International Journal of Engineering & Technology, 7 (4.38) (2018) 1420-1425

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An Interview-Based Approach to Elicit User Interface Design for Web Applications

Nur Atikah Shamat*, Shahida Sulaiman*, Jacline Sudah Sinpang

Department of Software Engineering, Faculty of Computing,
Universiti Teknologi Malaysia, 81310, Johor Bahru, Johor, Malaysia

Abstract

Elicitation of user interface design for Web applications is a process to gather precise information from users to aid software engineers in designing the user interface according to users' wants and needs. There are several issues that are related to the user interface design if they do not fulfill users' expectations. For instance, software engineers misunderstand what users need and lack of information from users about the requirements to design the user interfaces. Thus, the elicitation of user interface design is vital in order to reduce the problem especially for Web applications. This research proposes an interview-based approach to elicit user interface design for Web applications. A prototype tool has been developed to help software engineers elicit the user interface requirements from users precisely. The evaluation shows that the proposed approach has the possibility to reduce the misunderstanding in user interface design with the focus on input control elements.

Keywords: *User Interface Design, Web Applications, Interview-based Approach*

1. Introduction

User interface can be described as the interaction from users with any devices such as mouse and keyboard. Jincheng et al.1 state that user interface is the foremost and replaceable components of any software. User interface design (UID) for a Web application involves users who use the Web to perform the task that should be completed. The eliciting process of UID for a Web application is a process to gather more precise information from a user with regard to the requirement functionality via the UID for the Web application that will be developed. This research anticipates that the proposed approach and its tool can be used to guide the process of elicitation of UID for Web applications from users.

UID specifies what user needs in term of look and feel of a software system and what software engineers understand based on user requirements. However, sometimes software engineers develop user interfaces with little or few supports or guidance from professional user interface designers2. Besides, UID is a mind boggling process towards the achievement of a software system such that planning an interactive system which is attractive, accessible, and easy to use is a challenging task3. UID is the design of interfaces that users can see from any devices such as computers and tablets. Galitz4 defines the UID as "a subset of the field of study called human-computer-interaction (HCI)". HCI covers the possibility to support individuals interact with machines5. UID acts as an intermediary for the user to interact with the system and it only focuses on interfaces and in this research Web application will be the focus.

Web application is a common application to people nowadays. Web-based applications can be categorized into seven classes6 as shown in Table 1 in spite of the fact that a given application may belong to more than one class.

Table. 1. Category of Web applications6

Category	Examples
Informational	Online newspapers, product catalogs, newsletter, service manuals, online classified, online electronic books
Interactive (User-provided information or customized access)	Registration forms, customized information presentation, online games
Transactional	Electronic shopping, ordering goods and services, online banking
Workflow	Online planning and scheduling systems, inventory management, status monitoring
Collaborative work environment	Distributed authoring systems, collaborative design tools
Online communities, marketplaces	Chat groups, recommender system that recommends products or services, online marketplaces, online auctions
Web portals	Electronic shopping malls, online intermediaries

More applications nowadays are being relocated into Web applications. UID for Web applications should focus on its functionality to offer simple, intuitive and responsive user interfaces that give the user a chance to complete things with less exertion and time. According to Zhu7, a user interface portrays the elements of a Web application without explaining its utilizations and how the capacities are executed, and also the UID depicts the presentation of input and output as parts of the user interface. Islam and Bouwman8 state that interface sign can be called as the important elements of Web user interfaces such as navigational links, small images, thumbnails, short text, and button. Elements of interfaces consist of four categories which are input controls, navigational components, informational components, and containers. Table 2 describes the function of each element according to the categories.

Table. 2. Elements of user interface design[9]

Input Control	
Element	**Description**
Buttons	Indicates an action upon touch and is typically labelled using text, icon or both
Checkboxes	Permit user to select one or more options. It is typically best to show in a vertical list
Date picker	Permit users to select date so that the data is constantly organized and input into the system
Drop-down lists	Similar to the radio button. But are more solid letting you save space. Consider adding text to the field so that user identifies the essential act
List boxes	Same as checkboxes but are more compact and can support a longer list of an option if needed.
Radio buttons	Permits users to select one item at a time
Text fields	Permit user to enter text either a single line or multiple lines
Toggles	Permit users to change a setting between two states which are most effective when states on/off are visually distinct
Buttons	Indicates an action upon touch and is typically labelled using text, icon or both
Navigational Component	
Element	**Description**
Breadcrumb	Permit users to recognize their current location
Icons	Basic image as an instinctive sign to helps users to navigate the system
Image Carousel	Allow users to browse through a set of items and make a selection of one if they so choose
Pagination	Divides contents between pages and permit users to skip between pages
Search field	Permit users to enter keywords and submit to search the index
Sliders	Recognized as track bar which permits users to set or alter a value
Tags	Permit users to find content in the same category
Informational Components	
Element	**Description**
Message boxes	Small window that provides info to users and needs an action before proceed
Modal window (pop-up)	Needs users to cooperate in some way before they can return to the system
Notifications	An update message that states something new
Progress bars	Specifies where a user is as they advance through a sequence of steps in a process
Tooltips	Permit users to see suggestions when they hover over an item
Containers	
Element	**Description**
Accordion	Vertically loaded list of items that show/hide functionally

The input control elements are the focus to elicit UID using the proposed tool for this research. They include buttons, check boxes, date picker, drop-down list, list boxes, radio buttons, text fields, and toggles. In addition, a CSS is also an important aspect in designing Web documents. Paulson10 believes that many developers use CSS to create stylesheets and to define different elements for each Web page. The properties are animation, backgrounds and borders, basic box, basic user interface among others11. Although many integrated development environment (IDE) nowadays provide CSS, software engineers may overlook some aspects when eliciting user requirements that should be included in the UID. Avoiding any misunderstanding earlier in UID is crucial, at the same time it should follow the detail guidelines or the best practices when designing each element of UID. Hence, the main goal of this research is to propose an interview-based approach to elicit UID for Web applications. The prototype tool of the proposed approach should guide software engineers in eliciting UID from users.

Section 2 describes the related work followed by the proposed approach in Section 3. The evaluation is elaborated in Section 4. Finally, Section 5 concludes the study and its possible future work.

2. Related Work

The related work adopts the systematic literature review (SLR) to identify the gaps in current research and to give a system or foundation with a specific end goal to position new research activities12. There are limited existing works that use the SLR as their method to review UID issues systematically with different focuses. The work by Ngadiman et al.13 discusses the attractiveness and learnability factors in Web applications. Besides, Islam14 also uses SLR review as the method to identify the strengths, gaps, and challenges of the related work for the semiotics perception in user interfaces. In another study, the SLR has also been used to a physical exploration of some related works to identify the criteria and gaps of usability and security in UID respectively15.

With regard to the issues associated with UID, Eisenstein and Puerta16 state that the design issues involve the stylistic preference and flexible standard of achievement as human designers frequently follow the organization and ignoring any strict rule-based processes. This means that the proposed components make design tasks to be difficult to computerize. Hussey17 states that recording and documenting of UID has a deficiency in accurate method to be used by developers. The experimentation, exploration and continual assessment are the creative processes that are commonly non-linear and iterative which the modern user interfaces inexplicitly supported18. The user-centered method is still underused and hard to understand by software development teams as it is developed independently among software engineering community19.

Wenting et al.20 report that with different background of users, a different cognitive processing that affects the way they use the computer that may lead the UID as a significant issue in information processing system. Usability complications are unbearable as it affects the software systems negatively21. While UID is still not generally used even though it is being a set of proven, well-documented, contextualized approval for solving problems in UID as the designers lack of the tools to help them to resolve UID problems22. The user interface has become complicated as systems face the ambiguity when suitable functions are combined but cannot be fully used23.

The current solutions in UID elicitation include the use of metadata to store the elements of Web application user interfaces where software engineers can manage the elements dynamically without having to go through the codes24. The approach uses the user interface metadata editor as the tool to manipulate the metadata of user interface that allows people to edit the elements of user interface. The work by Ghiani et al.25 proposes MashUpEditor that allows end users to create the Web by reusing the existing components without requiring any knowledge of JavaScript. It has an editor to create new mashup widget from existing Web application components using a Web browser. Besides, it also uses an intuitive and familiar copy-paste metaphor to create novel Web applications.

By combining two presented features that are XML-content using Web 2.0 components, different output formats and application architecture using semantic Web facilities, the Web interfaces are generated automatically through an architecture for building and modeling application26. Lastly, Bojnord27 proposes WebSTUIRE that can assist developers to elicit and analyze user interface requirements by using a combination of both scenario-based and prototyping techniques. This tool also helps developers to communicate with the end user in a Web-based environment. Thus, our proposed approach aims to eliminate some of the weaknesses derived from the SLR by providing an interview-based approach in UID elicitation with the support of a tool to guide software engineers. The details of the SLR are available in our previous work28.

3. Proposed Approach

The proposed approach will gain and elicit information from the user about the UID with the focus in input controls elements for Web applications. The prototype tool provides the questions for the criteria that need to be elicited by the user regarding the elements used in Web applications. The detailed explanation of the case study is further elaborated based on the individual code, as stated in the Appendix. The analysis is categorized based on input elements as stated in Table 2.

3.1 Conceptual Diagram

Figure 1 illustrates the proposed approach that includes all input controls to elicit the elements of UID for Web applications.

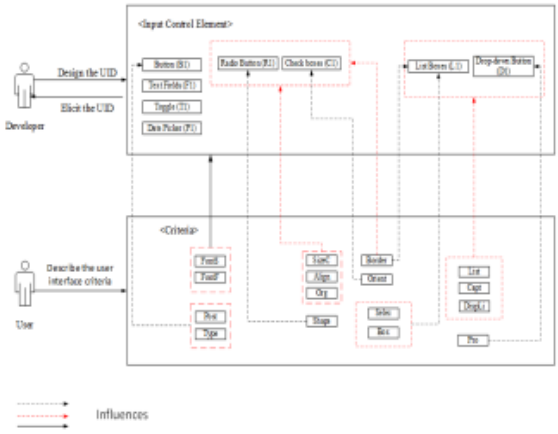

Fig. 1. Conceptual diagram of the proposed approach

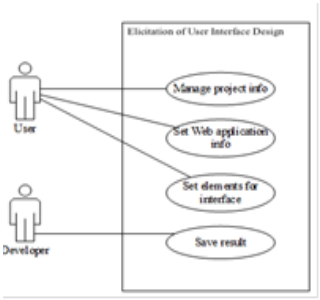

Figure 2 shows the four main requirements in the tool based on the proposed approach. The actors are user and developer.

3.2 The Prototype Design

The conceptual diagram is the reference to develop the prototype tool based on the functions and the elements of input controls of UID. The prototype tool consists of two main parts that are client and Web application information, and the element of input controls of UID. Figure 3 illustrates the algorithm.

```
Start:
Input:  ClientInfo {name, email, company},
        WebInfo {webName, webCat, webType, user, function},
        B1 {Type, Post, FontF, FontS},
        R1 {FontF, FontS, Shape, SizeC, Align, Org, Border},
        C1 {FontF, FontS, SizeC, Align, Org, Orient, Border},
        L1 {FontF, FontS List, DispLi, Border, Capt, Selec, Box},
        D1 {FontF, FontS, List, DispLi, Capt, Pro},
        P1 {FontF, FontS},
        F1 {FontF, FontS},
        T1 {FontF, FontS}
Output: Display result

Step:
        Read/input ClientInfo
        Read/input WebInfo
                If (webCat == webCat_1)
                        Then display webType_1
                Else if (webCat == webCat_2)
                        Then display webType_2
                Else if (webCat == webCat_3)
                        Then display webType_3
                Else if (webCat == webCat_4)
                        Then display webType_4
                Else if (webCat == webCat_5)
                        Then display webType_5
                Else if (webCat == webCat_6)
                        Then display webType_6
                Else if (webCat == webCat_7)
                        Then display webType_7
                EndIf
        Read/input Bi
        Read/input R1
        Read/input Ci
        Read/input L1
        Read/input Di
        Read/input P1
        Read/input Fi
        Read/input T1
        Display result
End
```

Fig. 3. Algorithm for prototype tool

Firstly, the tool will prompt the user to input their information such as name, email, and company name respectively denoted as ClientInfo. Then, the user needs to input the name of the Web application, select the category and type of the Web application, describe the user of the Web application, and lastly, describe the functionality of Web application denoted as WebInfo. For the category of the Web application (webCat), the user can choose one of the eight categories of a Web application, and it will prompt the type of Web applications (webType) according to what categories that user chooses (see also Table 1). From Figure 3, webCat_1 and webType_1 denote the category and the type of the Web application respectively.

3.3 The Implementation

Three categories in the prototype tool include information category, elicitation of input controls, and summary of the selected criteria and its respective elements. Figure 4 shows the screen shots of the prototype tool.

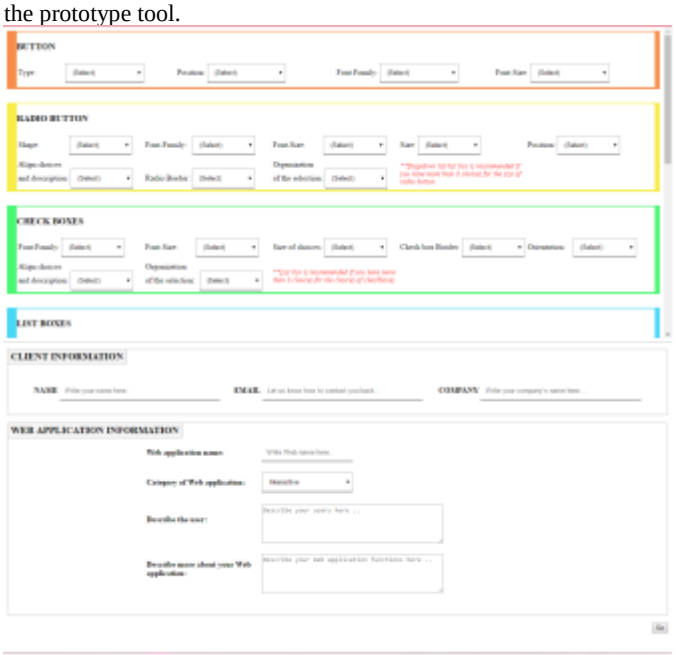

Fig. 4. Screenshots of the prototype tool

4. The Evaluation

The evaluation comprises two tasks: Task 1 (insert client and Web application information) and Task 2 (elicit user interface input controls). For Task 1, users need to access the prototype tool following the steps as described in previous section with regard to ClientInfo. Task 2 requires users to select each input control element and users need to view the provided form from the case study, which is SyerIlmu.com before starting to elicit the input controls as in Table 3.

Table. 3. Overall steps for users to elicit input controls

Step	Input Control	Description
1	Button	Select type of button used, position, font family, and font size
2	Radio Button	Choose the shape, font family, font size, radio button size, position, align choices and description, border, and organization of the selection of the radio button
3	Checkbox	Choose font family, font size, checkbox size, border, orientation, align choices and description, and organization of the selection of the checkbox
4	List Box	Choose the font family, font size, list size, box size, border, caption align, selection of lists, and summary of list box
5	Drop Down List	Choose font family, font size, list size, box size, prompt button, and caption position
6	Date Picker	Choose font family, and font size
7	Text Field	Choose font family, and font size
8	Toggle	Choose font family, and font size

The respondents were final year students in Software Engineering program invited from Faculty of Computing, UTM. Ten respondents with two different backgrounds who were students with a freelance job in Web application development and students without Web development experience. Those with Web development experience during their six-month industrial training were considered as respondents with experience. The questionnaire was distributed to each respondent, which took place at Faculty of Computing, UTM.

SyerIlmu.com was chosen to test the proposed approach to elicit UID using the prototype tool. The tasks covering the Add Group Member form of the portal comprises several input controls that are, button, radio button, check box, list box, drop-down list, date picker, and text fields. Users were provided with the manuals that explain the objectives and the guidance to complete the tasks. Each respondent needed to complete all tasks before answering the questionnaire.

4.1 The Questionnaire Design

The questionnaire consists of four sections: Section A: Task, Section B: Personal Information, Section C: Research Study and Section D: Prototype Tool. Section B includes respondents' personal information that are gender and experience in developing Web applications. Section C derives the opinions from the respondents about the study using the Likert scale 1 (low) to 5 (high). Users were directly asked about their opinions on the proposed approach used in this study. The structure of the questions for the study is as in Table 4.

Table 4. Questions on the research or the study

Code	Question
S1	Interview-based approach is suitable to elicit UID
S2	UID is the essential part in developing Web applications
S3	Software engineers should give extra attention to the UID for Web applications according to clients' wishes

Section D involves the use of the prototype to elicit the selected function in the case study. The questionnaire focuses on the example questions from the prototype tool to elicit the input controls of UID. The user was asked on how satisfied they were when using the prototype tool in eliciting the user interface elements. Section D seeks respondents' opinions about the prototype tool itself. The questions are listed in Table 5.

Table 5. Questions on the prototype tool

Code	Question
Q1	Based on the case study uses the prototype tool, do you agree that the questions related to every part of the Web page can help both users and software engineers to elicit the UID in Web applications?
P1	Type of font family used
P2	Size of the font used
P3	Position of user interface elements
P4	The number of choices for the radio button/checkboxes/ list boxes/dropdown list
P5	The chosen border for the radio button/ checkboxes/list boxes/ dropdown list
P6	Align choices and description for radio button/checkboxes
P7	Organisation of the selection for radio button/checkboxes
P8	Position for radio button/ checkboxes
P9	Orientation for checkboxes
P10	Caption align/position for list boxes/ dropdown list
P11	Recommendation to choose either radio button/check box/list box/ drop-down list
P12	The type of button
P13	The shape of radio button
P14	Selection of list for list box
P15	Summary for list box
P16	List size for dropdown list
P17	Prompt button for dropdown list
Q2	Does this prototype tool help to elicit the input controls of UID for the case study?
Q3	Do you think it is easy to use the prototype tool?
Q4	Do you think the prototype tool is useful to software engineers when eliciting UID, as in this case study the input controls?
Q5	Do you have any suggestions to improve the tool?

4.2 Analysis and Findings

Figure 5 shows the respondents' demographic in which male respondents were 70% of all the subjects. In term of experience, 60% of them had one to two years of experience in Web development.

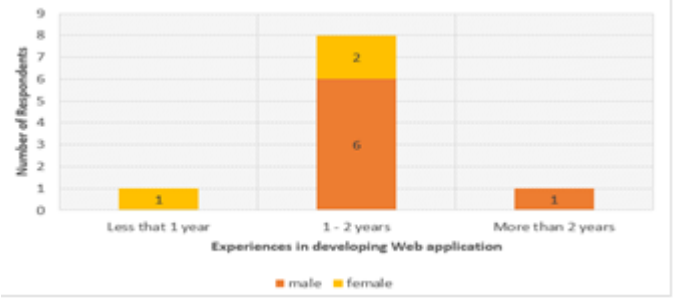

Fig. 5. Respondents' demographic

Three general questions related to the research study were provided. The question codes listed in Table 4 were used, and the scores based on the Likert scale were calculated as shown in Table 6. The

analysis shows the highest mean is for S2 that is 4.5. This concludes that most of the respondents agreed that the UID is essential when developing Web applications besides for the aspect of S1 and S3 with the mean 4.0 and 4.2 respectively.

Table. 6. Section in questionnaire set

Respondent	S1	S2	S3
R_1	4	5	4
R_2	3	4	4
R_3	4	4	5
R_4	4	5	5
R_5	4	5	5
R_6	5	4	4
R_7	5	4	3
R_8	4	4	3
R_9	4	5	5
R_{10}	3	5	4
Total score	40	45	42
Mean score	4.0	4.5	4.2
Note: R_x denotes Respondent no. x			

Figure 6 shows the mean for each code under Q1 as stated in Table 5. The analysis depicts that respondents gave the most positive opinion towards question P4, which is the prototype tool can help the user in choosing the number of choices for radio button, check boxes, list box, and drop-down list. The lowest mean is in question P5 and P13 that is 3.4 respectively. This implies that most of the respondents partially agreed on the aspect of the border and shape of the radio button.

Fig. 6. The mean for each question under the main question Q1

For Q2, eight out of ten respondents agreed that this prototype tool could help in eliciting the input controls of UID for the case study, while two respondents responded that a prototype tool is a moderate tool that can help in eliciting the input controls. On the other hand, half of the respondents agreed that the prototype tool is easy to use, while the other half of the respondent partially agreed that the tool is not easy to use, but too hard to understand when responding to Q3.

For the response to Q4, six of the respondents agreed that the provided prototype tool is useful to software engineers when eliciting UID for input controls. Only four of the respondents partially thought that the prototype tool is an average tool that could help software engineers to elicit the input controls of user interfaces.

For Q5, respondents were required to give suggestions on how to improve the prototype tool. From the questionnaire, three respondents suggested that the design of the prototype tool should be more interesting so that any users will know the important of UID for Web applications. Furthermore, another three respondents recommended that each question to be stated more clearly, so that the users will not be confused in using the tool. Two respondents recommended that the prototype tool should give an example of the user interface that has been chosen.

5. Conclusion and Future Work

UID is an important aspect when designing Web applications. Misunderstanding between software engineers and users can be eliminated by eliciting users' requirements specifically for UID besides functional and non-functional requirements. Thus, the proposed interview-based approach has shown the possibility to reduce the misunderstanding via the prototype tool that adopts the approach with the focus on input controls element. Future work includes the expansion of the proposed approach in three other elements of UID and providing the views of the designed user interface in the tool.

Acknowledgments:

This work is partially supported by the Research University Grant (RUG) of UTM, Cost Centre 11H70 and 14H09.

References

1. W. Jiancheng, L. Xudong, and L. Loi, "A Model of User Interface Design and its Code Generation," IEEE International Conference On Information Reuse and Integration on 2007, August 13-15, Las Vegas, IL: IEEE.128-133 (2007).
2. J. Johnson, "GUI Bloopers 2.0: Common User Interface Design Don'ts and Dos", Elsevier Inc: Amsterdam (2008).
3. J. C. Quiroz, S. J. Louis, A. Shankar, S. M. Dascalu, "Interactive Genetic Algorithms for User Interface Design," IEEE Congress on Evolutionary Computation on 2007, September 25-28 Singapore: IEEE. 1366-1373 (2007).
4. W. O. Galitz, "The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques," Canada: Wiley Publishing, Inc. (2002).
5. D. Stone, C. Jarett, M. Woodroffe, S. Minocha, "User Interface Design and Evaluation," Elsevier, Inc. (2005).
6. A. Ginige, S. Murugesan, "Web Engineering: an Introduction," IEEE Multimedia, 8, 1: 14-18 (2001).
7. B. Zhu, H. Miao, B. Song, "User Interface Design of Web Application Using Object-Z," Eighth IEEE/ACIS International Conference on Computer and Information Science, June 1-3. Shanghai: IEEE. 1200-1204 (2009).
8. M. N. Islam, H. Bouwman, "Towards User-Intuitive Web Interface Sign Design and Evaluation: A Semiotic Framework," International Journal of Human-Computer Studies, 86, 121-137 (2016).
9. User Interface Elements (n.d). Retrieved May 29, 2016, from http://www. Usability.gov/how-to-and-tools/methods/user-interface-elements.html
10. L. D. Paulson, "Buiding Rich Web Applications with AJAX," 38, 10, October 2005: IEEE. V1-280-V1-284 (2005).
11. CSS Reference (n.d). Retrieved May 29, 2016, from https://www.w3schools.com/cssref/
12. B. Kitchenham, S. Charters, D. Budgen, P. Brereton, M. Turner, S. Linkman, M. Jorgensen, E. Mendes, G. Visaggio, "Guidelines for Performing Systematic Literature Reviews in Software Engineering," (2007).
13. N. Ngadiman, S. Sulaiman, W. M. N. W. Kadir, "A Systematic Literature Review on Attractiveness and Learnability Factors in Web Applications," Open System (ICOS), IEEE, 22-27 (2015).
14. M. N. Islam, "A Systematic Literature Review of Semiotics Perception in User Interfaces," Journal of Systems and Information Technology, 15, 1, 45-77 (2016).
15. U. O. Nwokedi, B. A. Onyimbo, B. B. Rad, "Usability and Security in User Interface Design: A Systematic Literature Review," International Journal of Information Technology and Computer Science (IJITCS), 8, 5, 72-80 (2016).
16. J. Eisenstein, A. Puerta, "Adaptation in Automated User-Interface Design," Proceedings of the 5th International Conference on Intelligent User Inerfaces: ACM. 78-81 (2000).
17. A. Hussey, "Formal Object-Oriented User-Interface Design," Proceedings of Software Engineering Conference on 2000. Camberra, ACT: IEEE. 129-137 (2000).
18. M. Terry, E. Mynatt, "Recognizing Creative Needs in User Interface Design," Proceedings of the 4th Conference on Creativity and Cognition. ACM. 38-44 (2002).
19. A. Seffah, E. Metzker, "The Obstacles and Myths of Usability and Software Engineering," Communications of the ACM- The Blogsphere, 47, 12, 71-76 (2004).
20. L. Wenting, S. Tian, Z. Hong, Y. Ying, "Interface Design in the Ship Navigation Information System," 11th International Confer-

ence on Computer-Aided Industrial Design and Conceptual Design (CAIDCD) 2010. November 17-19. Yiwu: IEEE. 395-400 (2010).
21. M. Obeidat, S. Salim, "Integrating User Interface Design Guidelines with Adaptation Techniques to Solve Usability Problems," 3rd International Conference on Advanced Computer and Engineering (ICACTE) 2010, August 20-22, Chengdu: IEEE, V1-280-V1-284 (2010).
22. J. Janeiro, S. Barbosa, T. Springer, A. Schill, "A Flexible Model for Improving the Reuse of User Interface Design Patterns," Proceedings of the 28th ACM International Conference on Design of Communication: ACM, 215-221 (2010).
23. N. Hirasawa, S. Ogata, K. Kawai, H. Kasai, "Integration of User Interface Design Process," Proceedings of the 11th International Conference on Product Focused Software: ACM, 39-42 (2010).
24. D. G. Saputra, F. N. Azizah, "A Metadata Apprach for Building Web Application User Interface," Procedia Technology. 11, 903-911 (2013).
25. G. Ghiani, F. Paterno, L.D. Spano, G. Pintori, "An Environment for End-User Development of Web Mashups," Int. J. Human-Computer Studies, 87, 38-64 (2016).
26. E. Chavarriaga, J. A. Macias, "A Model-Driven Approach to Building Modern Semantic Web-Based Interfaces," Advances in Engineering Software, 40, 12, 1329-1334 (2009).
27. A. S. Bojnord, R. Ahmad, H. S. Bojnord, "WEBSTUIRE: Web-Based Support Tool for User Interface Design Requirement Elicitation," 3rd International eConference on Computer and Knowledge Engineering (ICCKE), IEEE, 52-58 (2013).
28. N. A. Shamat, S. Sulaiman, "A Systematic Literature Review on User Interface Design for Web Applications," Journal of Telecommunication, Electronic and Computer Engineering, 9, 3-4, 57-61 (2017).

APPENDIX

Input Control element	Code	Button Type (Type)	Position (Post)	Font Family (FontF)	Font Size (FontS)	Shape (Shape)	Size Choices (SizeC)	Align Choices (Align)	Organisation (Org)	Orientation (Orient)	List Number (List)	Display List (DispLi)	Border (Border)	Caption Align (Capt)	Selection List (Selec)	Summary Box (Box)	Prompt Button (Pro)
Buttons	B1	X	X	X	X												
Radio buttons	R1			X	X	X	X	X	X				X				
Checkbox	C1			X	X		X	X	X	X			X				
Listboxes	L1			X	X						X	X	X	X	X	X	
Drop-down lists	D1			X	X						X	X		X			X
Date picker	P1			X	X												
Text fields	F1			X	X												
Toggle	T1			X	X												