Adaptable and Adaptive Human-Computer Interface to Recommend Learning Objects from Repositories

Thomas Quiroz, Oscar M. Salazar, and Demetrio A. Ovalle(✉)

Universidad Nacional de Colombia, Sede Medellín, Medellín, Colombia
{tquirozv, omsalazaro, dovalle}@unal.edu.co

Abstract. In the last decades, some useful contributions have occurred to human-computer interfaces and e-learning system developments such as adaptation, personalization, ontological modeling, as well as, learning object repositories. The aim of this paper is to present the advantages of integrating ontologies as knowledge representation scheme in order to support adaptable and adaptive functionalities that can be offered by a human-computer interface when recommending LOs from Repositories. A human-computer interface model is proposed which is composed of several modules that allow deploying adaptable and adaptive functionalities such as the following: (1) store and retrieving of LOs from repositories, (2) representation of events by learners within the GUI, (3) performing of inferences through ontological reasoned, (4) adaptation of the GUI for each of the users' profiles and (5) monitoring of all changes made by the user on the GUI and storing of them in the system database for further processing. In order to validate the model a prototype was built and tested through a case study. Results obtained demonstrate the effectiveness of the proposed human-computer interface model which combines adaptability along with adaptive characteristics.

Keywords: Human-computer interfaces · Personalization · Adaptable and adaptive systems · Ontologies · Learning objects (LO) · LO repositories

1 Introduction

Nowadays, the access to a large amount of information is easier through the Web that allows data and knowledge repositories to be connected among them. However, it becomes a great challenge for people wishing to learn in a virtual way to assimilate this knowledge since it does not always fit with their learning styles, preferences, or even considering inappropriate ways of deployment of these digital educational resources.

Personalized Adaptive interfaces currently play a very important role in virtual learning environments since they seek to adapt the presentation and display of educational content, such as learning objects (LO), through the preferences, needs and cognitive characteristics of the students. According to Lopez [1] it should be distinguished between adaptability and adaptivity in user interfaces. Within an adaptable interface the user is who explicitly adapts the interface so that it fits their preferences and features. For example, window managers can allow the user to change the settings

P. Zaphiris and A. Ioannou (Eds.): LCT 2016, LNCS 9753, pp. 539–549, 2016.
DOI: 10.1007/978-3-319-39483-1_49

on the appearance of the desktop with respect to colors, fonts, desktop background or behavior of some of its components. In contrast, on an adaptive interface the same system is the responsible actor for activating the actions necessary to perform the adaptation. Thus for instance when a word processor automatically detects a grammatical error, the same processor marks it or even edits it without human intervention. Personalized Adaptive interfaces can then be defined as those parameters of the interface that automatically adapt to the characteristics of the users [2], allowing the improvement of the satisfaction and the permanence of the user interacting with the application on its computer, personal device or on the web site. Current trends are toward the web information retrieval systems allowing adapting results using personalized adaptive interfaces that consider the properties and settings of the users [3, 4].

The aim of this paper is to present the advantages of integrating ontologies as knowledge representation scheme in order to support adaptable and adaptive functionalities that can be offered by a human-computer interface known as GUI (graphical user interface) when recommending Learning Objects (LO) from Repositories. We propose a GUI model and develop a prototype which is composed of several modules that allow deploying adaptable and adaptive functionalities.

The rest of the paper is organized as follows: Sect. 2 presents the conceptual framework of this research. Section 3 reviews some related works concerning adaptable and adaptive interfaces. Section 4 describes the proposed model. Section 5 offers the model implementation and validation of the proposed model. Finally, the main conclusions and future research directions are presented in Sect. 6.

2 Conceptual Framework

This section provides main definitions used in this research work such as adaptable vs adaptive interfaces, learning objects, repositories, ontologies, among others.

2.1 Adaptable vs. Adaptive Interfaces

Human-computer interfaces have evolved in last decades from predominantly textual interfaces to more complex interfaces using multimodal interaction (e.g. communication by natural modes such as speech, handwriting, etc.). In addition, it is important to distinguish between adaptability and adaptivity [5, 6] in graphical user interfaces (GUI). An adaptable GUI allows users to explicitly customize several aspects of the interface so that he/she can fit its preferences and needs [7]. On the other hand, when using an adaptive GUI the system activates itself the actions necessary to perform the adaptation [8].

2.2 Ontologies

Ontologies can be defined as a formal representation of a particular domain using a well-defined methodology that allows the representation of the domain entities and the relationships existing among them [9]. Based on this, it is important to generate a formal representation of the adaptive learning course structure, in order to make inferences and generate recommendations for improving the learning process. Similarly, designing formal representations of a specific domain allows having readable and reusable information for computers and intelligent systems [10].

2.3 Learning Objects, Repositories and Federations

According to the IEEE, a LO can be defined as a digital entity involving educational design characteristics. Each LO can be used, reused or referenced during computer-supported learning processes, aiming at generating knowledge and competences based on student's needs [11, 12]. LOs have functional requirements such as accessibility, reuse, and interoperability. The concept of LO requires understanding of how people learn, since this issue directly affects the LO design in each of its three dimensions: pedagogical, didactic, and technological. In addition, LOs have metadata that describe and identify the educational resources involved and facilitate their searching and retrieval. LORs, composed of thousands of LOs, can be defined as specialized digital libraries storing several types of resources heterogeneous, are currently being used in various e-learning environments and belong mainly to educational institutions [13].

Federation of LORs serve to provide educational applications of uniform administration in order to search, retrieve and access specific LO contents available in whatever of LOR groups [14].

3 Related Works

This section examines some related research works that focus on adaptable and adaptive interfaces seeking to contrast the advantages and disadvantages of each work.

Letsu-Dake and Ntuen [5] design an adaptive interface for controlling a complex system where the user has to monitor an industrial process that has multiple variables. The main function of the adaptive interface is to help the user to maintain proper operation of the system by providing some warnings and alerts, graphics, identification of damaged components, among others. To verify the impact of the adaptive interface and its components authors compare the performance (measured in terms of the times the system needs to reach the goal) of two groups of users, one using the system with the adaptive interface and the other group using the system without adaptive interface.

Park and Han [6] develop a research about the effects of a help provided by an adaptive interface and user control through adapting menus. They use a variety of interface prototypes to validate the impact of the adaptation provided by the system. Each interface is distinguished by the amount of control of the user or system in the adaptation of the interface. Authors measure performance and satisfaction of the user with each of the interfaces. Another measured variable is the time that the user takes to find a specific option; his/her perceived efficiency, and user preferences on interfaces regarding menus.

Shakshuki et al. [7] present a distributed system based on software agents for obtaining and monitoring health metrics in real time. These metrics consider pulse, blood oxygenation or any other metric that can be monitored through the use of sensors. In addition, the system incorporates learning techniques for the deployment of an adaptive interface that accommodates the features of system users (patients or healthcare professionals). Learning techniques allow the system to collect and store historical data of user interaction with the system interface (Human-Computer Interaction). This process is carried out through three sub-components: learning, evaluation, and adaptation.

The learning component categorizes the data collected by the agent's sensor component into two categories: action (choices and preferences) and behaviour (statistical interactions with the interface). The evaluation component compares the new information with historical data and sends the differences to the next component. Finally, the adapting component updates the user model depending on the information received and adapts the interface to user needs.

Jorritsma et al. [8] integrate some mechanisms of adaptive customization to support natural work environment: the Picture Archiving Communication System (PACS) in radiology. The adaptive support is offered in the form of personalized suggestions, generated based on user behavioral data that can be accepted or ignored by participants. The adaptive customization support is designed to help users to effectively personalize the PACS's custom region. It is based on users' function usage, which was logged by the PACS's built-in logging tool, and consist of a table that gives insight into a user's function usage and a set of suggestions about which functions the user should add or remove to his or her custom region. To validate the system 12 Radiologists interacted with it, one half of the Participants received support and the other half did not. Participants who received support used the PACS's customization facilities more effectively than participants who did not receive support.

Ravi et al. [15] develop a system that categorizes MOOCs (Massive Open Online Course) teachers with different computer proficiency using learning authoring environments to provide adapted interfaces. To achieve this, authors categorize teachers in four broad classes by collecting data about teacher's computer performance during authoring process. The four classes are the following: Class A (Expert Users), Class B (Users' with a sound of knowledge), Class C (Knowledgeable intermittent users) and D (Novice users). The system uses this information to specify the design features in order to adapt its interface to teacher's profile. In addition, the system uses ADDIE (Analysis, Design, Development, Implementation, and Evaluation) instructional design model for supporting teachers to complete their authoring effectively in a step by step manner. Experiments made by authors attempt to compare the existing MOOCs authoring environment with their system. Results obtained by analytics showed a better performance among teachers using the system proposed.

Shakshuki et al. [16] propose an architecture of a multi-agent system designed to provide healthcare information about specific patients through continuous monitoring. It is important to highlight that the resulting data produced by the system is accessible not only by the patient to whom it belongs but also by his or her healthcare professional. The proposed system uses an adaptive user interface to improve the overall experience for users with poor vision or motor skills. Authors focus on the implementation of several of the key components involved in the adaptive user interface: learning component and the user model. The system architecture is composed of two kinds of agents: user agent and resource agent. User agents are in charge of three following goals: (1) adapt the UI to improve user experience, (2) manage health data, and (3) respond to healthcare professional user agent requests for health data. On the other hand, the resource agent is responsible to authenticate information requests among user agents, and to archive patient health data for long-term storage. To validate the system proposed two scenarios are provided that demonstrates the feasibility of the adaptive user interface.

From the previously reviewed research works we can conclude that several enhancements can be performed that allow improving the GUI (graphical user interfaces) by adding adaptivity and adaptability features. However, some of these works still present shortcomings such as the following: (1) do not recommend new content; (2) do not allow adapting the panels that contain specific functionalities; (3) do not offer the possibility of finding new educational resources; (4) some of them just adapt menu features; (5) the interfaces are not flexible for adaptation processes, i.e., the system adapts the interface but does not allow that users operate on it. The aim of this paper is to face these shortcomings and to provide a comprehensive solution concerning the use of adaptive and adaptable HCI interfaces in virtual learning environments.

4 Model Proposed

Figure 1 shows the model proposed, which considers five processes that can offer adaptivity and adaptability functionalities within a virtual learning environment, those functionalities will be detailed later. Processes that compose the model are described as follows:

- LO Recovery Process: This process is responsible for the search and compilation of LOs, which takes place from the integration of the system with the LOs repository called ROAP [13]. In this instance both metadata and LO contents are recovered in order to store them in the system central database.
- Grabber Events Process: This process is responsible for monitoring and storing interaction records performed between user and interface, with the aim of collecting those events that user performed when using interface adaptability features. In fact, this information is useful for supporting the intelligent adaptation of the GUI.
- Ontology Generator Process: This process allows the deployment and generation of records within the ontology, from the information concerning user profiles, performed events performed by the user within the GUI, and the interface structure. Subsequently, the generated ontology is delivered to the ontological reasoner. The details associated with the development of the ontology are presented later.
- Ontological Reasoner: is responsible for generating the inferences that allow obtaining relevant information for the interface presentation. The inferred information is sent to the adaptation process.
- Adapter Process: This process is very relevant since it enables the intelligent adaptation of the interface, from the information provided by the ontological reasoner. Adaptations performed to the GUI are presented in the following section.

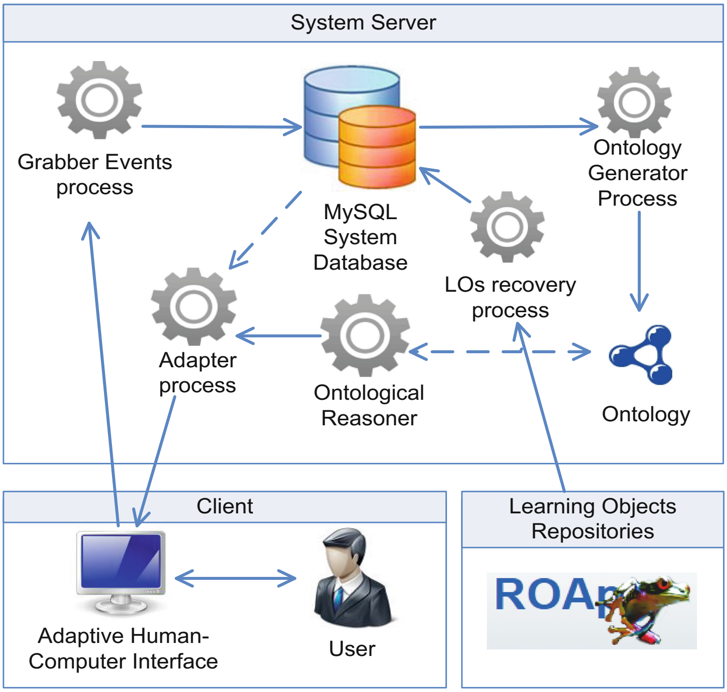

Fig. 1. HCI interface model proposed

Concerning the development of the ontological structure, the following two stages have been considered: the first one examines the characterization of user profiles and the second the structure of the GUI. The student profile is composed of personal information (name, identification, etc.), as well as features and preferences of specific learning process (learning style, font sizes, favorite formats, etc.). The structure of the interface in turn, is divided into five panels that offer the different system functionalities (detailed later). Figure 2 shows the ontology that describes characteristic informations of the GUI panels (height, width and position) and associates it with a specific student profile. The GUI performs this with the aim of storing the display preferences for every student.

The result of applying these two stages allows the generation of inferences from the ontology in order to perform real-time recommendations to users concerning the GUI adaptation. Likewise, the system can infer new kinds of adaptations by detecting new events (interactions) generated by students within the interface. This inferential knowledge abstraction is depicted by using SWRL (Semantic Web Rule Language) rules, some of which are presented in Table 1.

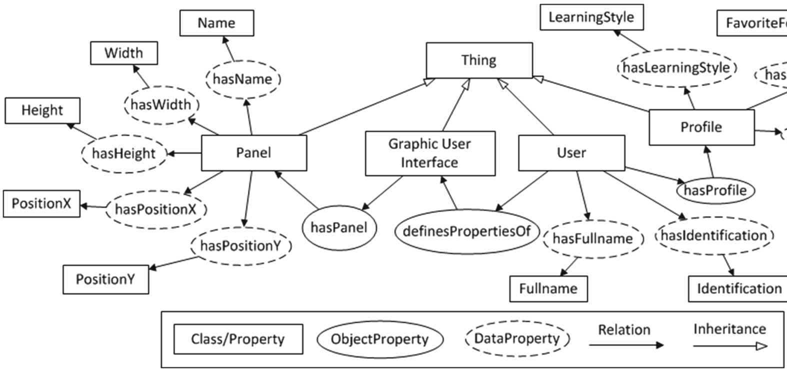

Fig. 2. GUI & User-profile ontological structure

5 Implementation and Validation

The model prototype along with its graphical user interfaces are developed using the Framework GWT-Google Web Toolkit [17]. This framework includes open source libraries that allow web developers to create and maintain complex front-end applications in JAVA.

Table 1. Ontological inference rules

Goal	SWRL Rule
Getting the height and width for a panel, defined by a particular user	PREFIX ont: <http://www.adaptivegui/#> SELECT ?subject ?panel ?height ?width ?name WHERE {?subject ont:hasIdentification 12345 . ?subject ont:definesPropertiesOf ?gui . ?gui ont:hasPanel ?panel . ?panel ont:hasName ?name. ?panel ont:hasHeight ?height . ?panel ont:hasWidth ?width . FILTER regex(?name, "search") }
Getting the position for a panel, defined by a particular user	PREFIX ont: <http://www.adaptivegui/#> SELECT ?subject ?panel ?positionx ?positiony ?name WHERE {?subject ont:hasIdentification 12345 . ?subject ont:definesPropertiesOf ?gui . ?gui ont:hasPanel ?panel . ?panel ont:hasName ?name. ?panel ont:hasPositionX ?positionx . ?panel ont:hasPositionY ?positiony . FILTER regex(?name, "LO Repositories")}

Regarding installation and deployment of the central database we use MySQL database engine for all requirements about relational models that the system needs. For the ontological integration, we use JENA framework [18]. It is important to highlight that the ontology was mapped to OWL language by Protégé framework. As a result, SPARQL query language is used to perform inferences from the ontology. This language is supported by the W3C to perform queries on RDF and OWL graphs, thus enhancing the information search and selection on the semantic Web.

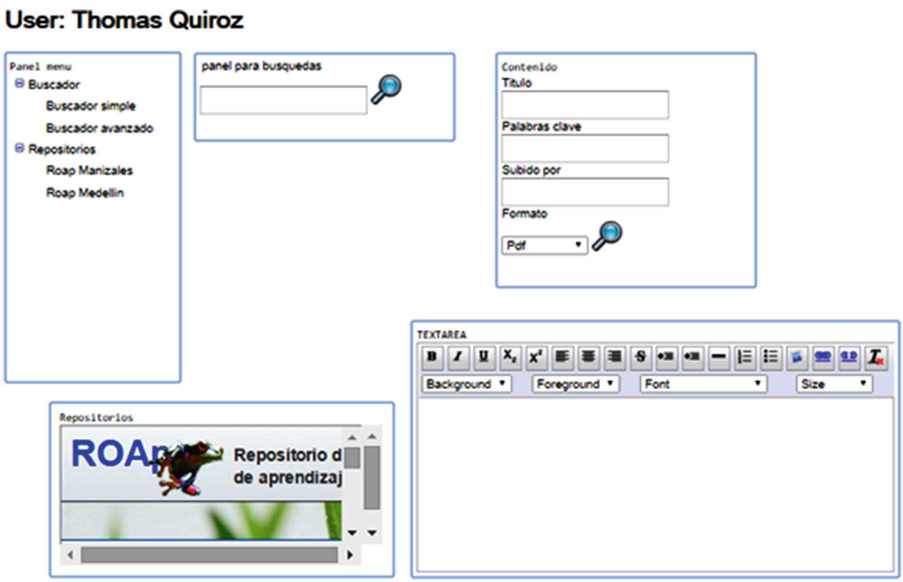

Fig. 3. GUI deployment

User profile is represented through a domain-specific ontology, which also involves the knowledge related to the GUI structure consisting of five panels, as follows:

1. LO Displaying Panel (on the upper right corner of the Fig. 3): This panel is responsible for showing the search engines (Simple and Advanced) and the list of found LO according to the search performed by the user on repositories. After an LO is selected by the system this panel is in charge of deploying it to the user.
2. All the searches performed by a specific user are stored by the system in order to feedback the LO recommendation mechanism. Furthermore, the system adds into the menu panel other objects that may be useful to the user according to his/her learning profile.
3. Menu Panel (on the upper left corner of the Fig. 3): This panel displays a specific user menu that involves a LO search engine (Simple or Advanced); a list of LO repositories, so the student can use them to find more objects that the ones shown by the system; finally, this panel contains a list of relevant LOs according to student learning style and preferences.

4. Search Panel (on the upper center of Fig. 3): It includes the functionalities of the LO simple search engine working like a shortcut mechanism avoiding the use of the menu panel.
5. Notes Panel (on the bottom right of Fig. 3): It enables students to build in real time their own writing notes without moving or change to another program. In addition, this functionality allows exporting these writing notes as plain text files that can be accessed later.
6. LO Repositories Panel (on the bottom left of Fig. 3): It allows a quick access to different LO repositories allowing students personalized searches. This functionality allows to find others LO that cannot be directly found by the system.

An important advantage exhibited by the GUI is the possibility of changing the position and size of each panel (adaptability characteristics). In addition, the GUI can be autonomously adapted to each of the users interacting with the GUI (adaptivity characteristics).

To validate the model prototype, several student profiles are considered and several LO searches are performed using the Colombian LO federation well known as FROAc (http://froac.manizales.unal.edu.co/dnia/main.php). We started with a single user, who applied a test in order to define his/her learning style and thus a user profile is created in the system. The system learn how the user organize size and position of all the panels (depending on the LO type that was displayed). Later, when similar users (i.e. having the same learning style) access the model prototype, it adapts the GUI based on information gathered from previous users that shared similar profiles.

Another case study to validate the proposed model is interacting with an experienced user who already has a lot of time interacting with the GUI and has defined his/her distribution for all panels with different types of LO. To this user the prototype presents a distribution based on all the knowledge it has about the user and all other users with the similar profile and learning style. Then the system assesses whether the user likes the distribution; also whether the user considers well the panel size according to the deployed LO.

6 Conclusion and Future Work

Using the advantages of user profiling in virtual learning environments our proposed model, which combines adaptive with adaptable issues, learns from a specific user-interface interactions in order to propose a similar distribution of the panels that compose the GUI to other users with similar profiles. In addition, the proposed model has the ability to adapt the GUI according to the deployed LO types. This fact facilitates the process of adaptation performed by the user.

However, some recommendations given by some users that used the prototype address following issues that will be considered as future work:

– Simple Search Panels should be removed in order to be added as part of functionalities offered by the content panel wherein the LO are displayed; making in this way the area of the GUI to be expanded.

- The area wherein different panels can move should not be restricted for users. However, since there is a restriction on the graphical libraries used by the system this issue would not be possible to change.
- The model proposed should incorporate user contextual characteristics in order to improve the GUI adaptation mechanism.

Results obtained from case studies demonstrate the effectiveness of the proposed human-computer interface model which combines adaptability along with adaptive characteristics.

Acknowledgments. This research was developed with the aid of the master grants offered to Oscar M. Salazar and Thomas Quiroz by COLCIENCIAS through "Convocatoria 645 de 2014. Capítulo 1 Semilleros-Jóvenes Investigadores". This research was also partially funded by the COLCIENCIAS project entitled: "RAIM: Implementación de un framework apoyado en tecnologías móviles y de realidad aumentada para entornos educativos ubicuos, adaptativos, accesibles e interactivos para todos" from the Universidad Nacional de Colombia, with code 1119-569-34172.

References

1. Lopez, J., Victor, M.: Interfaces de usuario adaptativas basadas en modelos y agentes software. Tesis de Doctorado, Departamento de Sistemas Informáticos, Universidad de Castilla-La Mancha, p. 324 (2005)
2. Gavrilova, T., Voinov, A.: An approach to mapping of user model to corresponding interface parameters. In: Proceedings of the Workshop Embedding User Models in Intelligent Applications, pp. 24–29 (1997)
3. Smyth, B., McGinty, L., Reilly, J., McCarthy, K.: Compound critiques for conversational recommender systems. In: IEEE/WIC/ACM International Conference on Web Intelligence (WI 2004), pp. 145–151. IEEE (2004). doi:10.1109/WI.2004.10098
4. Steichen, B., Ashman, H., Wade, V.: A comparative survey of personalized information retrieval and adaptive hypermedia techniques. Inf. Process. Manage. **48**(4), 698–724 (2012). doi:10.1016/j.ipm.2011.12.004
5. Letsu-Dake, E., Ntuen, C.A.: A case study of experimental evaluation of adaptive interfaces. Int. J. Ind. Ergon. **40**(1), 34–40 (2009)
6. Park, J., Han, S.H.: Complementary menus: combining adaptable and adaptive approaches for menu interface. Int. J. Ind. Ergon. **41**(3), 305–316 (2011)
7. Shakshuki, E.M., Reid, M., Sheltami, T.R.: An adaptive user interface in healthcare. Procedia Comput. Sci. **56**, 49–58 (2015)
8. Jorritsma, W., Cnossen, F., van Ooijen, P.M.A.: Adaptive support for user interface customization: a study in radiology. Int. J. Hum.-Comput. Stud. **77**(4), 1–9 (2015)
9. Tramullas, J., Sánchez-Casabón, A.-I., Garrido-Picazo, P.: An evaluation based on the digital library user: an experience with greenstone software. Procedia Soc. Behav. Sci. **73**, 167–174 (2013). doi:10.1016/j.sbspro.2013.02.037
10. Gaeta, M., Orciuoli, F., Paolozzi, S., Salerno, S.: Ontology extraction for knowledge reuse: the e-Learning perspective. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **41**(4), 798–809 (2011)

11. Rodriguez, P., Salazar, O., Ovalle, D., Duque, N., Moreno, J.: Using ontological modeling for multi-agent recommendation of learning objects. In: Workshop MASLE – Multiagent System Based Learning Environments, Intelligent Tutoring Systems (ITS) Conference, Hawaii (2014)
12. Learning Technology Standards Committee: IEEE Standard for Learning Object Metadata. Institute of Electrical and Electronics Engineering, New York (2002)
13. Rodríguez, P.A., Moreno, J., Duque, N.D., Ovalle, D., Silveira, R.: A model for the semi-automatic composition of educational content from open repositories of learning objects. Rev. Electrónica Invest. Educ. (REDIE) **16**, 123–136 (2014)
14. Van de Sompel, H., Chute, R., Hochstenbach, P.: The aDORe federation architecture: digital repositories at scale. Int. J. Digit. Libr. **9**, 83–100 (2008)
15. Ravi, R., Kumar, A., Bijlani, K., Sharika, T.R.: Self-adaptive interface for comprehensive authoring. Procedia Comput. Sci. **58**, 158–164 (2015)
16. Shakshuki, E., Reida, M., Sheltami, T.: Dynamic healthcare interface for patients. Procedia Comput. Sci. **63**, 356–365 (2015)
17. GWT Documentation (2016). http://www.gwtproject.org/. Accessed 10 Feb 2016
18. Ameen, A., Khan, K.U.R., Rani, B.P.: Extracting knowledge from ontology using Jena for semantic web. In: International Conference for Convergence for Technology 2014, pp. 1–5 (2014)