Proceedings of the 2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE), Larnaca, Cyprus, 11-13 November 2012

A Collaborative Wiki-based Tool for Semantic Management of Medical Interventions

Dionysia Kontotasiou, Dimitrios Zarpalas
Information Technologies Institute (ITI)
Centre of Research and Technology Hellas (CERTH)
Thessaloniki, Greece
{dkonto, zarpalas}@iti.gr

Charalabos Bratsas, Panagiotis D. Bamidis
Medical Informatics Laboratory
School of Medicine, Aristotle University of Thessaloniki
Thessaloniki, Greece
{mpampis, bamidis}@med.auth.gr

Abstract—**Semantic wikis have been widely adopted to support a variety of collaborative activities within the health domain [6], [7], [9]. In this paper, relevant existing tools that may be taken into account for the development of a Wiki-based tool are revisited. The paper then proposes a collaborative Wiki-based tool to be used for semantic management and classification of unstructured and semi-structured medical interventions [12] spread across the Web. The architecture of the tool and its functionality are described in the light of some evidence and a discussion on how this tool may become useful in the semantic Web description of elderly care interventions in the ageing society.**

Keywords: Semantic management, MediaWiki, medical interventions, ontologies

I. INTRODUCTION

This paper describes a Wiki-based tool which provides classification and semantic management of medical interventions [13] that are available on the Web. In order to specify in detail the functionality of the tool, it is useful to have basic knowledge of the state-of-the-art on wikis. The purpose of this survey is to provide the required knowledge on the tools that will influence the design and specifications of the envisioned tool, based on best practices and current implementations and reusing software components or systems if necessary, available as open source products. In order to fulfill the aforementioned purpose, this chapter describes the most representative development platforms for building Wikipedia-like websites.

A. Non Semantic Wikis

Wikis provide a very convenient means for generating informal documentation. They are easy to edit, support both individual and collaborative efforts, and scale extremely well. Most wikis offer at least a title search, and sometimes a full-text search. Wikis are often used to create collaborative websites, to power community websites, and for note taking. Below, two of the most representative wikis are introduced, namely the MediaWiki and PmWiki.

MediaWiki[1] is free server-based software, licensed under the GNU General Public License (GPL). It is designed to run on a large server farm for Wikipedia, a website with millions of hits per day. MediaWiki is an extremely powerful, scalable software and a feature-rich wiki implementation, that uses PHP to process and display data stored in its MySQL database. Pages use MediaWiki's wikitext format, so that users without knowledge of XHTML or CSS can edit them easily. When a user submits an edit to a page, MediaWiki writes it to the database, but without deleting the previous versions of the page, thus allowing easy reverts in case of vandalism or spamming. MediaWiki can manage image and multimedia files, too, which are stored in the filesystem. For large wikis with lots of users, MediaWiki supports caching and can be easily coupled with Squid proxy server software.

PmWiki[2] is a wiki-based system for collaborative creation and maintenance of websites. PmWiki pages look and act like normal web pages, except they have an "Edit" link. That link makes it easy to modify existing pages and add new pages into the website, using basic editing rules, thus the user does not need to know or use any HTML or CSS. Page editing can be left open to the public or restricted to small groups of authors. Some key features of PmWiki are: custom look and feel using skins and html templates, extensibility with hundreds of features (called "recipes") and finally completely self-contained access control system. However, as wikis grow big, retrieving content becomes harder. Text-based search mainly solves content location and more structure is needed. These shortcomings motivated the development of, a number of semantic wikis.

B. Semantic Wikis([8], [11])

Semantic Wikis are connecting established wiki concepts (easy to use and contribute, strongly interconnected, collaborativeness) with semantic technologies (machine processable, data integration, complex queries) by combining social intelligence with artificial intelligence.

[1] http://www.mediawiki.org/wiki/MediaWiki

[2] http://www.pmwiki.org/wiki/PmWiki/PmWiki

978-1-4673-4358-9/12/$31.00 ©2012 IEEE

Goals of Semantic Wikis include:

- simple annotations of existing Wiki content;
- tools that guide users from informal knowledge contained in texts to more formal structures;
- tools for ontology editing where the text is no longer in the focus of the system;
- semantic search queries.

This section includes examples of existing semantic implementations and a brief discussion on them.

Semantic MediaWiki [3] (SMW) is an extension to MediaWiki. It is a semantic wiki engine that enables users to add semantic data to wiki pages. This data can then be used for better searching, browsing, and exchanging of information. Semantic MediaWiki currently is developed very actively. The software is already used on a number of productive installations world-wide [5], but the main target remains to establish "Semantic Wikipedia" as an early adopter of semantic technologies on the web. Among many other sites, SMW also powers semanticweb.org.

OntoWiki[4] is a tool providing support for agile, distributed knowledge engineering scenarios. OntoWiki facilitates the visual presentation of a knowledge base as an information map, with different views on instance data. It enables intuitive authoring of semantic content, with an inline editing mode for editing RDF content, similar to WYSIWIG (What You See Is What You Get) for text documents. It fosters social collaboration aspects by keeping track of changes, allowing commenting and discussing every single part of a knowledge base, enabling to rate and measuring the popularity of content and honoring the activity of users. Furthermore, OntoWiki enhances the browsing and retrieval, by offering semantic enhanced search strategies. All these techniques are applied with the ultimate goal of decreasing the entrance barrier for projects and domain experts to collaborate using semantic technologies. In the spirit of the Web 2.0 OntoWiki implements an "architecture of participation" that allows users to add value to the application as they use it.

II. Analysis and Design

A. System Architecture

The architecture of the tool is illustrated in **Fig. 1** and is mainly intended for doctors. These users may use a Web browser to navigate through the Wiki-based tool user interface and use appropriate semantic forms to communicate with patients and pharmaceutical industries, respectively. The Wiki-based tool is hosted in an execution environment that is supported by a Web server, as well as a data management server (implemented as a database management system, MySQL) that is part of the same environment.

[3] http://semantic-mediawiki.org/wiki/Semantic_MediaWiki

[4] http://ontowiki.net/Projects/OntoWiki

Identify applicable sponsor/s here. If no sponsors, delete this text box. (sponsors)

The tool contains information about diagnoses and medical interventions, ingredients and materials of individual drugs, while data from client tests are also included. The front end side, with which the doctors interact, comprises the Semantic MediaWiki with Halo extension (SMW+).

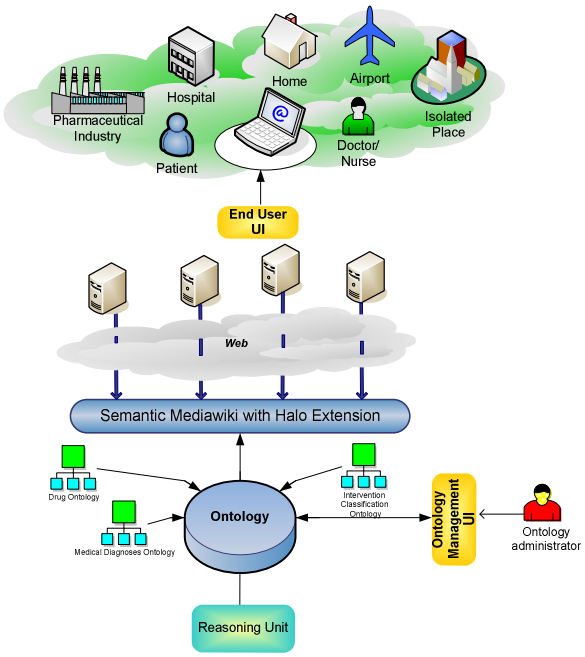

Fig. 1. System architecture

B. Ontology Architecture

The diagram of **Fig.2** describes the structure of the system, showing the ontology architecture upon which the Wiki-based tool is based [10]. The development of the proposed ontology deals with the following concepts (OWL classes) and their corresponding properties:

- Class **MIC** (Medical Intervention Classification): the superclass of all medical interventions to be managed. Each intervention instance is related to appropriate intervention action instances by the hasInterventionAction property. Additionally, each intervention can be related to one or more nursing diagnoses through the needsIntervention object property. It is subclassed to **BehavioralInterventions**, **CommunityInterventions**, **PhysiologicalInterventions**, **FamilyInterventions**, **SafetyInterventions** and **HealthSystemInterventions**. The wiki currently consists of 40 specific interventions.
- Class **BehavioralInterventions**: the superclass of all behavioural interventions, which deals with care supporting psychosocial functioning and changes in lifestyle.
- Class **CommunityInterventions**: the superclass of all community interventions, dealing with health care that affects a larger community.

- Class **PhysiologicalInterventions**: it contains a variety of topical interventions, dealing with areas as activity and exercise management, immobility management, nutrition support, physical comfort promotion and self-care facilitation.
- Class **FamilyInterventions:** it includes interventions for birth and reproductive care.
- Class **SafetyInterventions:** it is about interventions that deal with harm prevention, such as risk management.
- Class **HealthSystemInterventions**: in this class, interventions include proper use of hospital and clinical protocol, proper respiratory and medical history assessment.
- Class **InterventionAction**: the superclass of all intervention actions associated with each intervention through the belongsToIntervention property.
- Class **NursingDiagnoses: it** represents information regarding the type of nursing diagnoses and the necessary parameters for their description.

For the purposes of the ontology development, Protégé has been selected as the authoring tool [1]. In addition, the OWL W3C standard was selected as an ontology language. Finally, with the help of the Gardening Framework of SMW [3], which employs namespaces to distinguish several types of content pages, the ontology was imported in the wiki. OWL classes are represented as MediaWiki categories (pages within the Category namespace); OWL properties do not have a counterpart in MediaWiki, and were introduced by the Semantic MediaWiki extension making available the annotation of above categories; OWL individuals are represented by normal article pages.

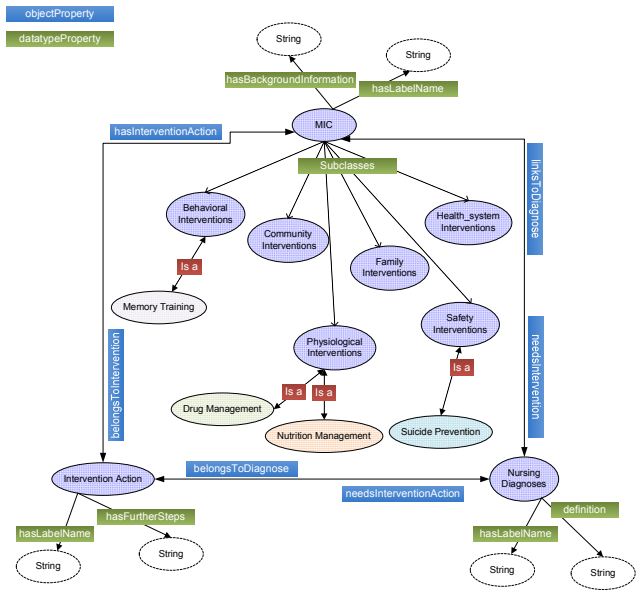

Fig. 2. Ontology architecture

C. *Sequence Diagram*

Important processes have been gathered inside the sequence diagram (**Fig. 3**), which describes completely the main interactive functionality of the system.

Starting from the top of the diagram (**Fig. 3**), an Administrator requests authentication (message 1) from server. The server receives the signal and responses with success signal (message 2), in case the authentication process is successful. The server continuously waits for user signals (messages 3, 5, 7), in order to authenticate them (Registered user 1-n). Registered user 1 decides to create an intervention for a specific domain of interest, and requests from the server the creation of this (message 9). Immediately after, Administrator accepts the intervention (message 10). The intervention is viewable from all the lifelines, hence even a simple user can browse it. Finally, a simple user makes a SPARQL query to the system (message 15) and the system responds with the appropriate results (message 16).

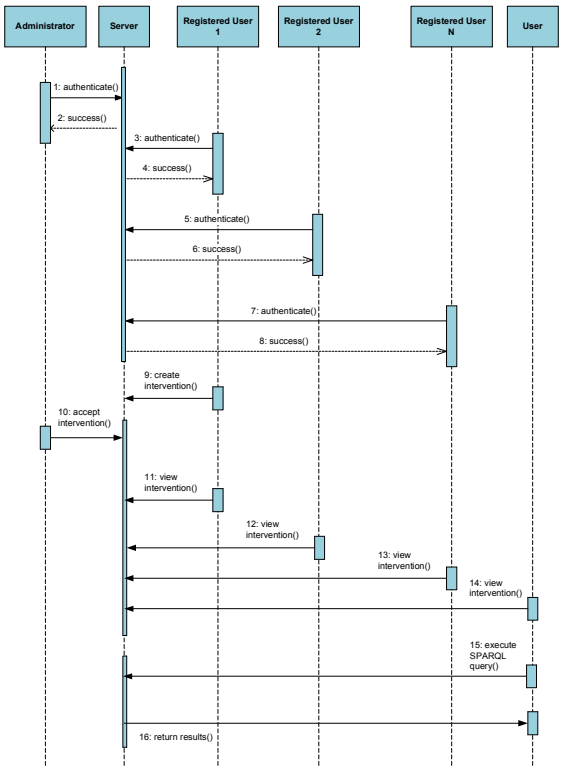

Fig. 3. Sequence diagram

III. IMPLEMENTATION

The system is structured in four layers in order to be modular and extensible. In the following diagram the full system architecture is displayed (**Fig. 4**). All the layers of the system are distinct, with no overlapping between them. The four layers are listed below:

- Modeling layer: This layer is responsible for the modeling of the ontology inside MediaWiki. It consists of MediaWiki categories and properties provided by the Semantic MediaWiki component. A useful tool is Halo component's OntologyBrowser, which helps you navigate to the ontology that has been modeled in the wiki.
- CRUD layer: This layer is responsible for the CRUD (Create, Read, Update, Delete) of the ontology instances. It consists of templates, semantic forms that are being used for handling category instances and consequently class instances of the ontology. Semantic Forms component provides the functionality for Create, Read and Update. The delete functionality is provided from MediaWiki itself.
- Composition layer: This layer is responsible for the composition of the first two layers and the realization of the system's logic. It consists of MediaWiki hooks, ajax calls and system calls.
- Integration layer: This layer is responsible for the retrieval and storage of RDF statements (concerning the ontology) to the relational database. It consists of the SMWRDFConnector component.

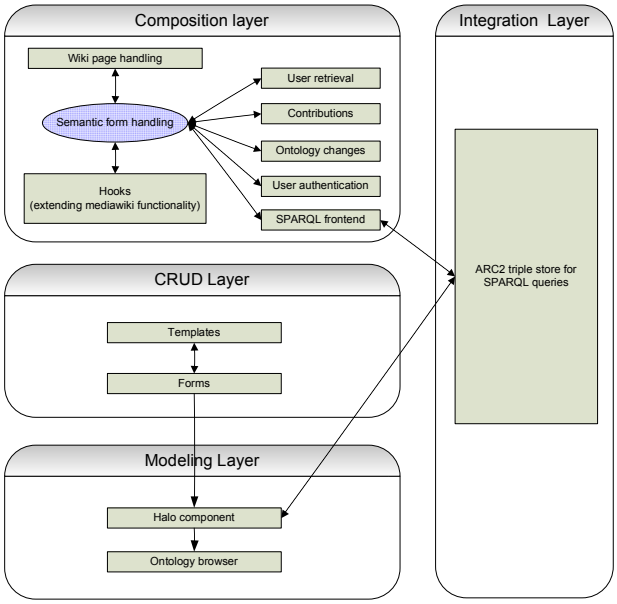

Fig. 4. Layer architecture

A. *Modeling Layer*

In the modeling layer, as mentioned above, the intervention ontology is being modeled inside the MediaWiki. Simultaneously with the ontology import, a set of categories and sub categories enriched with semantic properties are also imported in the MediaWiki environment. This means that pages are added to the Category namespace and to the Property namespace of MediaWiki. There is a correspondence between MediaWiki categories and the ontology classes. For example, the category Intervention_Hierarchy represents the class Intervention_Hierarchy of the ontology. The semantic properties of the Intervention_Hierarchy category correspond to the properties of the class Intervention_Hierarchy. In addition, all the MediaWiki pages, which belong to Intervention_Hierarchy category, correspond to the instances of the Intervention_Hierarchy class. Following the above logic, we created all the categories needed, with all the appropriate semantic properties, in order to model completely the intervention ontology. Using the OntologyBrowser provided by the Halo component, we can see below the modeling of the intervention ontology inside Mediawiki (**Fig. 5**).

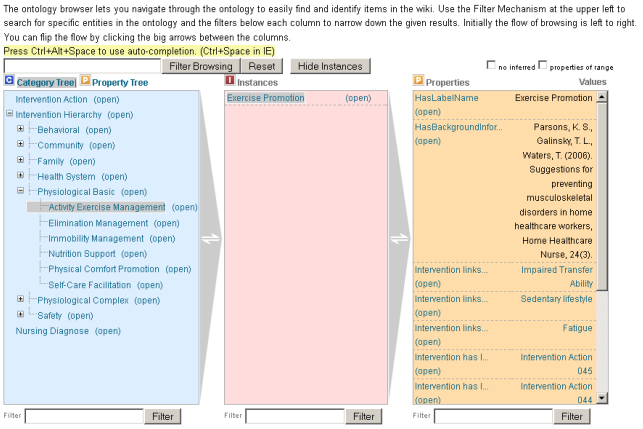

Fig. 5. Ontology inside MediaWiki

B. *CRUD Layer*

The MediaWiki pages correspond to instances of the intervention ontology as we mentioned on the Modeling layer section. The CRUD layer is responsible for the Create, Read, Update and Delete of those pages and consequently of the ontology instances. A set of templates and forms were created in the MediaWiki environment. Thus, pages are added to the Template namespace and to the Form namespace of MediaWiki. The Semantic Forms component allows users to add, edit and query data using forms. It enforces the use of templates in creating semantic data. It does not support direct semantic markup in data pages; instead, the entire semantic markup is meant to be stored indirectly through templates. A form allows the user to populate a predefined set of templates for a page. Templates are standard wiki pages whose content is designed to be embedded inside other pages. Therefore, we have created the templates and forms needed for all category instances and consequently for the ontology class instances. For example, for the "behavior therapy" interventions instances we have created the following template (**Fig. 6**) and form (**Fig. 7**). In the figures below the Halo component's side panel is depicted, which is used for viewing or annotating semantic properties and categories to pages.

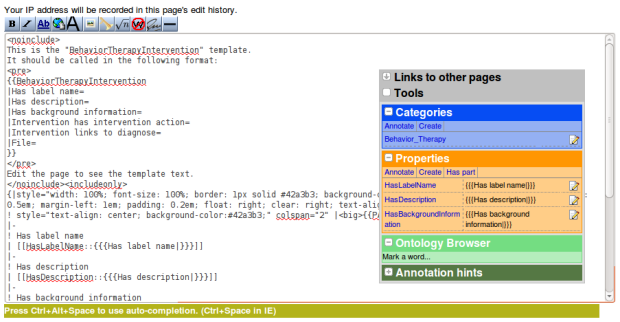

Fig. 6. Behavior therapy intervention template

It should be noted that the semantic properties are defined in the template (**Fig. 6**). The form (**Fig. 7**) utilizes the template in order to render the appropriate input fields in the form interface.

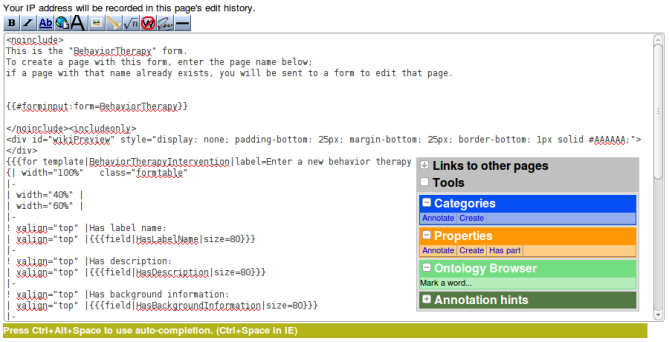

Fig. 7. Behavior therapy semantic form

The Create, Read and Update functions for the category instances are covered by installing all the pairs of template-forms, which the Semantic Forms component can handle. The Delete functionality is provided from MediaWiki itself, because category instances are also MediaWiki pages. Having available the CRUD layer for all category instances, the composition layer can use them in order to apply the functionality.

C. Composition Layer

Mostly, the implementation of the system's requirements belongs in the composition layer, since all the system's logic is realized in this layer. It is called composition layer, because it is the layer that composes all the system elements together and uses the other layers in order to cover system's logic. As we mentioned above, this layer uses MediaWiki hooks, AJAX calls and system calls in order to accomplish its purpose.

MediaWiki provides several hooks that can be used to extend the functionality of the MediaWiki software. Assigning a function (known as an event handler) to a hook will cause that function to be called at the appropriate point in the main MediaWiki code, to perform whatever additional task(s) the developer thinks would be useful at that point. Thus, in the proposed system there are hooks that are responsible for the addition of special pages (e.g NavTree, etc). Special pages in MediaWiki are wiki pages that are created by the software on demand to perform a specific function.

D. Integration Layer

The Integration Layer consists of the Triple Store Connector component that executes a SPARQL query proportionally to the given command and prints the results from the SPARQL query on screen.

IV. CONCLUSIONS

We have used the tool in a testing environment for debugging purposes, but a greater scale evaluation has not been performed yet. We plan such an evaluation in the near future, when a private alpha of the system will be held, providing us with valuable feedback from users. In general, the tool did meet our expectations providing a usable tool for describing interventions. During the development process, we made use of cutting edge Semantic Wiki extensions. Those extensions enabled us to integrate well to our existing infrastructure in relatively short time, and also demonstrated possible uses of technologies like triple store backed Semantic Wikis, which is a very recent and not yet widely applied advance.

Apart from the tool that was developed in this paper, we had the chance to experiment and present the extensions we used and the way they can be composed for the architecture of an extended system. This is the reason why this project could also serve as a demonstration of those technologies and generally for Semantic MediaWiki programming techniques, practices and architectures. It is a field of ongoing research in which advances are recent and examples are still only few. We hope that our research and presentation could be useful for anyone planning to use the same tools, helping her/him integrate this technology for building semantically enabled applications.

For more information, the tool can be found at: http://160.40.50.57/expo/.

REFERENCES

[1] Michael Denny, "Ontology editor survey". Available at: http://www.xml.com/2004/07/14/examples/Ontology_Editor_Survey_2004_Table_Michael_Denny.pdf

[2] Mediawiki Software. Available at: http://www.mediawiki.org

[3] Semantic Mediawiki Extension. Available at: http://semantic-mediawiki.org

[4] OWL language. Available at: http://www.w3.org/TR/owl-features/

[5] Jie Bao, Li Ding and James A. Hendler, "Collective Cognition with Semantic MediaWiki: Lessons and Experiences". 1st ITA Workshop on Network-Enabled Cognition, 2009.

[6] ElstLudger van Elst, Malte Kiesel, Sven Schwarz, Georg Buscher, Andreas Lauer, Andreas Dengel, "Contextualized Knowledge Acquisition in a Personal Semantic Wiki". In Proceedings of the 16th international Conference on Knowledge Engineering: Practice and Patterns, 172-187. 2008.

[7] Matthias Lsbe, M.K., Mycke, R.: Tim, "A semantic web application for the specification of metadata items in clinical research". In: SWATLS, Amsterdam, The Netherlands (2009).

[8] Adrian Groza, Radu Balaj, "Using Semantic Wikis for Structured Argument in Medical Domain". Proceedings of the 3rd International

Workshop on Semantic Web Applications and Tools for the Life Sciences, Berlin,Germany, December 8-10, 2010.

[9] Panagiotis Bamidis, Stathis Constantinidis, Eleni Kaldoudi, Nicos Maglaveras, Costas Pappas, "The use of Web 2.0 in teaching Medical Informatics to postgraduate medical students: first experiences". Proc. of Medicine 2.0 Conference, Toronto Canada, 2008, Appendix in: JMIR 2008;10(3):e22 pp. 22-23.

[10] Charalampos Bratsas, Evangelos Kaimakamis, Vasilis Koutkias, Panagiotis Bamidis and Nicos Maglaveras, "An Ontology-based Approach to Constructing Medical Computational Problems for Use in Electronic Medical Education". Proc. of 20th IEEE International Symposium on Computer-Based Medical Systems, Maribor, Slovenia, 2007, pp. 669-674.

[11] Charalampos Bratsas, George Kapsas, Stathis Konstantinidis, Gregory Koutsouridis, Panagiotis Bamidis, "A Semantic Wiki within Moodle for Greek Medical Education". Proceedings of CBMS 2009: The 22nd IEEE International Symposium on Computer-Based Medical Systems, Special Track: Technology Enhanced Learning in Medical Education, Albuquerque, New Mexico, USA, August 3-4, 2009

[12] Iowa Intervention Project (1996). Nursing Interventions Classification (NIC) (2nd ed.), St. Louis: Mosby-Year Book.

[13] Nursing Interventions Classification