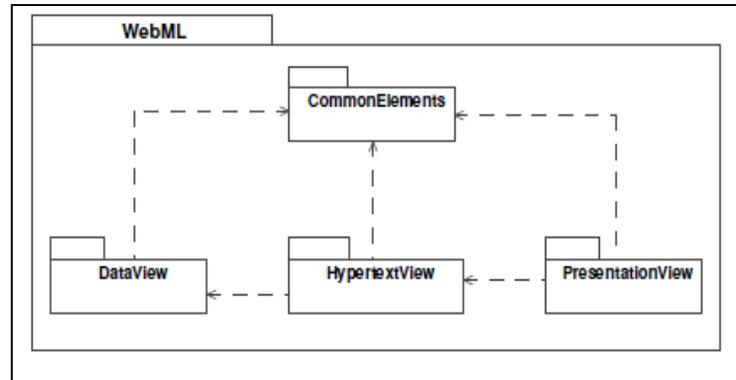


Figure 2. The WebML metamodel packages [23]

A. Data Modeling

An appropriate conceptual models' adaptation to design of data is the WebML data model, as previously used in different disciplines being used, for example, database design, knowledge representation and software engineering. The Entity-Relationship data model, employed in conceptual database design, and UML class diagrams, employed in object-oriented modeling are incompatible with it.

B. Hypertext Modeling

The arrangement and navigation of the site are distinctly explained through hypertext modelling. To arrangement can be beneficial to ensure which pages produce the hypertext and which content units create a page.

Atomic content components employed in order to publish the data explained in the data model are defined as Units. In order to compose pages in WebML, units are classified to seven types, including; data, entry, index (with its variants multi option and hierarchical), multi-data and scroller. To each unit one underlying entity is connected and is from which the unit component is calculated. The underlying entity's specification orders the object type from which unit content is extracted, for example artists, albums and so on. When it is suitable, a selector can connect to units in an optional manner. It means that the specification of a group of limitations, which specify the true underlying entity's instances, can be employed as the components of the unit at runtime. The elements or stereotypes for hypertext diagram are [21]:

1. Data Units:  : Data units are defined to select a mix of information, which provides a meaningful view of a given concept of the structure schema. More than one unit can be defined for the same entity or component, to offer alternative points of view.
2. Multi-Data Units:  : Multi-data units present multiple instances of an entity or component together, by repeating the presentation of several, identical data units.
3. Index Units:  : Index units present multiple instances of an entity or component as a list, by denoting each object as an entry in the list.
4. Scroller Units:  : Scroller units provide commands to scroll through the objects in a container, e.g., all instances of an entity or all the objects associated to another object via a relationship.
5. Entry units:  : Support form-based data entry. They are used for gathering input, which is typically employed to do the following: perform searches over the objects of an entity supply parameters to operations like content updates, login, and external services.
6. Hierarchic al index unit:  : In which the index entries are organized in a multi-level tree. The hierarchy is represented by a sequence of N source entities connected by N-1 relationship roles.
7. Multi-choice index unit:  : In which each element of the list of entries is associated with a checkbox, allowing the user to select multiple objects, instead of a single one.

Links determine the site Navigation. Definition of Links is connections between the units within just one page, connections between units located in distinctive pages, and between pages.

C. Presentation Modeling

The orthogonal role of explaining the pages' look and feel in a site view is known as presentation. A specific model for presentation at the conceptual degree is not included in WebML, which contains methods of the standard leverages with more similarities to specialists of graphic and communication [24].

The process design of Presentation Model according to [25] consists of the following steps:

- Concerned with the look-and-feel of web pages.
- Page is the basic unit of presentation.
- A page is associated to one or more style sheets.
- Style sheets are formally expressed in XML.
- A default page style is generated for each page.

3.3. Object Oriented Hypermedia (OOH) Method

OOH is an initiative started in 2000 by Gómez and Cachero [26-27]. OOH, was originally defined as an approximation addressed by the user requirements, object oriented-based and partially based on the standards. Based on the object-oriented paradigm, this approach provides designers with the semantics and notation necessary for the development of personalized Web-based interfaces [26].

3.3.1. OOH Metamodel

Figure 3 represents the ensuing OOH metamodel. Layer 0 and 2 modelling necessities are realized through consisting of the packages in OOH metamodel, as same the WebML metamodel that is respectively the service package and content package. Two packages are defined in layer 1 named presentation and navigation packages. Content model of OOH depends on the UML class diagram. A description of behavioural features and common structural are represented by a class.

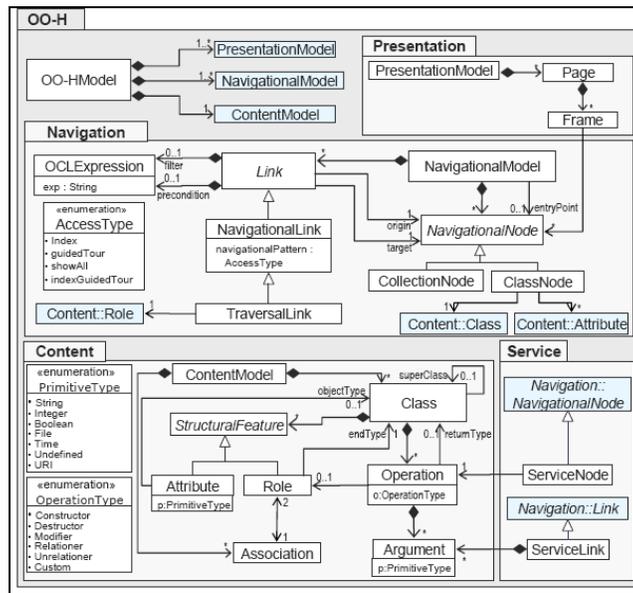


Figure 3. OOH Metamodel [23]

A. Class diagram

Class diagram has to mention that derived is stood by a slash (/) next to an attribute/method, and class-scope attribute/method is symbolized a dollar symbol (\$) next to an attribute/method name. The mechanism UML employs for explaining enumerated kinds is the <<enumeration>> stereotype which is determined by classes. Those stereotyped classes 'characteristics depict the mighty importance of the enumerated type [27].

B. Navigation Access Diagram (NAD)

Navigation classes navigation targets; navigation links and collections are the principal parts of the NAD.

- Navigation Classes: Navigation Classes (NC) possess their grounding in the classes recognized throughout the conceptual modeling stage, and they are symbolized via a rectangle with three domains [28-29]:

- Head: It includes the class's name.
- Attribute area: The names and the attributes scope (attributes of characteristics) pertinent to the regarded agent and view are included in this area.

- Service area: The services with the ability to being protected by the real NAD agent, are collected by the service area. Attribute visibility are classified to three models [28-29]:
 - Always Visible (V): All the object's views can show their value.
 - Referenced (R): In this type of attribute visibility their value is just referenced and thus their advising needs another s in the navigation path. The accessing way of this type of attribute can change depending on the implementation environment.
 - Hidden (H): Their value is not referenced and also is not displayed. Therefore, the single path to access them is via a system view with details.
- Navigation Targets: The NC are classified into Navigation Targets (NT) which is a group of NC equipping cooperatively the agent with a consistent view of the system. Association an NT to each navigation requirement of the user is the general rule. There is an associated scope in the NT: local to the actual(real)agent type and thus to the real NAD or global to the system. A rectangle that collects every class included in that view presents an NT representation through the diagrams [30-34].
- Navigation Links: Five different attributes including name, source navigation class, target navigation class, associated navigation patterns, associated navigation filters define a Navigation Link (NL).

However, four types of NL are included in OOH Method [28-29]:

- Lr (requirement link): The access point is displayed to the NT via Lr. Each NT has a requirement link and a black circle with an arrow referring to the root navigation class or also to a group within that NT can show it.
- Ls (service link): A navigation class service is pointed with Ls and a ray-arrow sketched to show it, and mightily values of parameters associated are involved in it.
- Li (internal link): A provided NT has both, its source and target NC inside itself. Its activation without alter the context of the user and without production of user disorientation is major traits of it.
- Lt (traversal link): It is explained between navigation classes owned by various navigation targets, and, as a result, explains optional visualization ways to the target classes' objects.
- There are four navigation patterns which are explained by OOH Method, and they possibly connected with navigation links and collections consists of Index, Guided Tour, Indexed Guided Tour and Show all, see [28-29]:

-Navigation Filters Associated to links and collections. In addition, a list of Navigation Filters (NF) can be explained. The order, the quantity (number) or the quality (attributes) of the target objects can be limited by a navigation filter. Conventionally, a navigation filter is defined as well created formula (declared in a dynamic logic's subset [35]) that causes a limitation on the target class's attributes. Filters have three types [28-29]:

- Attribute-letters: They clearly describe values (expense, cost) that should be matched by the similar attribute values of the target (aimed) population.
- Condition-letters: They can describe parameters of the method (in the case of connected to a service link) or extra principle and limitations on the target population. The value of the filtersymbolized by a \$ indicates that the user should give such value previously crossing the similar link. A method to define user-dependent target populations is prepared by this mechanism.

- Order-lters: The order is clearly described in one that the target population will be gained entrance. There is a partial view on the Librarian NAD. And the related requirement, navigation and service links are included in Order-lters.

C. Abstract Presentation Diagram (APD)

The elements APD are:

- Tstruct: Useable to capture the data that required to be displayed.
- Tform: Functional at the time that the page (apart from information) comprises calls to underlying(fundamental)logic.
- Tlink: The interconnection and dependencies among pages are taken byTlink.
- Tfunction: Client functionality that used in the different pages is collected by Tfunction.
- Texternal: It is employed to collect type, location and behavior of external components including images, applets that can filter the first interface.
- Tlayout: It is the place that catches the location of elements, and the definition of concurrent views and synchronization are captured.
- Tstyle: It is the place of features preservation that done by OOH, and these features are included typography or color palette for each of the interface element.
- Twidget: It is the place that implementation constructs are connected to different data and interaction (communication) items which are dependent on the ultimate implementation platform and language.
- Tlogic: It is the place that the system maintains details of implementation concerning interaction with underlying(basic) business logic (service, parameters, connection protocol, and so on).

4. COMPARISON BETWEEN THREE METHODS AND EVALUATION RESULTS

In the following, Table 1 shows the level of support of the model elements for developing web applications. In the table shows a comparison between metamodels of the three methods, which are UWE, WebML and OOH for supporting web applications. The following explain the type of support models design according the three metamodels:

- Conceptual model: In the conceptual model UWE, WebML and OOH fully support the web page content.
- Navigation model: In the Navigation model UWE cannot support all models in web page contents, among them WebML best method to support web page contents, but OOH also cannot fully support web page contents.
- Presentation model: In the Presentation model, three methods are good to support the presentation but cannot fully support, which OOH showed a weak support.

Table 1. Comparison between elements UWE, WebML and OOH

No.	Model Elements	Content Model			Navigation Model			Presentation Model		
		UWE	Web ML	OOH	UWE	Web ML	OOH	UWE	Web ML	OOH
1	Text	●	●	●	◐	●	●	●	●	●
2	Image	●	●	●	◐	●	●	●	●	●
3	Menu	●	●	●	●	●	●	●	●	●
4	Hierarchical index	●	●	●	◐	●	●	◐	●	●
5	Query	●	●	●	●	●	●	●	●	●
6	Input form	●	●	●	◐	●	●	●	●	●
7	Multi choice	●	●	●	●	●	◐	●	●	●
8	slider	●	●	●	○	●	◐	●	●	○
9	Custom component	●	●	●	●	●	◐	●	●	●
10	File up loader	●	●	●	○	○	◐	●	●	○
11	Button	●	●	●	○	○	◐	●	●	○
12	Media Object	●	●	●	◐	○	◐	●	●	○
13	Anchor	●	●	●	●	●	●	●	●	○
14	Selection	●	●	●	○	●	●	●	●	○
15	Button	●	●	●	○	◐	◐	●	●	○
16	Tab	●	●	●	○	○	◐	●	●	○
17	page	●	●	●	○	●	◐	●	●	○
Not support		○	Partially support		◐	Fully support		●		

Three methods, including OOH, UWE, and WebML have usability design metamodels, and present efforts are restricted to the demonstration of the concept of the first access of them. In addition, there is a restriction to the number of issues, which are different techniques must succeed in a small group of fundamental characteristics of Web applications.

According to Table 1 the WebML is the best method among the three methods for development web applications, and UWE is a good method but lower than WebML. OOH is the lowest method for development web applications.

5. CASE STUDY AND IMPLEMENTATION

In this paper, we are using a simple case study of showing part of a result Table1 as practically, we select ZiBA's website [36] website as a case study. For simply the implementation we design only navigation model for a case study of the three methods UWE, WebML, and OOH. Figure 4 shows ZiBA technology homepage.



Figure 4. ZiBA Technology homepage [36]

However, our case study is simple, but we can select a number category of design for showing the capability of the three methods like as Logo classified under picture category, menu with submenu under the main menu and so on. The Table 2 shows the category of model design from the ZiBA technology website.

Table 2. Category of Design for the case study

Category of Design	Models of design
Picture	Logo
Main menu	Menu with sub menu
Menu	Menu without sub menu
Slideshow	Shows the pictures with headline and move it
Search	Input box for search
Contact form	Input box for sending mail
Text	Information text such as Welcome to Ziba Technology

5.1. Design Case Study by UWE

For drawing navigation model from the conceptual model, we provided by ArgoUWE, and when Navigation Diagram from the menu ‘Create Diagram’ ArgoUWE creates navigation classes and associations for every “navigation relevant” conceptual classes and every association among them into a recent navigation model. Some links can be added by the modeler following the creation of the navigation diagram automatically.

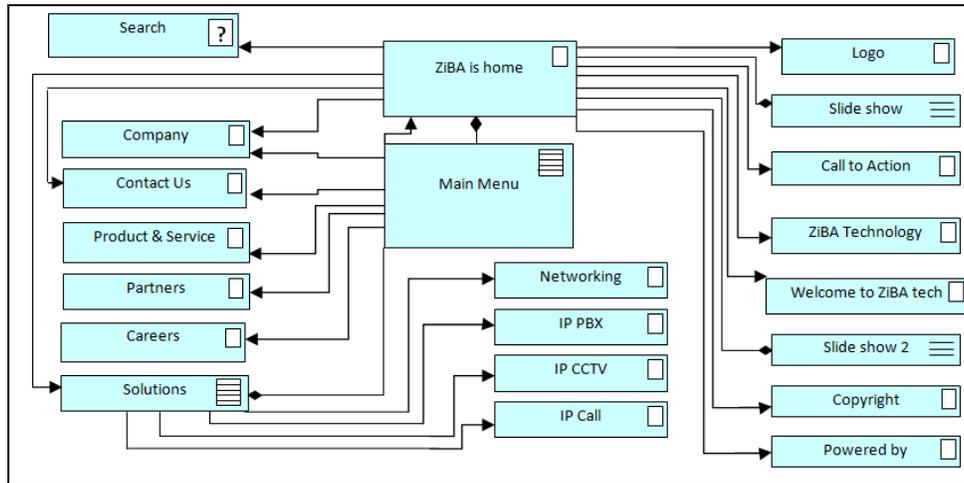


Figure 5. Design ZiBA Website by UWE

We designed UWE Navigation model for the ZiBA's website by using the current elements in UWE Navigation model and ArgoUWE as shown in Figure 5, in this diagram; we found the following points according to models:

- UWE fully supported picture, menu, search, and text.
- UWE half supported main menu and slide show; we used menu element instead of the main menu and used index instead of a slide show.
- UWE could not support contact Form; we used navigation class instead of special element.

5.2. Design Case Study by OOH

Navigation classes navigation targets; navigation links and collections are the principal parts of the NAD. We will design the NAD for the UTM homepage as showed in Figure 6.

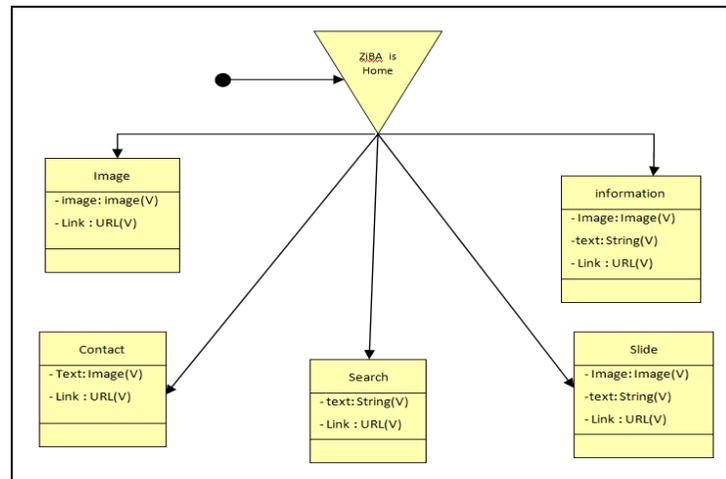


Figure 7. Design ZiBA Website by OOH

We designed OOH NAD for the ZiBA's website by using the current elements in OOH as shown in Figures 7, in these diagram OOH fully supported picture and text, but it is half supported other models, we used another element for representing it.

5.3. Design Case Study by WebML

Hypertext modeling clearly describes the composition and navigation of the site for drawing, Hypertext model in the WebML, we provided by WebRatio and Current elements from WebML.

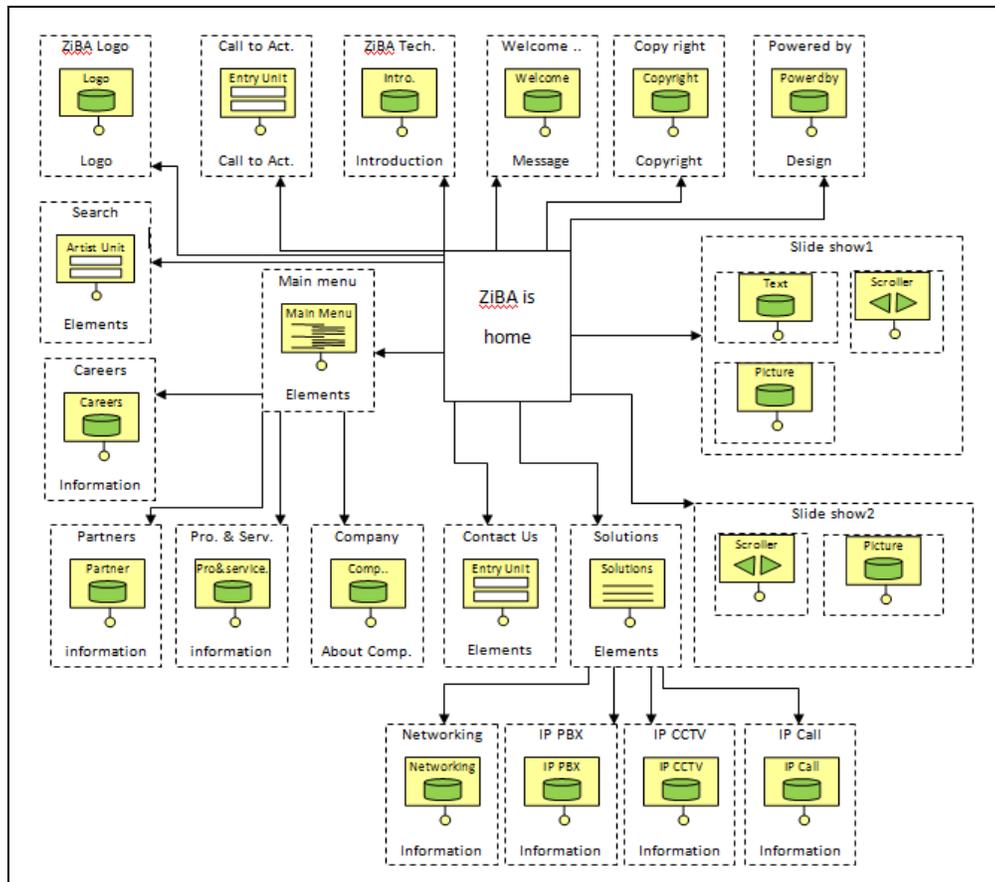


Figure 6. Design ZiBA Website by WebML

We designed WebML Hypertext model for the ZiBA's website by using the current elements in WebMI Hypertext model, and WebRatio as shown in Figures 6, in these diagrams showed the WebML can fully support all models because each of the models has a special element for representing it.

Table 3 shows the capability support the three methods for ZiBA's Website and comparison between the methods.

Table 3. Comparison of Navigation Model between UWE, WebML and OOH

Models	UWE	WebML	OOH
Picture	●	●	●
Main menu	◐	●	◐
menu	●	●	◐
Slideshow	◐	●	◐
Search	●	●	◐
Contact form	○	●	◐
Text	●	●	●

In Table 3, showed the resulting design of the navigation model in UWE, WebML, and OOH. UWE and OOH cannot fully support all models, but WebML is better than UWE and OOH because WebML is a graphical language which also has more elements for representing models of design. We can say Table 3 has the same result of the Table 1.

6. CONCLUSION AND FUTURE WORK

In this paper, we compared UWE, WebML, and OOH methods, through the current elements of the metamodels, to support web applications as showed in Table1. This comparison showed WebML is best to support web applications among them, also UWE can be well supported web applications, but OOH can week support web application compare UWE and WebML, after the design navigation model for ca case study by UWE, WebML and OOH, we make a comparison between them as showed in Table3, we got the same results. This comparison helps the designers how and which time chooses which method in web engineering for development web applications.

We recommend for the researcher for extend our idea through design more than one case study of navigation model and presentation model. Also, they can extend our comparison by used all methods of web engineering. We recommended for the researchers to enhance the web engineering methods weaknesses through the mechanism for adding new elements of metamodels.

REFERENCES

- [1] N. Koch and A. Kraus, "Towards a common metamodel for the development of web applications," in *Web Engineering*, ed: Springer, 2003, pp. 497-506.
- [2] A. Kraus and N. Koch, "A metamodel for UWE," Citeseer2003.
- [3] J.N. P. de Koch, "Software Engineering for Adaptive Hypermedia Systems-Reference Model, Modeling Techniques and Development Process," 2001.
- [4] H. Lee, et al., "A scenario-based object-oriented methodology for developing hypermedia information systems," in *System Sciences, 1998.*, Proceedings of the Thirty-First Hawaii International Conference on, 1998, pp. 47-56.
- [5] A. Gu, et al., "Web modelling languages: the gap between requirements and current exemplars," 2002.
- [6] S. Montero, et al., "Requirements for hypermedia development methods: A survey of outstanding methods," in *Advanced Information Systems Engineering, 2002*, pp. 747-751.
- [7] L. Baresi and S. Morasca, "Three empirical studies on estimating the design effort of Web applications," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 16, p. 15, 2007.
- [8] G. Rossi, et al., *Web engineering: modelling and implementing web applications vol. 12*: Springer, 2008.
- [9] N. P. d. Koch, "Software engineering for adaptive hypermedia systems," Ph.D. Thesis, Verlag Uni-Druck, Munich, 2001.
- [10] N. Koch and A. Kraus, "The expressive power of uml-based web engineering," in *Second International Workshop on Web-oriented Software Technology (IWWOST02)*, 2002.
- [11] O. OMG, "Unified Modeling Language (OMG UML)," ed: Superstructure, 2007.
- [12] M. V. Torres Bosch, "A web engineering approach for the development of business process-driven web applications," 2008.
- [13] C. Atkinson and T. Kühne, "The essence of multilevel metamodeling," «UML» 2001-The Unified Modeling Language. Modeling Languages, Concepts, and Tools, vol. 2185, pp. 19-33, 2001.

- [14] N. Koch and A. Kraus, "Towards a common metamodel for the development of web applications," *Web Engineering*, pp. 419-422, 2003.
- [15] O. Pastor, "Conceptual modeling meets the human genome," *Conceptual Modeling-ER 2008*, pp. 1-11, 2008.
- [16] P. Van Bommel, *Information modeling for Internet applications*: Idea Group Publishing, 2003.
- [17] C. Kroiß and N. Koch, "The UWE Metamodel and Profile—User Guide and Reference," *Technical Report 1101*, Ludwig-Maximilians-Universität München 2011.
- [18] U. U.-b. W. Engineering. (2012, Profile Overview. Available: <http://uwe.pst.ifi.lmu.de/profileOverview.html>
- [19] K. Siau and T. A. Halpin, *Unified Modeling Language: Systems Analysis, Design and Development Issues*: Igi Global, 2001.
- [20] A. Knapp, et al., "ArgoUWE: A CASE tool for Web applications," in *First International Workshop on Engineering Methods to Support Information Systems Evolution (EMSISE'03)*, 2003.
- [21] S. Ceri, et al., *Morgan Kaufmann series in data management systems: Designing data-intensive Web applications*: Morgan Kaufmann Pub, 2003.
- [22] N. Moreno, et al., "WebML modelling in UML," *Software, IET*, vol. 1, pp. 67-80, 2007.
- [23] N. Moreno, et al., "A UML 2.0 profile for WebML modeling," in *Workshop proceedings of the sixth international conference on Web engineering*, 2006, p. 4.
- [24] T. W. M. Language. (2013, 5-jan). Presentation Model. Available: <http://webml.org/webml/page6.do?dau4.oid=5&inu4.current=5&UserCtxParam=0&GroupCtxParam=0&ctx1=EN>
- [25] W. Engineering. (2012, 5-October). Model-Driven Web Engineering (WebML). Available: <http://webeng.blog.globis.ethz.ch/files/>
- [26] J. Gómez, et al., "Extending a conceptual modelling approach to web application design," in *Advanced Information Systems Engineering*, 2000, pp. 79-93.
- [27] J. Gómez and C. Cachero, "OOH Method: extending UML to model web interfaces," *Information modeling for internet applications*, pp. 144-173, 2003.
- [28] I. G. Fernández, *A-OOH, Extending Web Application Design with Dynamic Personalization*: Taller Digital, 2008.
- [29] J. Gómez, et al., "Extending a Conceptual Modelling Approach to Web Application Design," in *Proc. of the 1st International Workshop on Web-Oriented Software Technology*, 2001.
- [30] D. Schwabe, et al., "Systematic hypermedia application design with OOHDM," in *Proceedings of the seventh ACM conference on Hypertext*, 1996, pp. 116-128.
- [31] F. Garzotto, et al., "HDM—a model-based approach to hypertext application design," *ACM Transactions on Information Systems (TOIS)*, vol. 11, pp. 1-26, 1993.
- [32] P. Fraternali and P. Paolini, "A conceptual model and a tool environment for developing more scalable, dynamic, and customizable web applications," *Advances in Database Technology—EDBT'98*, pp. 419-435, 1998.
- [33] P. Atzeni, et al., "Design and maintenance of data-intensive web sites," *Advances in Database Technology—EDBT'98*, pp. 436-450, 1998.
- [34] S. Ceri, et al., "Design principles for data-intensive Web sites," *SIGMOD record*, vol. 28, pp. 84-89, 1999.
- [35] D. Harel, et al., *Dynamic logic*: Weizmann Institute of Science, 1983.
- [36] ZiBA. (2014, 8 March). ziba technology. Available: www.zibatechnology.com

Authors

Karzan Wakil he is a lecturer at the Fine Arts Institute in Sulaimaniyah-Iraq, received B.Sc. Degree in Computer Science from the University of Salahaddin-Iraq, 2006, and M.Sc. in Computer Science from University Technology Malaysia (UTM), Malaysia, 2013. His research areas are Web Engineering, Software Engineering and Web System. karzanwakil@gmail.com.



Dayang N.A Jawawi is an Associate Professor in Software Engineering Department, Faculty of Computing, Universiti Teknologi Malaysia (UTM). She received her B.Sc. in Software Engineering from Sheffield Hallam University, UK, and her M.Sc. And Ph.D. in the field of Software Engineering from Universiti Teknologi Malaysia. She has been an academic staff at Software Engineering Department for over ten years, and she is currently serving as the Head of Department since October 2009. She is a member of the Software Engineering Research Group (SERG), K-Economy Research Alliance, UTM, dayang@utm.my, <http://comp.utm.my/dayang>.

