

A Flexible Architecture for Constructing and Executing Tutoring Processes

A. Martens, J. Bernauer, T. Illmann, C. Scheuerer, A. Seitz
Department of Computer Science
University Ulm, University of Applied Science Ulm
D-89069 Ulm, Germany

Abstract. The tutoring process of conventional case-oriented medical training systems can be characterised as either guided or unguided. In contrast to that, the aim of the system "Docs'n Drugs" is to distinguish between different levels of guidance. The author can realise the tutoring case as a guided, a half guided or an unguided tutoring process. The system architecture distinguishes between an authoring system and a tutoring system. Fundamentals of these are the tutoring process model and the case data model. This structure allows the reuse of elements of existing tutoring cases. The tutoring cases can be realised in different languages, e.g. German and English.

1. Introduction

At university, traditional medical education is more focused in systematic knowledge about diseases provided by text books and lectures, than on a case-based and problem-oriented application of this knowledge. Yet, the necessary systematic knowledge has to be complemented by a practice oriented training in medical issues with emphasis on patient contact. However, this patient-centred education is cost and time intensive.

Modern technologies have a growing impact on studying and teaching at universities. So, computer-based training (CBT) or web-based training (WBT) systems have become a promising supplement to theoretical courses and lectures, even in the domain of clinical medicine. Constructing a case-oriented computer-based training system for students has been realised by different research projects. Examples for programs are CASUS [1], D3 [2], ProMediWeb [3], PRIMEPRACTICE [4] and "The Interactive Patient" [5].

The project „Docs' n Drugs“, which is presented in this paper, is build on the experiences of existing CBT-programs. „Docs' n Drugs“ provides a case-oriented and web-based training system for students of medicine and other medicine-related disciplines, e.g. medical informatics and medical documentation. The medical education is complemented by interactive learning with virtual patients. As a key-feature of “Docs'n Drugs” the author is able to design different levels of guidance, i.e. guided, half-guided and unguided tutoring processes. Furthermore, a distinction between different user knowledge levels is possible, for example beginner, advanced, expert. It is planned, to complement the this coarse distinction of user knowledge levels by an intelligent system, which should adapt the training case to the users behavior during runtime.

The program will be realised in German and English language. Generally, its structure lends itself for a realisation in other languages as well. In the following section, we will give a short description of the architecture of „Docs' n Drugs“.

2. The Architecture

The architecture of „Docs'n Drugs” is based on two separate models, the tutoring process model and the case data model. The tutoring process model specifies how learning objectives of a training case are mediated and describes the flow of the tutoring process. It

will be described in chapter 3. The case data model, which is described in chapter 4, contains case knowledge and medical knowledge. The case knowledge embeds the knowledge parts that can be extracted from medical cases. The medical knowledge comprises the knowledge parts of all tutoring cases and supplementary medical concepts and dependencies between them. We distinguish two systems, the authoring system, which allows the modelling of training cases by authors, and the training system, where the students work at the tutoring cases. These two systems are merged in the tutoring system.

Figure 1 depicts the components mentioned above.

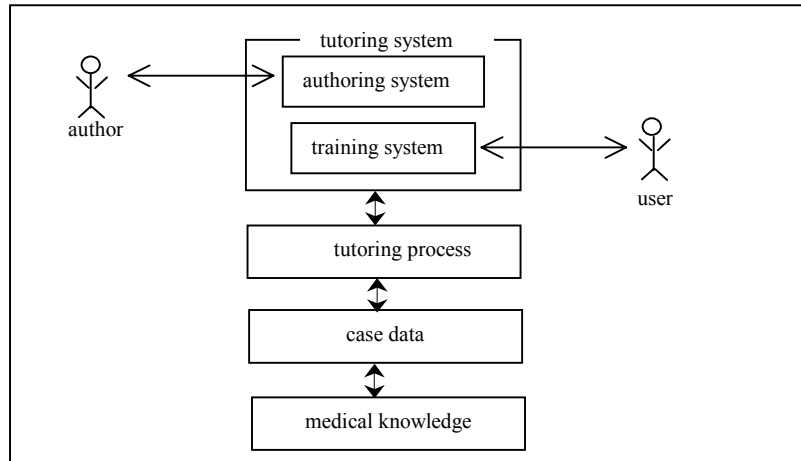


Figure 1: Architecture of „Docs' n Drugs“

3. The Tutoring Process Model

The tutoring process should simulate a patient treatment as realistic as possible, be flexible enough to adapt to different levels of user knowledge and be well structured and understandable.

In hospital, the physician uses no default sequence of asking questions and making first examinations. Usually, she has to decide ad hoc, which is the correct treatment according to the current situation. Therefore, the tutoring process model must allow the construction of different tutoring processes.

To provide a maximum support for students with little experience in patient treatment, the tutoring process should be rather fixed, because it can be assumed that the students has a lack of knowledge about necessary steps. Single steps represent decisions a student has to make during the tutoring process. With an increasing amount of skills, the order of steps that have to be performed within the training case can become more variable. These requirements lead to a structure, which consists of situations, which themselves comprise a set of information and decision pages. Thus, situations and pages are the elements that constitute the tutoring process. The flow between those elements, i.e. the tutoring process itself, is realised by the navigation between the elements.

A situation embodies an interaction phase between the physician and the patient. For example this can be an emergency or an inpatient situation. Situations structure the treatment process within a training case. Each situation consists of:

- a) the according description of the situation,
- b) a tutoring goal of this situation and
- c) a situation navigation.

The description of the situation can inform the user about the environment in which she interacts. A tutoring goal can be for example the differential diagnoses of an important symptom. Each situation comprises several pages. We distinguish between decision and

information pages. The former are used to check the users knowledge and their ability to make the right decisions. The latter provides information necessary for the work at the training case. It can for example represent the results of an examination.

Decision pages provide a correction of the students' answers. A special kind of page is the differential diagnosis. The student should note differential diagnoses according to the actual state of the case. During the work with training case the student has to correct and adapt her differential diagnoses list according to additionally gained information about the patient. Moreover, the student has the opportunity to ask the system for a correction of this list. Besides, the student is supported by a medical record, which contains every examination and finding acquired.

When constructing a training case, the author has to decide whether to construct a guided, a half guided or a unguided navigation between the pages and even between situations. Guided tutoring provides a strict sequence of pages. Thus, it is very useful for students at the beginner level. The author is able to incorporate additional information and comments, because he is able to reason about every step the student can make. Guided tutoring is realised for example in CASUS [1]. In contrast to guided tutoring, the unguided tutoring allows students to make decisions or ask for information arbitrarily. Normally, such a navigation cannot contain predefined information or comments. Thus, it is suitable for the expert level or it should be supported by a human tutor. This kind of guidance is realised in PRIMEPRACTICE [4] and "The Interactive Patient" [5]. The half guided tutoring mixes up guided and unguided sections. It can be realised in different granularities. The author may provide the possibility to choose between different situations. For example the student might decide between an immediate operation on the patient and taking some more examinations. In each situation, the author can enable the choice between different pages and to skip certain pages. The half guided tutoring allows for additional information and comments because the author has knowledge about the students navigation route on a coarse level.

Furthermore, the author of an half guided tutoring can build up his tutoring case for different levels of user knowledge. If a student starts working with a case, she announces her level of knowledge and the tutoring process adapts the training case to this level. A student with a lower level of knowledge gets more help or more guidance, a student with a high experience level gets less help or guidance. Additionally, the structure of the half guided tutoring allows for a dynamical adaptation during runtime. For example, the system can provide further information which an author has not embedded in the current training case. Such information can e.g. be found in the case data or the internet.

Figure 2 shows a screenshot of a web-based tutoring case. The situation description and the situation navigation is at the top of the page. Underneath there is the page itself and the navigation. This example shows a screenshot from an half guided training case. The student can make a choice between different next steps, i.e. history, physical examination, technical examination or laboratory test. The navigation list also shows the differential diagnosis button and the medical record button. It furthermore offers the possibility to choose the "next"-button, which will show the page the author would suggest to be the following one. Because the presented page is a decision page, the student should answer the according questions. She might choose to have it corrected.

The tutoring process is implemented in Java [8] and the training cases are stored in an Interbase database.

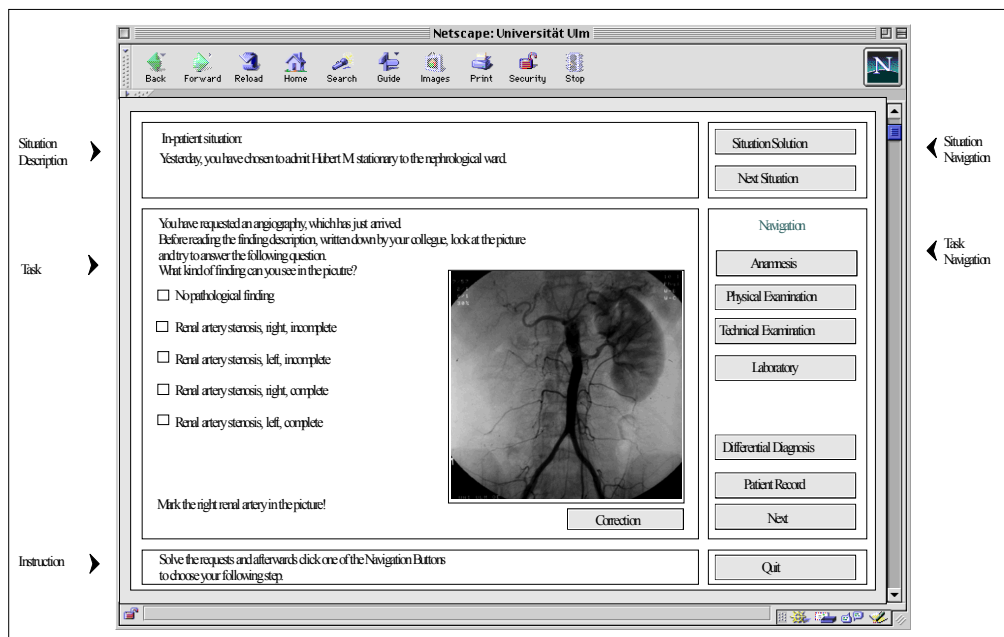


Figure 2: A Screenshot Showing a Decisionpage in a Training Case.

4. The Case Data Model

The case data model should meet the following requirements.

1. The patient data should be categorised according to certain standardised concepts. This allows specific inquiring of facts during the tutoring process. The usage of standardised concepts also allows the reuse of parts of existing tutoring cases [6].
2. The concept representation systems, MeSH [7] and ICD-10 [9], are available in different languages and thus provide multilinguality.

The knowledge a medical case contains is basically represented as concepts and relations between these concepts. We have constructed a database that consists of those entities. This database doesn't represent the whole medical knowledge, but helps to standardise the knowledge embedded in a medical tutoring case. We have separated the following categories of concepts: examinations, anatomy, phenomena, diagnoses and therapies [6]. Furthermore we have added another category named REMS referring to the initial letters of "Risks", "results of Examinations", "pregiven Medicine" and "Symptoms". An element of the "risk" part of REMS is e.g. "patient smokes fifteen cigarettes a day". Relations between concepts might be:

- An examination reveals a special symptom, e.g. an angiography shows a stenosis of the right renal artery.
- A symptom relates an anatomy and a phenomenon, e.g. the symptom "headache" is composed by the anatomy "head" and the phenomenon "ache".
- An examination result supports a diagnosis which requires a certain therapy.

The medical concepts are partly inherited from different concept representation systems. For example, we use a part of MeSH [7] for the anatomy hierarchy and ICD-10 [9] for the diagnoses hierarchy.

The relations mentioned above are explicit, because they are realised by a direct connection between two concepts. We are also able to derive implicit relations. For example a relation between a differential diagnosis and an anatomy and a relation between this anatomy and an examination establishes a kind of implicit relation between this

diagnosis and this examination. The relation has the meaning “this differential diagnosis can be validated by this examination”. The case-based knowledge model allows the realisation of another project aim: the multilinguality. The modular design, which separates the knowledge model from the tutoring process enables a change of the medical knowledge base from German to English language. We provide a one-to-one linkage between the German concepts in the database and the relating English concepts - an author has the possibility to construct a training case in German or in English.

The case-based knowledge model is implemented in Java and bases on a Interbase database.

5. Conclusion

The architecture presented in this paper provides a large amount of flexibility in different domains.

- The structure of the tutoring process model facilitates the distinction between different levels of guidance and the support of different knowledge-levels. It will also allow an individual support of single users.
- The case data model enables the storage of training case elements according to some key-concepts. This enables supporting functions to provide authors and students with further elements in the design-phase or the study-phase of the tutoring process.
- The separation of the knowledge-base and the tutoring process enables multilinguality.
- The web-based application of the tutoring system will allow different kinds of working groups or chat-functions. Students can use the system at home for their private studies or at the university. They will be able to work single or in groups [10].

The case data model is realised and the physicians of our project are working with it. The tutoring process model is tested on several patient cases and proves to be realisable. The first prototype of the tutoring system is part of the curricula in medicine in this summer semester. Currently, the authoring system consists of a set of html forms. Later, this will be replaced by a Java implementation. For further information see our web-page at <http://www.docs-n-drugs.de>.

References

- [1] Fischer M, Schauer S, Gräsel G, et al., Modellversuch CASUS - Entwicklung eines Autorensystems für die problemorientierte Lehre in der Medizin. *Zeitschrift für Ärztliche Fortbildung* 1996; 5.
- [2] Reinhardt B., Generating Case Oriented Intelligent Tutoring Systems. AAAI Fall symposium, IST Authoring Systems 1997.
- [3] Baehring T, Weichelt U, Schmidt H, et.al., ProMediWeb: Problem based case training in medicine via the World Wide Web. *Proceedings of the ED-MEDIA World-Conference*, Freiburg, 1998.
- [4] PRIMEPRACTICE - A CD-ROM Series for Primasy are Practioners, Hematology, Onkology, Volume 2, Number 3, 1996, mayo, Published by: IVI Publishing, Distributed by: Churchill Livingstone
- [5] Lehmann HP, Lehmann CU, Freedman JA: The Use of Simulations in Computer-Aided Learning Over the World-Wide-Web; *JAMA (Journal of the American Medical Association)* 278 (21): 1788, Dec3, 1997
- [6] Seitz A, Martens A, Bernauer J, et al., An Architecture for Intelligent Support of Authoring and Tutoring in Multimedia Learning Environments. *Proceedings of ED-MEDIA*, Seattle, June 1999.
- [7] U.S. National Library of Medicine, Medical Subject Headings (MeSH), 1999
- [8] Martens A., Uhrmacher AM., Modeling a Tutoring Process as a Dynamic Process - A Methodological Approach. *Proceedings of ESM*, Warsaw 1999.
- [9] World Health Organization. International Statistical Classification of Diseases and Related Health Problems, Tenth Revision. Volume 1, 1992.
- [10] Martens, A., Illmann, T., Seitz, A., et.al., CSCW in webbasierten Lehrsystemen, *Proceedings des 5. Workshop der AG CBT der Gmds*, Mai 2000