International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-1, November 2019

Agent-Based Ergonomic User Interface Development Environment: Analysis Phase

Md. Abdul Muqsit Khan

***Abstract:** Internet around the world has become the prime source to satisfy the requirement of Entrepreneurs, customer and end users. As user comfort is the primary goal of all the entrepreneurs, the end user interacts through the internet to mine the knowledge. The user interacts the internet by apps and web. To attract the users, the user interface (UI) designers focus on Visual presentations to abet users to interact, comprehend, and navigate to the information. The work presents that how visual perceptions of Web page intricacy by perceptive users' behavior, the indispensable cognitive effort for interaction with the UI can be understood. The work presents an extension of current methods for designing UI using model-based technics, with the methods essential for the design of adaptive capacities required in different phases of development. These developments can be fused in Agent-Based Ergonomic User Interface Development Environment. This technique is bolstered by a solution based on Multi-Agent System (MAS) which offer adaptive capabilities to users, designed using the anticipated technique of Agent-Based Ergonomic User Interface Development Environment (AB-EUIDE). In [27] author has presented the various phases of the AB-EUIDE Framework. In this article author presents the analysis phase of the AB-EUIDE and explains how a compromise is done between Adaptation and Ergonomics while designing the user interface.*

***Keywords**: Adaptation, Analysis phase, Ergonomic user interface (EUI), and Multi-Agent System (MAS).*

I. INTRODUCTION

There are anonymous benefits of applying an efficient way to develop any application software. The design of a UI for software is a difficult task [1]. Within the paradigm of AB-EUIDE, which is designed to complement the classical model-based approaches [2, 3], with the features needed for carrying out the adaptation process. Ergonomics is increasingly becoming an investigative topic amongst cognitive science researchers, including linguists [18], and computer ergonomics [21,23, 25,26]

The proposed architecture use MAS. One of the main requirements of the architecture is the need to make decisions about which alternative adaptation is most appropriate.

Revised Manuscript Received on November 30, 2019.
***** Correspondence Author
Md. Abdul Muqsit Khan*, MANUU Polytechnic Darbhanga, Satellite Campus of Maulana Azad National Urdu University, Hyderabad, India. Email: abdul_muqsit_khan@yahoo.co.in, abdulmuqsit@nic.in

In this regard, an architecture based on the concept of agent will be a great benefit to the system, since it allows human decision-making modelling process [5]. MAS also encourages the sharing of responsibilities between different actors, facilitating the current trend of decentralization.

The organization of the paper is as follows: Section two presents analysis phase and domain model of AB-EUIDE. In section three user profiles in the system are discussed. Section four presents a compromise between adaptation and ergonomics. Finally, conclusions and feature direction of the work is presented in section five.

II. PROPOSED METHODOLOGY

During the analysis phase of the AB-EUIDE captures both non-functional and functional requirements, and model the same in a language that does not present ambiguities.

A. Modelling Domain Objects

The domain model consists of a diagram describing those objects that user interacts with the UI to perform the tasks. The two most widely used notation for modelling are entity /relationship and UML class diagrams. In AB-EUIDE proposes the use of class diagrams as a means to represent the domain model, since nowadays design methods are based on object-oriented/Software Composition paradigm, as they are more widespread than the model-based methods entity/relationship. The Software Composition paradigm like aspect oriented, role oriented, agent-oriented modelling and ontology is discussed in [5,6,22].

The domain model is needed for a comprehensive depiction of the data types involving both attributes and methods of classes. The attributes and methods along with interactive tasks; permit users to select the specific objects appropriate for interaction in each case. In this sense, the description of the types listed used greatly facilitates the selection of appropriate interaction concrete objects.

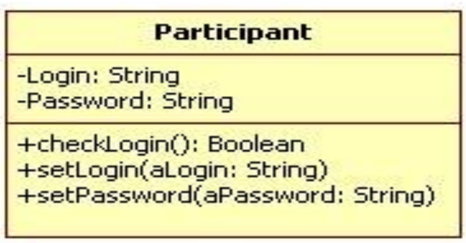

Figure 1. Domain Model of AB-EUIDE.

In Figure 1, exhibits how the user can register in the system; it describes the domain model for the realization of the use case Login identified in the requirements phase [27] of AB-EUIDE. The user is described by a user name (login) and a password. Three actions can be performed on such objects.

checkLogin: Validate the username and password.
setLogin : Assign the value to the user name.
setPassword : Assign the value to the password

III. THE USER PROFILES IN THE SYSTEM

Often, there are different types of people, enjoying different privileges or possessing different characteristics that can be specified statically at design time. For example: a sales application, have different types of profiles (roles) in the system for users. Of course, we have users / buyers, and the store manager. It would not be logical for all user profiles have access to the entire system, example: buyer shouldn't be permitted to access areas of the store management. The system manager should be able to monitor the available stock and the current appearance of the store, thus introducing the concept of inheritance between profiles.

To address the modelling of system profiles and their relationship has chosen a UML class diagram has chosen, for its versatility to represent relationships between entities and especially their ability to express inheritance relationships. Figure 2 describes a possible model profile for the example described above. The concept of Role Oriented Software Composition is discussed in [5, 6, 22].

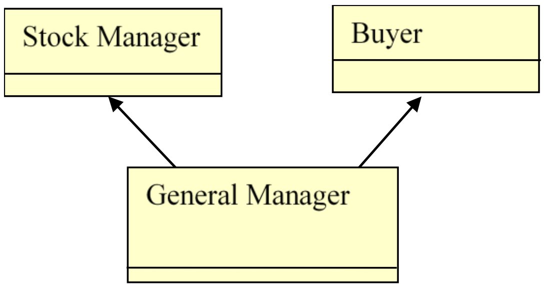

Figure 2. Class diagram for AB-EUIDE

IV. RESULT ANALYSIS

A COMPROMISE BETWEEN ADAPTATION AND ERGONOMICS: When developing the UI for a new application, apply a series of techniques that seek to create UI with a high degree of usability as discussed in [25,26]. Among them we can highlight: The user-centered methods [7, 8], includes user in the design process, allowing validations of different prototypes at different levels by the user. The policy allows reuse of the experience amassed by developers during the development of applications over the years. This experience may be expressed as style guides [9, 10, 11] patterns [12,13] or other methods [14], to apply transformations to convert a model other than or equal to levels of abstraction. These Transformations can be signified by different formalisms, such as rewriting terms XSLT transformation sheets or graph as in [15].

Within an adaptive UI, the UI is evolving, and this evolution will lead to a possible change in ergonomic criteria applied during the design of the UI. However, the evolution of ergonomic criteria of UI cannot be left to chance, but should be possible to define how these might evolve over the adaptation process of a UI, can restrict what type of variations in modified ergonomic criteria are valid and which are not. Example: when a UI undergoes a variation in size of the space where it is being displayed, it is necessary to assess possible adaptations of the UI to the new size. Importantly, the evolution of UI in different ways is subject to the current context.

The different ways UI evolves must be controlled in terms of ergonomic criteria that should be maximized for each platform. It required specification of what criteria should be prioritized, because although ideally all criteria are important, often, increasing in one of them may lead to the reduction of another. For example, if a user wants to maximize the visibility criteria, may be inconsistent with the maximization of accessibility criteria. If the user wants to maintain a large enough font size to improve the UI for vision impaired people will not always be compatible with maintaining the visibility of all elements necessary for the current task. Therefore, it is necessary a priori specification of criteria which should be maximized when an adaptation in response to a change in the current context of use.

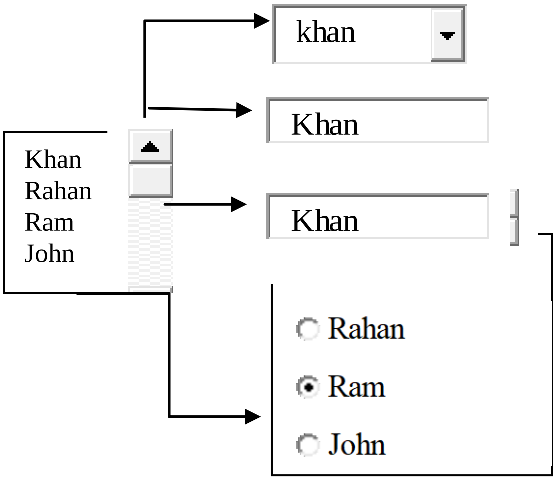

Figure 3. Example of adaptation options depending ergonomic criteria.

Figure 3 describes the different possibilities of adaptation for selecting a list, and the different criteria that affects. Notice how different parameters affect different adaptations, or the same but in different degrees. For example, if the user converts the selection list in a dropdown list, maintaining a high degree of consistency since both elements are handled using similar interaction techniques. However, if it becomes the selection list in a spin, consistency is still acceptable, but lower than in the previous case, since now the interaction technique is a little different, instead of choosing one option among all options displayed simultaneously, the user has to switch from one to another with the buttons above and below the spine. If the user wants to keep the visibility, conversion to a group of radio buttons would be a viable option, since all options are simultaneously visible on the widget as the original. Finally, note as if we choose to select the option by writing in a box, they would be choosing different interaction techniques of the original, and it would be drastically reduced visibility.

The commitment to ergonomic specification enables the modelling of the criteria that the system must continue to choose between different possibilities of adaptation presented.

A. Usability Specification commitment

The specification of the commitment of usability (usability trade-off) involves the description of usability requirements for each platform where the application can be executed. As we have seen, within the Requirements Engineering it pursues to adequately capture the requirements of a system and its transformation into a valid and useful input for the system analysis phase. The Requirements Engineering is responsible for identifying the goals to be accomplished by the future system, the operation of such goals as constraints and services, and to assign the responsibilities for the resulting requirements to agents (software, devices or humans). The system requirements are usually classified into non-functional and functional requirements.

In the present case pursued specifying usability criteria that must preserve adaptivity during the evolution of the UI for its adaptation. A Requirement Goal-Oriented Language (GRL) [16] is proposed, based on the notation I* [17], which allows the designer to capture usability requirements and documents.

B Goal-oriented requirements in adaptive UIs

The specification of the compromise between the several criteria that make up the usability of the future system is performed using GRL notations. In AB-EUIDE describes a commitment to usability for each one type of target platforms. The usability commitment doesn't have to be equal for a desktop computer and mobile device since the characteristics of both platforms are different.

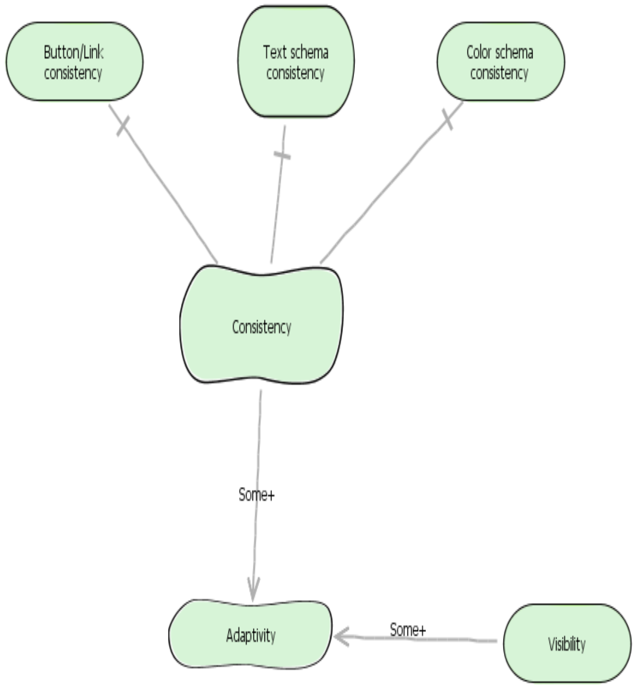

Figure 4. The usability criteria contribute to the main objective: Adaptivity.

Usability criteria are represented by non-functional goals and objectives that contribute to the ultimate goal (the criterion of adaptivity) (see Figure 4). Thus, the priority criteria of usability should be within the future system when the application is represented by adaptation contribution relationships of each one criterion to the general objective of adaptivity. Criteria can be broken down into sub-criteria, for greater precision. Usability criteria that can be evaluated directly be represented using targets, while those depend for evaluation of a set of sub criteria are represented using non-functional objectives. In Figure 4 specifies possible compromise usability. In this case, two objectives contribute to target (non-functional) to achieve adaptivity consistency and visibility. Note how the visibility is specified by a target, the designer believes it is directly measurable, however the consistency is shown as a non-functional goal, as its not directly assessable. For evaluation it's decomposed into three assessable objectives: (1) buttons/links consistency, (2) text outline consistency, and (3) color scheme consistency. Both the consistency criterion and visibility like in this case have the same weight (positive) engagement during the evaluation of usability (some+).

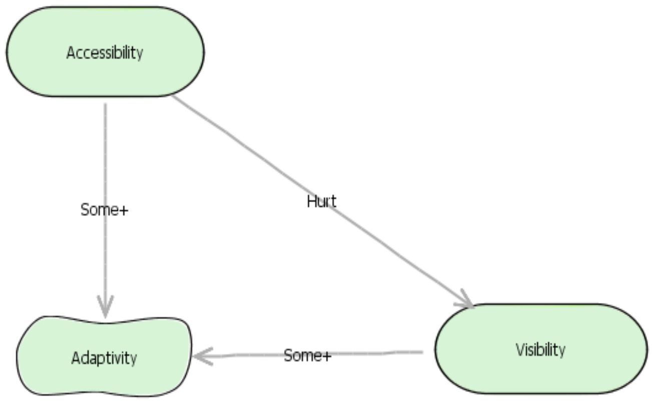

Figure 5. The accessibility criterion has a negative impact on visibility

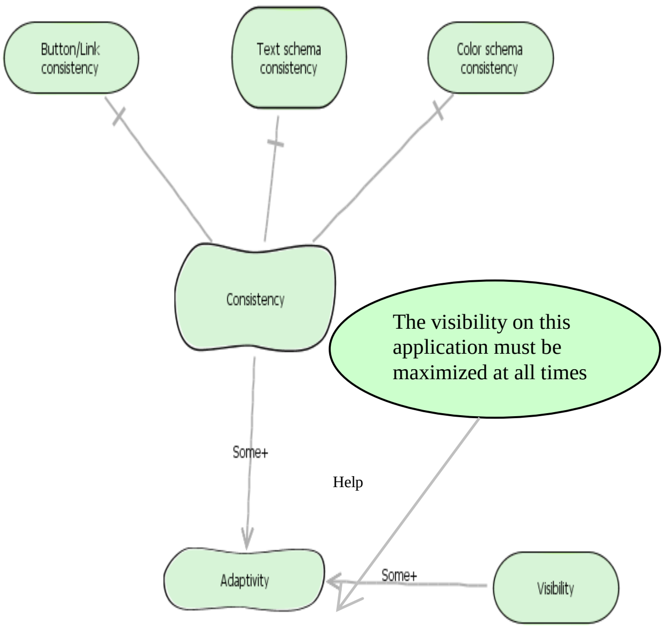

Figure 6. Documentation contribution of beliefs visibility criterion

Note how the visibility is specified by a target, the designer directly measured, the designer can specify how equally influence other criteria using the correlation relationship in GRL (See Figure 5). Relationships correlation equals and unknown types are not considered in the model.

In this way, the designer identifies which are the conflicts between the various criteria that make a commitment to usability. In the example, in Figure5 the designer has specified the criteria of accessibility go against the criterion visibility (hurt). Therefore, when evaluating the visibility, this shall be adversely affected by the value of accessibility obtained. If instead, the designer would specify a relation of the type correlation Help (aid), the value obtained in the evaluation of the visibility has been influenced surely by the value obtained when evaluating the accessibility.

To improve communication between developers and project documentation, the designer can document the decisions taken during the design by adding usability commitment beliefs into the specification itself. Figure 6 has documented the contribution of visibility by a belief.

V. CONCLUSION

The work described an architecture that allows the user to provide a set of adaptation capabilities that are designed using the method proposed. The proposed architecture is built on the notion of an agent, also describes how to carry out each stage of the process of adaptation in MAS. The work has posed an effort to improve software quality, especially an improvement in the quality of ergonomic UI. To do this, in recent years, it has delved to improve the usability of the system on several fronts. In effect, there has been interesting work on improving the usability of user interfaces by incorporating the experience, the design interfaces post-WIMP user methodology for the design of UI solutions, and ultimately improving the ergonomics of the system through the adaptation of the system to the characteristics of the context of use (platform, environment user, & current task) and their changes. This work gives a solution to the design of UIs which intelligently adapts to the context of use while maintaining their usability.

Future directions of the research will focus on

- The Work will also present the remaining phases (Design, and Implementation) of AB-EUIDE.
- The development of the AB-EUIDE for affective computing [19], and social computing [20] for emotional well-being of users can be explored.
- The methods to register the Intellectual property (IP) was identified in [24], an extension of the same will be instigated for EUI.
- The problem of designing an AB-EUIDE for the Internet of things is extremely pertinent and has a lot of unsolved and controversial tasks and decisions.

ACKNOWLEDGMENT

I am indebted to the MANUU Polytechnic Darbhanga, Satellite Campus of Maulana Azad National Urdu University, Hyderabad, India, for providing all facilities for my research work. I am also thankful to professor Chandra Sekharaiah k, for his continuous encouragement and guidance in my research work.

REFERENCES

1. Brad A. Myers, “Why are HCI difficult to design and implement?”, Technical Report, CMU-CS-93-183, Computer Science Dept., Carnegie Mellon University Pittsburgh, July 1993.
2. Francois Bodart, Jean Vanderdonckt, A. Hennebert, Isabelle Sacre, and J. Leheureux, “Architecture Elements for Highly-Interactive Business-Oriented Applications,” In 3rd International Conference on HCI. EWHCI 1993. Lecture Notes in Computer Science, Volume: 753. Springer, Berlin, pp. 83-104.
3. A. Puerta. “A Model-Based Interface Development Environment”. In IEEE Software, Volume: 14, Issue: 4, Jul/Aug 1997, pp. 40-47. DOI: 10.1109/52.595902
4. Michael E. Bratman; “Intention, Plans and Practical Reason”. CSLI Publications, 1987.
5. Chandra S. k, J. Ram, & Abdul Muqsit Khan, “The Peculiarities of Software Composition Models”, J. of Digital Information Management, Volume: 3, Issue: 3, September 2005, pp.181-187.
6. Francois Bodart, J. Vanderdonckt Anne-Marie Hennebert, and J. Leheureux, “Towards a Dynamic Strategy for Computer-Aided Visual Placement,” in Proceedings of Workshop on Advanced Visual Interfaces AVI94, ACM, NY, Jan. 1994, pp.78-87.
7. Donald A. Norman, and Stephen W. Draper, “User centered system design: New perspectives on human-computer interaction,” L. Erlbaum Associates Inc. Hillsdale, NJ, USA, 1986.
8. Md. Abdul Muqsit Khan, and Chandra S.K., “Ergonomics for web applications development,” in Proceedings of 4th Int. Conf., IETET, India, 2013, pp. 62-72.
9. Ben Shneiderman, “Designing the User Interface Strategies for Effective HCI,” 5th edition, publisher Pearson, 2009.
10. Sidney L. Smith and Jane N. Mosier. “Design Guidelines for Designing User Interface Software”. Tec. Rep. MTR-10090, The MITRE Corporation, Bedford, Massachusetts, USA, Aug.1986.
11. International Business Machines Corporation. “Object–oriented interface design: IBM common user access guidelines,” QUE, Carmel, 1992.
12. D. V. Duyne, Jason I. Hong and James A. Landay, “The Design of Sites: Patterns, Principles and Processes for Crafting a Customer-Centered Web,” Addison Wesley Professional Publication. 2003.
13. Jenifer Tidwell, “Common Ground: A Pattern Language for HCI Design”, http://www.mit.edu/~jtidwell/interaction_patterns.html, 1999.
14. J.M.C. Bastien, D.L. Scapin, “Evaluating a user interface with ergonomic criteria,” International J. of HCI, Volume: 7, Issue: 2, pp. 105- 121, 1995.
15. Quentin Limbourg, B. Michotte, J. Vanderdonckt, L. Bouillon, and V.L Jaquero, “UsiXML: a Language Supporting Multi-Path Development of UIs”, in Proc. IFIP Int. Conf. on Engg. for HCI LNCS, Volume: 3425, Springer, 2005, pp. 200-220.
16. www.cs.toronto.edu/km/GRL
17. Eric Yu, “Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering”, In Proceedings of ISRE 97, the 3rd IEEE Intrnational Symposium on Requirements Eng., at Annapolis, MD. USA. 1997, pp. 226-235.
18. Chandra K. S., Md Abdul Muqsit Khan, and Gopal, “Obstacles to Machine Translation,” Int. J. of Translation, Vol. 18(1-2), Aug. 2006, pp. 89-102.
19. Chandra Sekharaiah. K., and Md Abdul Muqsit Khan, “Affective Computing: Next Generation AI Software Systems”, The IUP J. of Information Tech., Volume: 3(4), Decmber 2007, pp. 61-74.
20. Chandra S.K., and Md Abdul Muqsit Khan, “Towards Metrics for Social Computing,” Proceedings of World Academy of Science, Eng. and Tec. 37, January 2009, pp 1086-1090.
21. Chandra S.K., and Md. Abdul Muqsit Khan, “Computer Ergonomics: Relooking at Machines vs. Environment,” The IUP J. of Information Tech., Volume: 2, Issue 3, 2006, pp. 19-27.
22. Md. Abdul Muqsit Khan, and Chandra S.K., “Perspective of MAS Modeling Through Roles and Ontology,” in Int. J. of Recent Trends in Eng., Volume:1(2), 2009, pp. 294-298.
23. Chandra S.K., and Md. Abdul Muqsit Khan. “Computer Ergonomics: Relooking at Machines vs. Environment,” in ch. 21 of Environmental Protection, Daya Publishing House, 2007, pp.104-112.

24. Chandra S. K., and Md. Abdul Muqsit Khan, "The Patents Act and Related Issues: Indian Scenario," in Proceedings of the National Workshop on IPR Curriculum for Eng. and Science Students, IIT Roorkee, India, 2005, pp. 18-19.
25. Md. Abdul Muqsit Khan, "Ergonomic User Interface: System Assessment and Design Process ", Int. J. of Eng. Tech. Management and Applied Sci., Volume: 4(11), Nov. 2016, pp. 79-84.
26. Md. Abdul Muqsit Khan, "Ergonomic User Interface: System Logical Analysis and Design Process," in Int. Journal of Eng. Technology Sci. and Research, Volume: 4(11), Nov. 2016, pp. 64-71.
27. Md. Abdul Muqsit Khan, "Framework for Multi-Agent Based Intelligent Ergonomic User Interface System: Requirements Phase" in Int. J. of Advanced Research in Computer Science. Vol. 8(7), Jul-Aug. 2017, pp. 599-603.

AUTHOR PROFILE

Dr. Md. Abdul Muqsit khan pursed Bachelor of Engineering from Gulbarga University of Gulbarga, India in 1996 and Master of Technology in Software Engineering from JNTU, Hyderabad in the year 2002 and Ph.D from JNTUA, Anantapur, India. And currently working as Principal MANUU polytechnic Darbhanga, Satellite Campus of Maulana Azad National Urdu University of Hyderabad, India since 2009. He is a Fellow of IE, IETE, IRED and ISRD. He is a life member of the ISC, IAENG, IS. He has published more than 20 research papers in reputed international journals, including Thomson Reuters (SCI & Web of Science) and conferences including IEEE, ACM and it's also available online. His main research work focuses on Ergonomics, Ergonomic User Interface, Multi-Agent Systems, Affective computing, NLP and social computing. He has 24 years of experience: 11 years as principal polytechnic, 9 years of teaching experience and 4 years of industrial experience.