CLINICAL PRACTICAL GUIDELINE EDITOR
Clinical Practical Guideline-based Decision Support Tool

Plavčak Aleš
MIPS programska oprema d.o.o., Cesta k Tamu 7, 2000 Maribor, Slovenia

Keywords: Decision-support systems, medical informatics, clinical practice guidelines, knowledge modelling.

Abstract: According to quick growth of information and complexity of medicine, the development of informatics in medicine is in full bloom. Medical decision support systems have been developed to help choose the appropriate medical treatment procedures, ensure the quality of health care and enable the control of resources. Clinical Practical Guidelines have greatly contributed to the accelerated development. Many different modelling methods and tools have been developed for executing guidelines. Here, the three of many applicable guideline modelling techniques are represented in greater scope. Also, a new technique of representing the clinical knowledge has been designed, taken from the studies of already existing models. The implementation of the application for editing, browsing and executing clinical guidelines has been implemented as well. The application is capable of generating recommendations for a specific clinical state and evaluation of the already existing health care process. This paper covers the general presentation of informatics in medicine and the techniques for modelling of medical knowledge which nowadays represents a gateway for prosperous development, and paves the way for broader use and implementation.

1 INTRODUCTION

The experiences of individual physicians, opinions of professors in medical schools, medical textbooks, clinical journals and clinical trials have guided the practice of medicine for most of this century, so the standardization is not necessarily obligatory, but strongly recommendable. The medical community has always standardized medical care to some degree in order to provide what it thought was the best care, to efficiently use resources, to satisfy patients, and to withstand third party scrutiny.

The idea of studying which treatments work best is nothing new, although systematic treatment procedures are a novelty in many branches of medicine. It is well known what the right things to be done are, but we have to make it happen, so here is methodology and computer based decision support systems to assist practitioners and patients in making decisions about appropriate management of specific clinical conditions.

Our use of computers has been driven not only by the increasing need to manage large amounts of information, but also by the imperative to make evidence based and cost effective decisions on a daily basis. Furthermore, there is accumulating evidence to prove that computer aided medical tools address the growing information needs of the busy clinician and improve healthcare processes as well as patient outcomes. In turn, this has led to the rapid proliferation of a variety of clinical decision support system (CDSS). A computerised CDSS is a computer based tool using explicit knowledge to generate patient specific advice or interpretation. It is now universally agreed that conforming to state-of-the-art guidelines are the best way to improve the quality of CDSS.

Nowadays the basic of developing a CDSS is a clinical practice guideline (CPG) that is a subject discussed by number of researchers who are trying to develop different technologies for delivering computerized guidelines in clinical care. The new research movement could revolutionise the health care industry by improving quality and reducing costs, say the experts that are studying, developing and evaluating CPGs. Above is depicted a newly proposed design and implementation of application for modelling, executing and evaluating CPG.

Aleš P. (2008).
CLINICAL PRACTICAL GUIDELINE EDITOR - Clinical Practical Guideline-based Decision Support Tool.
In *Proceedings of the First International Conference on Health Informatics*, pages 112-118

2 KNOWLEDGE MODELLING

Medical experts are increasingly expected to always make the best decision. This is difficult. The amount of medical information in the world is increasing, yet then capacity of a human brain is not. Computers have the ability to help deal with all this information, so generally speaking, computers are better than humans at managing loads of information and solving complex problems.

To be effective, these tools should have access to the patient's medical record, use standard medical vocabularies, have clear semantics, and facilitate knowledge maintenance and sharing. In addition to that they need to be sufficiently expressive in order to explicitly capture the design of the rational process and outcome intentions of the guideline's author, while at the same time they must enable the attending physician to use their own preferred methods to achieve a certain degree of flexibility during the application runtime.

The level of standardization in today’s guidelines is relatively new. When predicated on sound medical and scientific data, these guidelines can lessen provider variability in treatment and diagnosis. Better standardization allows better measurement of resources used and assessment of benefits obtained. These guidelines can be particularly effective when applied to high-prevalence, high-cost diseases or conditions.

2.1 Clinical Practice Guidelines

CPGs are developed to reduce inappropriate variations in practice, to improve health care quality, and to help control costs. Although the importance of guidelines is widely recognized, health care organizations typically pay more attention to guideline development than to guideline implementation for routine use during care process.

The American Medical Association calls them “practice parameters” and defines them as “...strategies for patient management developed to assist physicians in clinical decision making.” As already mentioned, they should not be rigid and static; rather, they should be flexible and dynamic road maps aimed at reducing clinically significant and unexplained variations in patient care process.

“CPG are systematically developed statements, based on best evidence, intended to assist practitioners and patients in making decisions about appropriate management of specific clinical conditions” (Institute of Medicine, 1990). This definition emphasizes the decision-making aspect of clinical practice guidelines.

Samson Tu and Mark Musen have identified five principle tasks that computerised guidelines and guideline representation methods should be capable of supporting: making decisions, sequencing actions, setting goals (e.g. specific patient states) to be achieved, interpreting data, refining actions (i.e. breaking up into sub-components).

But not all of them are able to fulfil all the principles. They cover many methods and demonstrate the use of different representation formalisms and computational techniques.

2.2 Guideline Modelling Methods

Based on a literature search of computer-based guideline specific representation models, three published research projects were included in this review and represent the base for developing a CPG support tool. Arden Syntax was chosen for its simplicity and represents a pioneering achievement for guidelines. PROforma is an easy understandable and flexible language for encoding medical knowledge. GLIF represents a complex guideline modelling method and has well defined object-orientated design.

2.2.1 Arden Syntax

Arden Syntax is a standard, formal procedural language that represents medical algorithms in clinical information systems as Medical Logic Modules (MLMs), and uses rule-based specification for encoding medical knowledge. It is the first standard for representing medical knowledge. An MLM is a hybrid between a production rule (i.e. an "if-then" rule) and a procedural formalism. Each MLM is invoked as if it were a single-step "if-then" rule, but then it executes serially as a sequence of instructions, including queries, calculations, logic statements and write statements.

Arden was developed for embedding MLMs into proprietary clinical information systems. It was specially designed to support clinical decision making. An individual MLM should contain sufficient logic to make a single medical decision. Sequencing tasks can be modelled by chaining a sequence of MLMs. MLMs have been used to generate clinical alerts and reminders, interpretations, diagnoses, screening for clinical research studies, quality assurance functions, and administrative support.

2.2.2 PROforma

PROforma is a formal executable logic language for describing clinical and other processes in terms of the decisions and other tasks that a physician needs to carry out to achieve its goals. It is capable of capturing the structure and content of a CPG in a form that can be interpreted by a computer. The language represents the basis for a method and a technology for developing and publishing executable CPGs. PROforma combines the features of formal specification languages as known in software engineering with the features of knowledge representation languages as known with artificial intelligence.

The PROforma language structure is based on a simple but versatile clinical process model known as the *domino model* shown in “Figure 1”. This model derives from a variety of empirical studies of clinical decision-making and the development of aids to support patient management.

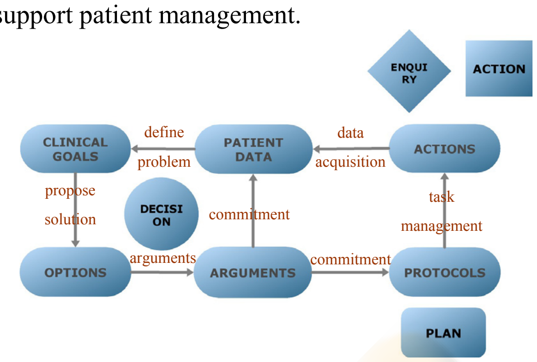

Figure 1: The relationship between the domino model of clinical process and PROforma tasks.

The left side of the diagram represents decision-making and the right the plan enactment. Given a set of beliefs, an agent may set certain goals and various solutions to these goals. With multiple options, such as alternative diagnoses or treatments, the agent must consider the arguments for and against these alternatives and make decisions based on the validity of each of the arguments. The domino model describes a relationship between actions, decisions, beliefs, plans, goals and candidate solutions, and the inference and processes linking them.

A further result was a reconstruction of the domino model into a minimal set of executable generic tasks: enquiries, decisions, plans and actions as shown in “Figure 1”. Tasks are formal software objects that can be composed into networks representing CPGs or other processes, and it is from these tasks and the logical construct associated with each task, that the PROforma language is derived.

2.2.3 GLIF

GLIF is a computer-interpretable language for modelling and executing clinical practice guidelines. GLIF supports sharing of computer-interpretable clinical guidelines across different medical institutions and system platforms. It has a formal representation and defines the ontology for representing guidelines, as well as a medical ontology for representing medical data and concepts.

GLIF2 enables guideline modelling as a flowchart of structured steps, representing clinical actions and decisions. GLIF’s guideline class also specifies maintenance information, the intention of the guideline, eligibility criteria, and didactics. The GLIF guideline instance syntax, which was based on a separately developed language, specifies the format of text files which contain GLIF-encoded guidelines. These files are used for sharing and interchange. However, the attributes of structured constructs are defined as text strings that can not be parsed, and such guidelines can not be used for computer-based execution that requires automatic collaboration.

In the year 2000 a new version of GLIF was introduced (GLIF3), an evolving revision of GLIF that attempted to overcome several of GLIF2’s limitations. The GLIF3 model is object-oriented. It consists of classes, their attributes and the relationships among the classes which are necessary to model clinical guidelines. The model is described using Unified Modelling Language (UML) class diagrams.

2.2.4 Approach to Guideline Modelling

We were trying to examine the increasing sets of resources to obtain sufficiently amount of knowledge that is needed to design development tools and technique for building healthcare application that comply with the highest possible quality, safety and ethical standards.

To build an effective tool for capturing medical knowledge in a systematic and executable way among other criteria the following should be satisfied: access to the patient's record, use of standard medical vocabulary, clear semantics, knowledge maintenance and sharing, sufficient expressiveness to explicitly capture clinical processes of the guideline's author and leave flexibility during application runtime.

The primary goal of this project was to design an open source decision support application for decision-making between health professionals and patients.

2.3 Guideline Evaluation and Design

We have designed a machine interpretable guideline model as a sample of application that can improve inappropriate variations in practice, health care quality and to help control the costs. The guideline model was preceded by the profound study of the above mentioned guideline modelling methods. It includes an analysis of the literature and three published research projects concerning computerized specific representation models. We have tried to overcome the problematic of representing clinical knowledge in computerised manner by modelling clinical knowledge with CPGs. In achieving our goals we have followed the Stanford team dimensions: organization of guideline plan components, goals/ intentions, model of guideline actions, decision model, expression/ criterion language, data interpretation, medical concept model, patient information model.

These dimensions capture the essence of modelling the logic of computer-interpretable guidelines. The first four represent the core guideline components, and the last four link the guideline model to the patient data.

For identifying key elements of our CPG model the following elements were chosen:

Action specifies clinical actions that are to be performed during the patient's care process. These clinical actions can include diagnostic or therapeutic procedures that need to be performed in a treatment process, if an action step is triggered. Actions were modelled merely in a descriptive manner.

Enquiry describes the patient's temporary physical state. It contains the list of the attributes which are necessary for evaluating the patient at the temporary point of treatment. It looks into a patient's medical record and retrieves the values of the attributes described in the enquiry list, or asks the user for the required data.

Decision controls the flow of a guideline. It contains a group of candidates and a group of arguments for an individual candidate. Arguments are described by given weight. It supports inference in propositional and predicate logics, together with certain non-classic logic for reasoning and control of the guideline flow.

Plan components represent actions, decisions, enquiries, or hierarchical decomposed sub-plans of the guideline and their relationships. Plans merge together individual atomic tasks (decisions, actions and enquiries) or sub-plans into logical groups, control complexity and enable control of grouped elements.

Basic tasks that a guideline needs to satisfy are shown in "Figure 2". Data needed at the specific point of guideline execution is acquired by Enquiry. Based on defined values we can make decisions that evaluate patient's temporary physical state. Actions are defined for specific patient state, and are executed as needed.

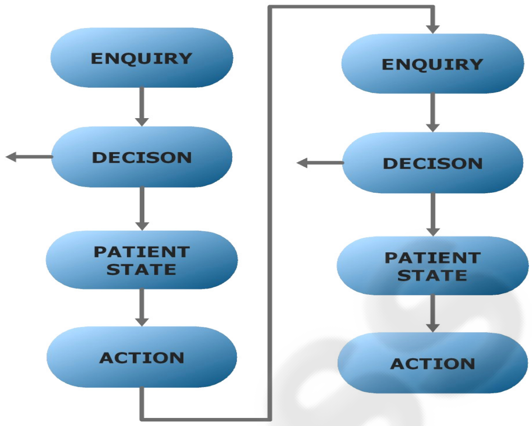

Figure 2: Logical flow of a guideline.

Constructs organization and decision model clearly defines goals of guideline and appropriate actions. The goal of a guideline flow is to choose an appropriate medical procedure for treatment of specific patient's state, by either excluding possible variations, or by choosing an optimal treatment process. Our decision support model supports the following features: each task can interrupt the guideline execution, some tasks are optional and can be ignored at execution time, some tasks need user interaction and have to be confirmed, all tasks can define preconditions that have to be fulfilled in order to execute that task, execution of each task brings system into a new state and sets new conditions, all tasks can have cycle execution, decision model can execute alternative tasks in parallel by excluding preconditions of each task, decision model is non-deterministic and uses arguments rules that can choose more than one candidate among available alternatives, expression language is represented using mathematical expressions which, when evaluated, give logical result *True* or *False*.

A guideline can be viewed in two abstract levels:

Conceptual level represents guideline as a flow chart, and is used for browsing and navigation through constructs. It represents a clear and simple overview of a guideline.

Executable level represents guidelines in full consistency and completeness. Definitions of enquiries items, clinical action and flow of the algorithm are specified at this level.

2.3.1 Database Support

Our model uses a support of relational database that is easy extendable and enables transformation or different representation of data. We have built an entity relational database diagram (ERD) with minimal set of entities as shown in “Figure 3”. Starting-point of a guideline is the “*G*LGuideline” entity that describes general information about intention and its temporary state. For the moment we have four general tasks of guideline, the common attributes of which are collected in the “GLSteps” entity. Future extensions with additional type of tasks are easily deployable. Decision specific attributes are described with candidates and pertaining arguments. Enquiry specific attributes are described with predefined attributes and their pertaining values. All schedules that represent possible selection are modelled with separate entities.

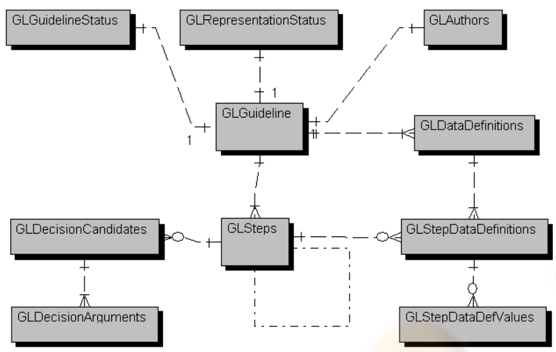

Figure 3: ERD diagram.

2.3.2 UML Design

We present a conceptual object oriented design of our application that capture the structure of guideline-based decision support tool in “Figure 4”. The “DecisionProcessor” is a controller class that represents business logic and enables iteration through guideline steps. For evaluation of our expressions parameters and arguments are needed. “Candidate”, “Argument” and “Parameter” are persistent classes. Postfix calculator can evaluate an expression with given arguments and parameters. The “DecisionStepState” enables the monitoring of each state of execution. Chosen actions and evaluated decisions are represented with classes “Action” and “Decision”. In this sub-section a brief description of actual design is stated, so basically a more complex model was implemented.

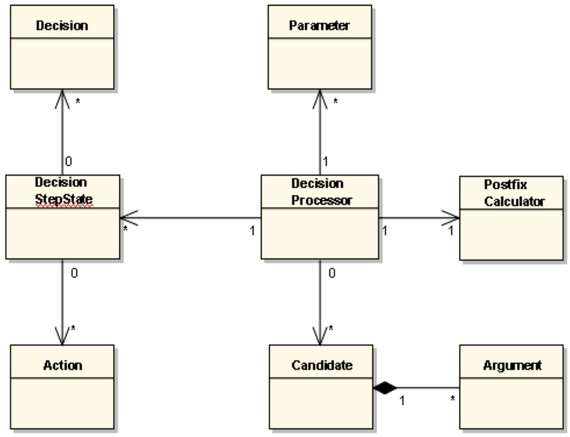

Figure 4: Conceptual class diagram.

2.3.3 Decision Model

The guideline modelling methods use a variety of decision models. The decision models ranges from simple if…then….else or switching constructs to complex models such as decision trees. Our model uses a decision tree, and selection among elements is modelled by mutually exclusive preconditions that depend on result of a decision task. The model uses argumentation rules for/against/choice for selection among alternatives. A none-deterministic choice is possible, where more than one alternative may be justifiable for a patient.

Expression language that defines preconditions, post conditions, cycles and candidates is defined as a complex expression containing operators, functions and variables. Operators are arranged into formula lexicon, term lexicon and unary lexicon. Each of these describes the operators available at certain parser priority. We have included 12 function lexicons which cover a wide range of standard functions, string functions, and other complex functions. Also, 2 operator lexicons are included. We have implemented more than 20 operators and more than 200 functions in an expression parser that is easily extendable with additional lexicon, or a new operator or function.

2.3.4 Constructing Guideline

We have implemented an information system based on CPG for modelling clinical knowledge. The result of this implementation is a graphical composer for creating machine interpretable guidelines. Those guidelines are fully implemented on executable level and can be executed to obtain appropriate clinical actions. The main window of the guideline editor is shown in “Figure 5”. Top node of the tree represents a main task that specifies information unrelated to the health knowledge, and is used for maintenance, change control, version control and other basic information of the entire guideline.

The enquiry element is used for acquisition of attributes that are needed at execution level of a guideline. Attributes are described as optional or mandatory, and their selection mode can be single or multiple. They can be predefined, and each of them ca be encapsulated within a specific attribute type.

Action task of a guideline is an important aspect, but in this project it has been modelled as simple textual description, and it is anticipated to be executed outside of the information system.

Decision uses a well defined decision processor with a rich mathematical expression language. Augmentation rules are used to choose one or more appropriate candidates. For each of them, we can specify a multiple arguments that, when evaluated, give logical result *True* or *False*. Each argument has a given weight. A selection control can be set to numeric mode or selective mode with the following rules:

- Rules that strictly exclude the alternative
- Rules that argue against the alternative
- Rules that argue for the alterative
- Rules that confirm or express strong preference for the alternative

A decision can also act as enquiry, in case it requires additional data for its execution.

Execution of tasks depends on their predefined precondition. Each executed task can inject a new condition into algorithm flow.

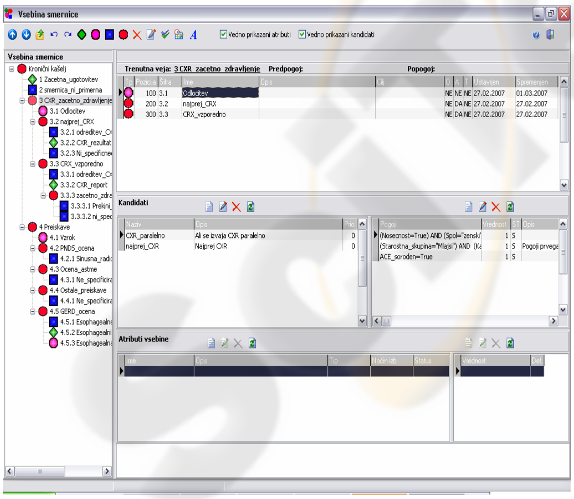

Figure 5: Guideline composer.

3 EXECUTING GUIDELINES

This subsection represents execution of guidelines modelled with our guideline composer. As a test of guideline execution, two guidelines were composed and executed for the purposes of testing. The first was treatment of chronic cough taken from the studies of PROforma, and the other one the determination and classification of hypertension disease taken from textual format using a book “Evidence based guidelines” and transformed into executable form.

At this point a guideline is represented at the executable level. The guideline execution is shown in “Figure 6”. The left side of the picture shows a guideline content in execution process which provides the overview about which elements are chosen and which rejected. A user can also see a diary of events and check guideline for errors. During the runtime some occurrences need a user conformation which enables an interaction in guideline execution process. Also some tasks are optional and are to be confirmed or rejected. The left-top part of the “Figure 6” shows guideline attributes and their values which are valid at a moment. Also, chosen decisions and their evaluated candidates are shown for each execution step. In the right-bottom part appropriate actions are shown, together with their description and intention.

The intention of executing CPGs with this application is to obtain regularity of CPGs or to confirm already valid health care process.

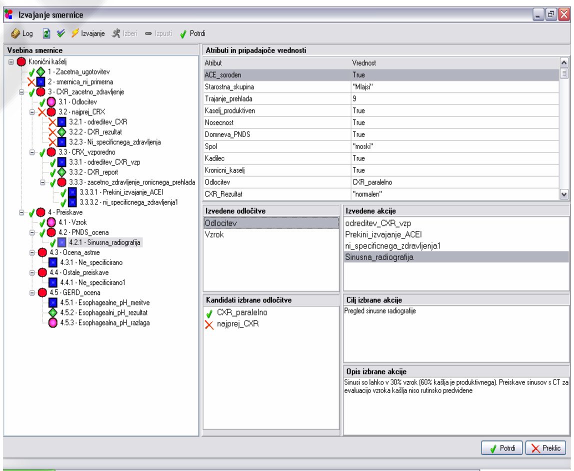

Figure 6: Guideline execution.

4 CONCLUSIONS

Growth of information about appropriate clinical treatment is enormous and makes its appropriate use in practice impossible. The purpose for building decision support systems for treatment process is to enable easy access to clinical knowledge. That is what should be the lead force for developing an appropriate and standardized CDSS.

By studying some of the many methods for representing clinical knowledge, guideline modelling and execution tools were developed. One of the most important aspects when developing a decision support tool is sharing information among other institutions which leads to a need to build a centralized data storage. For this purpose a relation database model has been developed and implemented. Model is flexible and fully extendable for further development. Application itself uses a multi user application server that enables sharing medical knowledge among users and institutions. For building a clinical guideline with composer a simple and easy understandable guideline constructs were implemented that are understandable to a person with none or poor computer knowledge.

Design of the application is object orientated and, if needed, extendable with other construct. Expression/criterion language uses a reach postfix mathematical parser. Among many already implemented operators and functions, it is possible to develop as many as needed user defined lexicons and inject them into the parser. This leads to a very flexible and adaptable expression language that can be used for complex decision making.

A newly proposed design represents an innovation in that it uses relational database support and a reach mathematical expression language parser which enables an infinitive and complex decision modelling.

For now, the application's primary goal is to build clinical practice guidelines and execute them in patient care process in order to obtain recommendable actions. Further development could lead to inductive learning, the statistical evaluation of effectiveness and appropriateness of guidelines by testing their regularity in a specific care procedure.

REFERENCES

Wyatt, J., Spiegelhalter, D. 1991. Field trials of medical decision-aids: potential problems and solutions, *Proceedings of the 15th Symposium on Computer Applications in Medical Care*.

Osheroff, J., Bankowitz, R., 1993. Physicians' use of computer software in answering clinical questions. *Bull Med Library Association*.

Hunt, D., Haynes, B., Segal, H., et al., 1998, Effects of computer-based clinical decision-support systems on physician performance and patient outcomes, JAMA, 280:1339–46.

Fox, J., Bury, J., Humber, M., Sutton, D., Integrating Bayesian inference into the PRO*forma* language, *Advanced Computation Laboratory, Imperial Cancer Research Fund, London*

Bury, J., Fox, J., Sutton, D., The PRO*forma* guideline specification language: progress and prospects.

Peleg, M., Boxwala, A.A., Ogunyemi, O., Zeng, Q., Tu, S.W., Lacson, R., Bernstam, E., Ash, N., Mork, P., Ohno-Machado, L., Shortliffe, E.H., Greenes, R.A., GLIF3: The Evolution of a Guideline Representation Format,1 *Stanford Medical Informatics, Stanford University School of Medicine, Decision Systems Group, Harvard Medical School, Brigham & Women's Hospital, Department of Medical Informatics, Columbia University*

Sordo, M., Ogunyemi, O., Boxwala, A.A., Greenes, A.R., Tu, S.W., GELLO: An Object-Oriented Query and Expression Language for Clinical Decision Support, *Decision Systems Group, Brigham & Women's Hospital, Harvard Medical School, Section of Medical Informatics, Stanford University School of Medicine, Stanford.*

Peleg, M., Boxwala, A., Modeling Clinical Guidelines in a Sharable and Computer-interpretable Way: Development, Implementation, and Use Requirements Instructors, *Stanford Medical Informatics, Stanford University School of Medicine, Decision Systems Group, Harvard Medical School, Brigham & Women's Hospital, Boston, MA.*

Tu, S.W., Campbell, J., Musen, A.M., The Structure of Guideline Recommendations: A Synthesis, *Stanford Medical Informatics, Stanford University School of Medicine, Stanford, University of Nebraska Medical Center, Nebraska Health Systems, Omaha*

Plavčak, A., 2007. Clinical Decision Support System, *Diploma work.*