XP-PROTOTYPING APPROACH FOR USER INTERFACE DESIGN OF LARGE WEB-BASED APPLICATIONS

Omaima N. A. Al-Allaf Department of Computer Information Systems Faculty of Sciences and Information Technology Al-Zaytoonah Private University of Jordan, P.O. Box 130, Amman (11733) Amman -Jordan omaimaalallaf@alZaytoonah.edu.jo

ABSTRACT

The development of interactive web applications with Graphical User Interface (GUI) has become increasingly common in recent years. The user interface (UI) of web applications includes multimedia such as text, images, video and sound. The success of such applications depends on how the combinations of media are used and good design. The large enterprises need effective methods, tools and experienced developers to design UIs of large web applications because these applications consist of complex interacting activities, supports multiple users and developed by large number of developers. We addressed in this paper the problems of UI design based on literature and a survey conducted in five large Jordanian enterprises which undertaken large web development. We proposed in this paper a UI design approach (XP-PROTOTYPING) that combines XP agile with Throwaway Prototype processes to overcome UI design problems of large web applications.

Based on our initial evaluation, the combination of these two processes (XP-PROTOTYPING) could be more effective to make web application easier to develop; reduce risks by increasing flexibility to change and maintain, quality but this may require experienced professionals to work as XP team.

KEY WORDS

Web-based Software Engineering, Agile Software Development, Graphical User Interface (GUI), Extreme Programming (XP), User Interface (UI), Throwaway Prototype.

1. Introduction

Web engineering is concerned with the use of scientific, hypertext engineering, software engineering practices, management principles, iterative process and tools to successful development, evaluation and maintenance of high quality web applications. Web engineering aims to bring web application development under control, minimize risks, enhance quality, and manage diversity and complexity to avoid failures of web development [1]. The web site's functionality and information contained within it may change continuously over time to keep the information current and to meet user requirements [2].

The user interface (UI) design is the design of software applications with the focus on user's experience. The UI let the users interact with web applications, and users judge a system by its interface rather than its functionality [3]. A poorly designed UI can cause a user to make errors [4]. All of web applications interact through GUIs. The GUI characteristics are windows, icons, menus, pointing devices, and graphics [3]. The UIs combine several kinds of media (text, animation, images, voice, and music) to help people interact with web applications [1][5].

Successful designers build UIs with primary emphasis on the user and determine which human sensory system is most efficient at receiving specific information, then they use media that involves that human sensory system [6][7]. In recent years, the development of highly interactive web applications has become increasingly common and the acceptance of web application depends on quality of their UIs. The Prototyping approach was used to improve planning and evaluate quality of software projects by developing executable prototypes, generating ideas about how a UI can be designed [8][9]. Good UI may result in increased productivity, reduced training costs, user satisfaction, products with higher quality, and preventable user errors [1]. Two problems must be addressed in UI design: how information from the user should be provided to system? And how should information from the system be presented to user? [4]. Jeff Koke [10] presented top ten web design mistakes. Many researches addressed some common UI design problems such as lack of consistency, too much memorization, no guidance, no context sensitivity, poor response, inattention to user needs and lack of user testing, and finally not enough human computer interaction (HCI) experts involved [11][4]. There is a lack of researches related to UI design of the large scale web-based applications. Therefore, this paper addresses problems in UI design when developing large web applications based on literature and survey in five large Jordanian web development enterprises and proposes a new approach (XP-PROTOTYPING) that provides a suitable solution for these problems and lists a

set of steps that the UI designer should adopt when designing UIs of large scale web applications.

Section2 includes literature review and related studies, whereas Section 3 describes the large web-based applications. Section 4 includes the research methodology and Section 5 includes analysis of collected data. Section 6 describes in details the XP-PROTOTYPING approach and Section 7 includes analysis and evaluation of UI. Finally, Section 8 concludes the paper.

2. Literature Review

A Web application is an application that is invoked with a Web browser over the Internet. Jazaveri [12] addressed Key components of modern Web applications such as: search; tagging; user participation; and user interaction and collaboration. Many researches addressed UI design principles such as: user interaction with interface should be easy; UI should use familiar metaphors; UI should allowed the easy safe exploring with web site; should use suitable colors and fonts; suitable using of sound, touch; and aesthetics of layout [13][6][7][5]. The UI design process includes several phases: gathering of functional user requirements; analysis of system users; information architecture and flow; development of paper prototypes; usability testing prototypes with user; and actual look and feel design of GUI [11][4][14]. Throwaway prototyping is the creation of an incomplete model of parts of the system at an early stage and that will be discarded rather than becoming part of finally delivered software [11][4]. Throwaway Prototyping can be used to let the users have a first idea about the system. This has several advantages: designer can obtain early feedback, contractor can compare if the software matches specification, and allows software engineer to make accurate estimates. The use of prototyping reduced time and costs, increased user involvement and get feedback [3]. The Prototyping process includes [11][4]: identify user requirements; develop a prototype using appropriate screen layout; dialogue design and guidelines; use and evaluate the prototype with the aid of user tests. There are many literature researches related to UI design for web applications. Xavier Righetti [8] explained functions of rapid prototyping and reviewed different taxonomies. He listed the features that a prototyping tool should possess, and then he classified tools into different categories: paper and pencil, façade tools, interface builders and programming languages. Bäumer, et. al. [9] presented concepts of UI prototyping and classified the tools supporting the prototyping. Then they presented a case study of nine industrial projects where the main focus was on UI prototyping and where different tools were used to build different kinds of UI prototypes. Antti Hätinen [15] reviewed main points of two methodologies XP and Goal Oriented User Interface Design (GUID), and then presented the actual experiences of the usage of the combined methodologies Goal Oriented Extreme Programming (GO-XP) during the project. She addressed

that (GO-XP) technique requires highly experienced professionals to work as interaction designers and XP team coaches.

Edrington [16] developed a usability test to determine how well a specific interface communicates the necessary information to its user and facilitates task completion. A list of recommended changes to the interface was produced from the results of the study. The complexity of web applications presents a challenge in seeking to design UIs that are accessible to people.

The WWW is being used to host complex applications in diverse fields important for peoples work and study. The complexity of such applications presents a particular challenge in seeking to design UIs to them that are accessible to people with different disabilities. Cooper, et. al. [17] introduced an approach that can be used to address this challenge, making complex applications accessible to people with differing needs. They suggested separating interface from the application such that different users can have different interfaces depending on their needs.

Yang [18] addressed drawbacks of search and web-based applications such as: lack of standard; lack of effective search; lack of consistency; and lack of customizing support. He presented the design and implementation of the Concordia Indexing and Discovery (CMDI) system based on WWW. It emphasizes on GUI design and implementation for CINDI system. UML has been used in design and analysis procedure of CINDI system. PHP has been used in the implementation of the UI.

Balasubramanian, et. al. [19] introduced the need for a formal planning and idea generation phase in the UI design process prior to prototyping, to minimize the effects of ad hoc design practices. They proposed a systematic design methodology containing fifteen nonsequential, highly connected, and iterative set of design tasks to structure the creative thought processes of the designer with appropriate guidance. Zhang, et. al. [20] provided a conceptual framework and foundation for systematically investigating features in web environments that contribute to user satisfaction with UI.

UIs design is a major task especially on changing circumstances in web application development. One important task in web UI design is the development of data input pages. Xinyong Zhang, et, al. [21] addressed UI designers' problems such as how to meet multiple requirements of individuals, adapt to environments, enhance UI usability, and evaluate final design. They proposed an approach to address issues concerning data input page development. In this approach, data input panels can be automatically generated for all the tables in a database and then, the generated data input panels are self-adaptable to suit user interaction patterns.

Bolcer [22] talked about UI design assistant (UIDA) that addresses the specific design problems of style and integration consistency throughout UI development process and aids in the automated feedback and evaluation of a system's GUI according to knowledgebased rules and project-specific design examples. The UIDA system is able to quickly identify inconsistent style guide interpretations and UI design decisions resulting from distributed development of multiple UI sub-systems. This case arises when each sub-system conforms to the general style guide rules, but when integrated together, may appear inconsistent.

Wroblewski and Rantanen [23] outlined a set of guidelines for web application UI, addressed issues associated with UI design and presented a method for thinking about them. Finally, Sastry and Reddy [24] proposed some design principles for user interfaces for digital libraries for its effective implementation such as: Simple; Support; Familiar; Informative Feedback; Design Dialogues to Yield Closure; Prevent Errors; Multimedia Support; Profile Based Support; Lithe and Simple; Pan and Zoom Support; Accuracy; Efficient Searching with NLP support; Support of Semantic approach and Resource Description Frame Work (RDF) Technologies; Sharing and Reusing of Information; Multilingual Support; Platform Independent; Future Plug-ins Support. From the above, all the mentioned researches and studies focused on UIs designing and problems of UIs in general without notice to the size of web applications. Therefore, there is a need for studies related to the design of UIs for large scale web applications

3. Large Web-based Applications

The web applications vary in scope and complexity from small to large applications distributed across the internet. There are many categories of web applications: informational, interactive, transactional, workflow, collaborative work environment, online marketplaces, and web portals [11][1]. The web applications have properties of network intensiveness. unpredictable load. performance, availability, data driven, content sensitive, continuous evolution, and security [11]. The main factors that should be considered when designing web pages are: effectiveness, efficiency, satisfaction, content, structure, access, and style. The following general principles should be aimed at when designing web applications: completeness, controllability, consistency, redundancy, orientation, feedback, flexibility, and reversibility [4]. Many researches [25][26][27] defined a large software project to include large quantities of source code typically millions of lines, include large numbers of developers (more than 50), include high complexity of interaction between components, use of multiple programming languages, and use of multiple persistence mechanisms (files, databases).

4. Research Methodology

The practical work of this study depends on quantitative research methodology. This is done by carried out survey using questionnaire in five large Jordanian enterprises which undertaken the development of large web

applications (as a research sample). The questionnaire's instrument has two parts: identifying UI methods used by these enterprises; and symptoms of UI design that large enterprises suffer from. This questionnaire was written in English language and reviewed and validated many times by four professional professors currently working in different Jordanian Universities for graduated studies and specialized in software engineering and management. The selection of these large enterprises was based on definitions of large projects mentioned in literature researches [25][26][27]. At the same time, the size of enterprises in Jordan is determined by the Statistical Office according to many factors such as: number of employees, budget, size of projects, time required for the development process, and number of branches which belong to these enterprises. The selected enterprises have many characteristics: involve 50 and more developers in development process, use more than three programming languages in development, develop web applications to provide 50 functions or more and more than 100 web pages to users and with more than hundreds of thousands of lines of code, project duration ranging from one to three years, finally many of these companies have many branches in other countries.

A two of statistical techniques (descriptive statistics and explanatory factor analysis) were utilized in data analysis. Descriptive statistics (frequencies and percentage) were used to identify the major characteristics of UI design methods in these enterprises. The descriptive statistics and multivariate statistical were used in term of description of possible symptoms especially UI design problems at these enterprises.

5. Analysis of Collected Data

The research sample consisted of two hundred developers working in these five large enterprises in Jordan. The determination of the: number "200"; distribution of different number of questionnaires; and identification and selection of invitees were done according to the supervision of the Human Resource Department and Manager in each one of the five enterprises and this is depend on the number of developers in each enterprise. The number of retrieved questionnaires is one hundred and thirty, the reason of why the other 70 developers were not answer is that; many of them may be very busy (haven't enough time to fill the questionnaire); or they were outside Jordan (working, training or holiday) at survey time. Whereas the number of retrieved questionnaires which containing valid answers are only one hundred. Valid answers mean that the developers answer all the questions in the questionnaire with real values without any conflicting between her/his other answers and without letting many questions without answering. These one hundred questionnaires were used later in this research in the statistical analysis.

From the questionnaire results in these enterprises, we considered that many of these enterprises suffer from the following problems:

- All of these enterprises involve a number of development teams during the development process but each team is specialized in only one development phase such as: requirement gatherer and analyzers' team, UI designers' team, coders' team, quality assurance's team. This process lead to team dependencies problem (one team may be wait long time for the outcome of another team).
- The designer team communicates with all other teams and this may lead to designer team member confusing.
- The UI requirements are changing during and even at the end of UI designing process.
- Communication cap between the coders, testers, and the UI designers.
- Lack using of XP principles during UI design such as refactoring, reusability, and customer communication and feedback throughout the development process.

At the same time, the problems of UI design for web applications that are addressed in this paper are missing in most of related literature. These problems are as follows:

- This paper concentrates on UI design problems for large web applications. The previous researches focused on UI for web applications in general and did not consider web application size.
- These researches talked about web applications requirements in general and didn't focus on UI requirements.
- They did not offer a suitable method for arranging and organizing the UI design process of large web application.

6. XP-PROTOTYPING Approach

The focus of this paper is on proposing a new UI design approach named XP-PROTOTYPING that related to large web applications development to avoid the problems of UI design. The UI design of large web application depended on analysis of literature and on results of survey. The XP-PROTOTYPING includes the following steps:

- a) The enterprise should build at the beginning of the development process a management team. The role of this team is to coordinates the overall development process among the sub teams.
- b)At the beginning of the development process, the management team conducts an initial customer communication to obtain main characteristics (size, objectives, and requirements) of large web application. The objective of this activity is to classify requirements into classes according to size, complexity, and objectives.
- c)Divide the large web application into a set of sub web applications according to the requirement classes as shown in figure 1.

- d)Divide large number of developers into a set of sub teams and these teams may work in parallel or in incremental manner according to the dependability of the sub systems on each other. Assign one manager for each sub team and this manager should be a member of management team.
- e) Distribute sub web applications among these sub teams.
- f) The UI designers of each sub team start to build the initial UI using Throwaway Prototyping software process model according to the user requirements. To minimize cost, we suggest the use of paper throwaway prototype. To build the throwaway prototype, the sub team needs to communicate more with the customers. After the initial prototype is build, customer communication is undertaken to verify the prototype throw the feed back obtained from the customer. This process is repeated until the UI designer gets all customer requirements especially those that are related to UI of sub application. We suggest that XP principles should be adopted during this phase and mixed with the prototyping steps to get best customer satisfaction, low cost, and early delivery of sub application. Figure (2) shows the main steps of UI design of web applications.
- g)We suggest that each UI designer should take care of web application environment, end user culture, web application objectives, and technology used by UI designers when designing the UI for web application as shown in figure (3). The UI designers should also take in their consideration the activity analysis and planning of web application, dialogue management, and context management when designing UI.



6.1 Requirements Analysis of Web Applications

The main benefits of generating Prototype are: generating visions; supporting decision making; evaluating look and feel; supporting analysis of domain; and showing technical feasibility [9]. The understanding of user requirements is very important for producing usable web applications [11]. Gathering user input at every stage of

the process can help to reduce ongoing cost of web development [3]. The activities of this phase are done through answering the questions: what are web applications' objectives? Who do you visit site of web application?; what type of information will attract users and meet their needs?; and what are the quality goals which can be evaluated?. Then identify responsibilities for achieving quality objectives, and estimate resources and budget for these activities.



6.2 Using of XP Principles in UI Design

Agility can be applied to any software process for effective response to change, effective communication among stakeholders, drawing the customer onto team, and organizing a team to control the work performed in order to achieve rapid and incremental delivery of software [11]. Kent Beck, et. al. [28] defined 12 principles behind agile manifesto such as customer satisfaction, welcome requirements, deliver working software changing frequently, users and developers must work together project, motivated individuals, face-to-face conversation within team, working software is measure of progress, agile processes promote sustainable development, continuous attention to technical excellence, simplicity, self-organizing teams [29][11][30].

In XP-PROTOTYPING, we combine XP principles with Throwaway Prototyping in UI design to obtain results that satisfy user requirement throughout development process of large web applications. The XP team is sensitive to abilities of team members. When forming XP team, a special focus should be put on choosing suitable people who are willing and interested in physically close team and pair work.

6.3 UI Design Guidelines with XP-Prototyping

We list below a set of guidelines that should be used together with XP-PROTOTYPING to avoid UI errors [31]:

- UI designer must consider culture, context, needs, experience and capabilities of system users and should be aware of people's physical and mental limitations.
- Goal of UI design is to make the experience for the user as simple, pleasurable and useful as possible.
- UI designer must consider the language in which the system communicates with user and talk one language.
- UI designer must carefully control the amount of information to be displayed,
- Use lower case if possible.
- The UI designer should remember aesthetics and the style in which things are communicated to user. Aesthetics often come down to simple choices it's not number of colors used [6].
- Allow customization and make default representation suitable for all users.
- The UI designer must consider human factors and cognitive issues such as memory capacity, visual and audio capabilities.

7. Discussion and Evaluation

This paper proposed a new method for UI design for large web applications. Any method for UI design for web applications is tested through the usability of resultant design. Usability is measured through the following: speed of each interaction cycle; precision with which user interactions can be executed; work load required for users to execute interactions; familiar icons for moving between pages; robustness of tasks under all situations; and helpfulness with user problems [11]. In addition to these points, the usability is measured in terms of the following [11][4][14]:

- Effectiveness: testing whether a task can be performed without total failure.
- Efficiency: testing that the most routine tasks can be completed quickly and easily.
- Satisfaction: testing whether web site is enjoyable and gives control to users.
- Other specific test: to measure navigation, search facility, and failure points in site. Analyze UI errors and examine why this happens and how to avoid errors.

As an initial evaluation of the proposed XP-PROTOTYPING, we adopted interviews with the UIs designers and professional developers working in the selected five enterprises and discussed its properties with them. This is done by making simple comparisons between the proposed XP-PROTOTYPING and the earlier used UIs design models such as traditional Prototyping and the model suggested by Sommerville [4]. At the same time we compare it with the properties of the model GO-XP [15] and also with the XP agile process. We should note that most of the selected five enterprises didn't adopt any process model for UIs design and worked AD-HOC during the UIs design. Table (1) shows the comparisons between many approaches of UID and the XP-PROTOTYPING approach according to many factors.

As shown in table (1), each of the traditional Prototyping, Sommerville [4], XP, and GO-XP [15] UID approach didn't consider the large application, large enterprise and large number of developers in their considerations. Some of these approaches lack of many important properties and factors that should be considered during the UID process such as standardization, refactoring, and multidisciplinary teams. At the same time each of the GO-XP [15] and our approach XP-PROTOTYPING require professional developers and more training and effort. Finally for real evaluation, we should test the proposed model in the real world. It will be used in real UIs design situation in one or two of the selected five large enterprises and data will be collected regarding the above points.

Table (1): Comparisons between different approaches of UIs De

	Properties of the approach	Traditional Prototyping	Sommerville [4]	GO-XP [15]	ХР	XP- PROTOTYPING
1	Suitable for large Enterprise	×	×	×	×	
2	Suitable for Large number of developers	×	×	×	×	\checkmark
3	Suitable for large web applications	×	×	×	×	\checkmark
4	Offer requirement management	×	×	×	×	
5	Offer user-developer communications	V	\checkmark	\checkmark	\checkmark	\checkmark
6	Standardization	×	×	V	V	\checkmark
7	Refactoring (reuse of building blocks)	×	×	\checkmark	\checkmark	
8	Offer simple design	V	×	V	\checkmark	\checkmark
9	Multidisciplinary teams	×	×	\checkmark	\checkmark	
10	Accept changing requirements during the development process.	×	\checkmark	\checkmark	\checkmark	\checkmark
11	Evolutionary system development	\checkmark	\checkmark	V	V	\checkmark
12	Require professional developers	×	×	V	\checkmark	\checkmark
13	Require more training and effort	×	×	V	\checkmark	\checkmark

8. Conclusion

We addressed in this paper the problems of UI design. The paper is based on content analysis of current literature and a survey conducted in five large Jordanian enterprises which undertaken large web applications development. We noted from the survey and literature studies that success factors of web applications are related to the usability of their UIs.

As a solution to these problems we proposed a new UI designing approach (XP-PROTOTYPING) that combined the advantages of both XP agile methodology and Throwaway Prototype. Also this approach includes the required steps that should be adopted by UIs designers of web applications to overcome the problems of UIs design especially for large web applications.

We focused in this paper on how to use XP-PROTOTYPING for UI design in large web applications in large organizations having more than 50 developers. We solve this by dividing the large web application into a set of sub applications according to the number of requirement classes. The UI design team of large scale web application is divided into sub teams. Each sub team is assigned to UI design of sub system. XP-PROTOTYPING suggests a centralized management team to coordinate UI design process for large web applications.

We listed a set of UI design guidelines that must be adopted by UI designers to improve UI of web applications such as offer feedback, clarity, readability of text to be displayed, display only relevant information to context, display help messages, make actions reversible, and finally error information handling by presenting errors to user in understandable form and contain information on how to recover, and indicate negative consequences of error.

An important aspect to test in the new method is to check whether it can be learned faster than other methods, and whether it can be used more efficiently than other methods.

Finally, when we discussed the proposed XP-PROTOTYPING model with the interviewers from many Jordanian enterprises, we found that, the operation of dividing the large number of developers into small sub discipline teams each with different web engineering skills (multidiscipline sub team consists of requirements' gatherer, analyzer, UI designer, coder, tester, and QA), this will lead to overcome as possible as the above mentioned problems.

We suggested in this paper according the results of interviews with the professional developers and UIs designers working in these five enterprises that a combination of these two methodologies could be more effective and efficient to reduce risks by increasing flexibility, quality and learning for inexperienced teams. The XP-PROTOTYPING approach may require experienced professionals to work as XP team.

9. Future Work

The future work would be to try XP-PROTOTYPING in practice in many of large scale Jordanian enterprises with teams of professionals and skilled interaction designers to gain more experience. An important step would be to measure the performance levels before and after the change of methodology. Then, we will make a comparative study between the XP-PROTOTYPING and other UI design techniques used by these enterprises.

References

- [1] A. Ginige and S. Murugesan, Web Engineering: An Introduction, *IEEE MultiMedia: Electronic Edition* (*IEEE Computer Society DL*), 8(1), Jan2001, 14-18.
- [2] A. Ginige and S. Murugesan, The Essence of Web Engineering, Managing the Diversity and Complexity of Web Application Development, *IEEE Multimedia*, *Electronic Edition IEEE Computer Society DL*, 8(2), 2001, 22-25.
- [3] Wikipedia Encyclopedia, User Interface Design, *Wikipedia Foundation, Inc.*, page was last modified Apr2008.
- [4] I. Sommerville, Software engineering 7th edition (Pearson Addison Wesley, Pearson education limited, 2004).
- [5] L. Weinman, Designing web graphics.4, the definitive guide to web design and development (Fourth Edition, New Riders, www.Amazon.com, 2003).
- [6] L. J. Najjar, Multimedia User Interface Design Guidelines, *IBM Technical report TR52.0046*, Atlanta, bibliography 59, Sep1992.

- [7] L. J. Najjar, Principles of educational multimedia user interface design, *Human Factors*, 40(2), 1998, 311-323.
- [8] X. Righetti, Study of Prototyping Tools for User Interface Design, Bachelor Thesis, Information System Interfaces, Univ. of Geneva, Jan2006.
- [9] D. Bäumer, R.Walter, H. Lichter, and Z. Heinz, User Interface Prototyping: Concepts, Tools, and Experience, *Proc. 18th International Conf. on Software Engineering (ICSE)*, Berlin, Mar1996, 25-30.
- [10] J. Koke, Top Ten Web Design Mistakes, Knowledge Capital Group Perspective on Marketing and Selling, Jan2001.
- [11] R. Pressman. *Software engineering: a practitioner's approach* (6th edition, McGraw-Hill International edition, 2005).
- [12] M. Jazayeri. Some Trends in Web Application Development, Future of Software Engineering (FOSE'07), *IEEE Computer Society*, 2007.
- [13] T. Vaughan, *Multimedia: making it work* (Sixth Edition McGraw-Hill Osborne Media, 2004).
- [14] M. Theo, User/System Interface Design, Encyclopedia of Information Systems, Academic Press, 4, 2002, 256:1-15.
- [15] J. Antti Hätinen, Extreme Programming and Goal Oriented User Interface Design in Practice, *Research* Seminar on Software Engineering, Univ. of Helsinki, Dept. of CS, Espoo, 2002.
- [16] L. Victoria Edrington, User Interface Design and Usability Testing: an Application, Master Thesis in Information Science, Faculty of School of Information and Library Science, Univ. of North Carolina, Chapel Hill, North Carolina, Apr1999.
- [17] M. Cooper, P. S. Liliana, D. Alexis and S. Paul, User Interface Approaches for Accessibility in Complex World-Wide-Web Applications-an example approach from the PEARL project, 6th ERCIM Workshop "User Interfaces for All", Position Paper, CNR-IROE, Florence, Italy, Oct2000.
- [18] Y. Xiaomei, Web Based CINDI System: Graphical User Interface Design and Implementation, *Master Thesis of Computer Science, Dept. of CS, Concordia University*, Montreal, Canada, 2001.
- [19] V. Balasubramanian, T. Murray, and U. David, A Systematic Approach to Support the Idea Generation Phase of the UI Design Process, *IEEE Proc. 31st Annual Hawaii International Conference on System Sciences*, 1998.
- [20] P. Zhang, V. Ruth Small, Gisela M. von Dran, and Silvia Barcellos, Websites that Satisfy Users: A Theoretical Framework for Web User Interface Design and Evaluation, *IEEE Proc. of 32nd HAWAII International Conf. on System Sciences*, 1999.
- [21] X. Zhang, Xiangshi Ren, and Shijian Lu, a Novel Approach for Web-Based Data Input Panel Design, *IEEE Proc. of 5th International Conf. on Computer* and IT, 2005.

- [22] Gregory Alan Bolcer, User Interface Design Assistance for Large-Scale Software Development, *IEEE Proc. of 9th Knowledge-Based Software Engineering Conf.*, 23, 1994, 142-149.
- [23] L. Wroblewski and E. Rantanen, Design Considerations for Web Based Applications, Proc. of 45th Annual Meeting of Human Factors and Ergonomics Society, Santa Monica, 2001.
- [24] H. G. Sastry and L. C. Reddy, User Interface Design Principles for Digital Libraries, *International Journal* of Web Applications, 1(2), June 2009.
- [25]B. Douglas, Software engineering, a programming approach (3rd edition, Addison Wesley, Pearson education limited 2000).
- [26] J. Garland and R. Anthony, large-scale software architecture, a practical guide using UML (John Wiley and Sons Ltd, The Atrium, Southern Gate, Chichester, England, 2003).
- [27] M. Burstin and M. Ben-Bassat, A User's Approach to Requirements Analysis of a Large Software System, *Proc. ACRE 84 Annual Conf.*, the 5th Generation Challenge, 1984.
- [28]B. Kent, G. James, and C. Robert, Manifesto for Agile Software Development. *Site design and artwork, Ward Cunningham*, 2001. http://www.agilemanifesto.org/.
- [29] D. Cohen, L. Mikael and C. Patricia, Agile software development (a DACS State-of-the-Art Report. Fraunhofer Center for Experimental Software Eng. Maryland and Univ. of Maryland. Data and Analysis Center for Software, New York, Jan2003).
- [30] K. Jonna and R. Jenni, Agile software development in theory and practice (software business program, master thesis, Univ. of Jyvaskyla, Jyvaskyla, 2002).
- [31] O. N. Al-Allaf and K. Saleh, Guidelines for Using Multimedia in Webpage and User Interface Design, Proc. of the 4th International Multiconference on CS&IT (CSIT2006), ASU(private), Amman, Jordan, 3, 2006, 568-577.