TECHNISCHE
UNIVERSITÄT
WIEN

VIENNA
UNIVERSITY OF
TECHNOLOGY

DIPLOMARBEIT

Management of Antibiotics for Urinary Tract Infections

ausgeführt am
Institut für Medizinische Experten- und Wissensbasierte Systeme
der Medizinischen Universität Wien

unter der Anleitung von
Univ.-Prof. DI Dr. Klaus-Peter Adlassnig

und

Dr. Alexander Blacky
als verantwortlich mitwirkender Arzt

durch
Jana Rajtarova
Wien

Abstract

Judicious prescriptions of antibiotic substances are becoming crucial as more and more bacterial resistances develop. Unavailable information about the etiology of infections results in applying broad-spectrum antibiotic substances, which foster the development of antibiotic resistances. The objective of this master thesis is to build a prototype of a computerized decision support system for aided prescription of antibiotic substances for urinary tract infections (UTIs) based on extended gathering of information on the incidence of pathogenic organisms and antimicrobial resistances in the respective medical institutions.

For this purpose, data collected within MONI, a knowledge-based system for the early detection and continuous monitoring of nosocomial infections in operation at the Hygiene Department of the Vienna General Hospital, are exploited. From this retrospective patient database, records with data similar to those of the patient under consideration are retrieved. The criteria for selection are same site of infection, and sex, and age, and ward or department. The distribution of pathogens in the group of similar records determines the expected pathogen spectrum. Using fuzzy rules on this spectrum, we define a degree of difficulty of the present case. For patients with complications, it is required to specify a diagnosis which is either determined by the physician, or, if only symptoms are selected, the diagnosis is derived from them by the program's rules. For this application, knowledge about antibiotics, therapies for specific pathogens, and UTIs was collected and a knowledge database was established. Further, a set of UTI specific rules was added to identify complications. With this knowledge and data about the current patient, a list of possible antibiotic therapies is generated. By considering possible side effects, contraindications, allergies, resistance information, and co-morbidities, a patient-specific antibiotic therapy is proposed.

In order to test the program in a first step, we developed five test cases modeling some real life situations. The segregation model safely distinguished simple from complicated cases. The knowledge database and rules provided valid diagnoses and sound antibiotic suggestions. Recommended antibiotic therapies suggested by the program were compared to those published in medical literature. They turned out to coincide with the published recommendations.

The application at hand has shown the feasibility of a computerized decision support system for the management of antibiotics in case of UTIs.

Abstrakt

Sachgemäße und effektive Verschreibung von Antibiotika wird durch die Entwicklung neuer, resistenter Bakterien immer wichtiger. Fehlende Information über die Ursache einer vorliegenden Infektion resultiert oftmals in der Verschreibung von Breitbandantibiotika, welche die Entwicklung von Resistenzen fördern. Das Ziel dieser Arbeit ist es, einen Prototyp eines entscheidungsunterstützenden Systems für die Verschreibung von Antibiotika bei Harnwegsinfektionen (HWI) zu entwickeln, welcher auf gesammelter Information über die Inzidenz von Erregern und deren Resistenz basiert.

Für diesen Zweck werden Daten verwendet, die im Rahmen von MONI, einem wissensbasierten System zur Überwachung von nosokomialen Infektionen, welches an der Klinischen Abteilung für Krankenhaushygiene des Allgemeinen Krankenhauses in Wien eingesetzt wird, gesammelt werden. Von dieser retrospektiven Patientendatenbank werden Datensätze abgerufen, deren Daten ähnlich denen des aktuellen Patienten sind. Die Ähnlichkeitskriterien sind definiert durch Übereinstimmung in Infektionsort, Geschlecht, Alter und Abteilung oder Station. Die Verteilung der Erreger in der Gruppe ähnlicher Datensätze stellt das Erregerspektrum dar. Mit Hilfe von Fuzzy-Regeln wird anhand des Spektrums der Schwierigkeitsgrad des vorliegenden Falles ermittelt. Für Patienten mit Komplikationen ist es notwendig eine Diagnose anzugeben. Diese wird entweder vom Arzt direkt bestimmt, oder, falls Symptome ausgewählt wurden, durch Regeln des Programms abgeleitet. Für diese Applikation wurde Wissen über Antibiotika, Therapien für spezifische Erreger und HWI gesammelt und eine Wissensbasis erstellt. Weiters wurden HWI-spezifische Regeln für das Erkennen von Komplikationen hinzugefügt. Mit all diesem Wissen und den Daten über den aktuellen Patienten wird eine Liste möglicher antibiotischer Therapien generiert. Unter Berücksichtigung möglicher Nebenwirkungen, Gegenanzeigen, Allergien, Resistenz der Erreger und Komorbidität wird eine patientenspezifische antibiotische Therapie präsentiert.

Um die Applikation in einem ersten Schritt zu testen, wurden fünf Testfälle entwickelt, die Situationen aus dem klinischen Alltag widerspiegeln. Das Segregationsmodell unterschied sicher zwischen einfachen und komplizierten Fällen. Die Wissensbasis und die Regeln lieferten gültige Diagnosen und Empfehlungen für Antibiotika. Die von der Applikation vorgeschlagenen antibiotischen Therapien wurden mit den in der medizinischen Fachliteratur publizierten verglichen und in allen Fällen konnte Übereinstimmung nachgewiesen werden.

Mit der vorliegenden Applikation konnte erfolgreich die Realisierbarkeit eines entscheidungsunterstützenden Systems für das Antibiotikamanagement bei Harnwegsinfektionen gezeigt werden.

Contents

1 Introduction

Management of antibiotics is becoming a much discussed issue as multiresistent bacteria emerge. Since the discovery of the first antibiotic, this group of drugs has gone through a great evolution. With the initial success antibiotics were believed to be "magical bullets" able to cure anything [1, pages 11–23]. But already a few years later due to the sometimes rash prescription of this medicine first resistent bacteria were found. This discovery started the research for newer, more effective antibiotics, ignoring the ability of bacteria to evolve and adapt to a hostile environment. This approach started a vicious circle with even more bacterial strains developing resistance and others to become multiresistant. These resistant bacterial strains first developed in healthcare facilities, where immunocompromised patients needed massive antibiotic therapy. Nowadays, due to benevolent antibiotic prescription, they can also be found outside these facilities. The danger of these bacteria was shown in various clinical studies. All over the world it was found that resistant bacteria are the cause of higher mortality of patients, longer hospital stays, higher amounts of medication—all resulting in higher costs. For example, a university hospital in Saudi Arabia reported a mortality of methicillin-resistant staphylococcus aureus infected patients with 60.8%. 37.8% of these deaths were caused by the resistant pathogen [2].

To counteract this development worldwide antibiotic prescription strategies (e.g., "Antibiotika Strategien (ABS)" in Austria) and guidelines were developed, which led to educational campaigns. Researchers and physicians acknowledged the possibility, that research may reach the point, when no other new, effective antibiotic will be found. The strategy for antibiotic prescription changed from prophylactic prescribing to etiologic, which tries to find the causative pathogen and its resistancies. New hygiene guidelines and guidelines for reporting of resistant pathogens and healthcare-acquired infections have been introduced.

To be able to prescribe antibiotic treatment wisely the causative pathogen must be known. However, tests to determine a pathogen's resistance to antibiotic treatment may cause a delay in the prescription process. Finding a pathogen takes up to 24 hours, to obtain a sensitivity pattern of this pathogen up to 72 hours. Therefore the strategy, how to prescribe antibiotics without this knowledge, has become approximative. In every site of infection a more or less specific bacterial spectrum can be found. This spectrum varies due to patient's conditions, age, sex, and immunity. In this way a spectrum is approximately estimated for the current patient instead of waiting for laboratory results. Based on this approximative pathogen-finding process several questions arise. What if this commonly known distribution changes? How is the physician notified about the change? To be up-to-date with these facts, statistics to view the current situation are needed during the prescription process. Another difficulty is that the true pathogen is unknown, therefore therapy must try to eliminate all commonly known potential pathogens. The result is a broad-spectrum antibiotic therapy. Broad-spectrum antibiotic regimen however have higher toxicities, higher cost, and promote the development of antimicrobial resistance [3]. Additionally to the pathogen-estimation problem patient-specific factors must be included in the prescription process. The patient's conditions (e.g., pregnancy, renal or liver insufficiency), other diseases (e.g., diabetes mellitus), contraindicating factors (e.g., allergies) must be assessed and the cost-benefit ratio calculated.

Prescribing is a complex process. It needs a lot of information about the patient, site of infection, clinical signs, and last but not least antibiotic regimen. All this knowledge must be combined to prescribe a treatment. It is a difficult task and becomes even unsolvable considering the number of possible sites of infections and resulting therapies.

The objective of this master thesis was to build a prototype of a computerized decision support system for aided prescription of antibiotic regimen based on collected information of incidence of organisms and antimicrobial resistances. This system is supposed to support the physician in the prescription process by providing all necessary information. The prototype should focus attention, manage information, and provide for patient-specific consultation [4]. The expected result of this prototype is to estimate the spectrum of expected pathogens and to propose a possible treatment. The application visualizes local pathogenic situation and analyzes resistance development of pathogens. Further recommendations of antibiotic substances, specific for the current patient, will be proposed to the physician. The prototype is built for use by physicians in a clinical setting to support the daily process of prescribing. For this prototype we decided to delimit the sites of infection to the urinary tract. Resistant bacteria are often present in urinary tract infections (UTIs). This development is caused by common ambulant treatment without assessing of causative pathogen and prophylactic antibiotic treatment of the UTI, which promotes bacterial resistance. Untreated or wrongly treated UTI may lead to a urosepsis and result in death.

In cooperation with the Hygiene Department of the Vienna General Hospital the idea of analyzing an existing patient database for prescription purposes and monitoring of antibiotics was developed. The database we used for our prototype was originally developed for a system for monitoring of nosocomial infections (MONI) at the healthcare facility [5].

1.1 Problem definition

Based on the described initial situation we defined following requirements for this prototype application:

- exploit retrospective data from the patient database
- show current local pathogen situation
- generate list of recommended antibiotic substances
- consider specific conditions of the current patient (allergies, contraindications, etc.)
- separate knowledge processing from knowledge
- create an appropriate graphical user interface

1.2 Work outline

First the technical (2.1) and medical (2.2) background necessary to understand further work is explained. In section "State of the art" (2.3) two other systems supporting the process of prescription (HELP and ICONS) are described.

The technical background section introduces the reader to knowledge-based systems and decision support systems. Further various forms of knowledge and systems that work with knowledge are discussed. Also general concepts of fuzzy logic and data binding are introduced.

To elucidate the medical background, the etiology of infections, diagnosis of infections, and specifics of antibiotic therapy are discussed. A special section on UTIs with diagnosis and therapy details is followed by the introduction of the prescription process.

In "Material and methods" concepts used for this application are explained step by step.

In "Implementation" the application is described and explained in detail.

Finally in "Results and discussion" results of this work as well as the evaluation of the application follow. The thesis is closed with a discussion, a comparison of this prototype to other existing systems, and suggestions for future developments.

2 Background and state of the art

In this chapter relevant concepts for this master thesis are described. First, the technical background knowledge is explained, then a medical introduction follows. The section state of the art introduces in detail two similar decision support systems for the prescription of antibiotics.

2.1 Technical background

In this chapter we briefly introduce the term "knowledge-based systems", describe their architecture, the application field of medicine, and special types of knowledge-based systems. Further the problem of knowledge acquisition, formalization, and representation as well as various types of knowledge are explained. The formats XML (Extensible Markup Language) and XSD (Extensible Schema Definition) are introduced in the last part of this chapter, as they are necessary to understand certain concepts in this thesis.

2.1.1 Knowledge-based systems

Knowledge-based systems (KBSs) are computer systems with a typical architecture which operate on domain knowledge.

The main characteristic of KBSs is their architecture, which separates knowledge processing and the knowledge itself. The knowledge is collected to build the knowledge base, which must be extendable without having to change the knowledge processing module as well. A knowledge base contains domain-specific knowledge. The algorithms for knowledge processing should be contained in a domain-independent problem solving component.

A special group of KBSs are expert systems. These systems try to imitate an expert of a specific domain. This hierarchy of systems is based upon the origin of knowledge the system contains in its knowledge base. A KBS collects all available knowledge for the domain. An expert system additionally contains knowledge from a domain expert [6, page 11].

The application area of knowledge-based systems is wide. One of the most important fields is medicine. These systems try to solve problems for example of storage and organization of medical knowledge, the process of learning from medical experience, or diagnosing diseases based on a set of clinical signs [6, page 10].

Architecture

Knowledge-based systems are defined by their special architecture. Knowledge contained in these systems is not included in algorithms or other program parts. Knowledge exists in a bundeled form separated from its processing module.

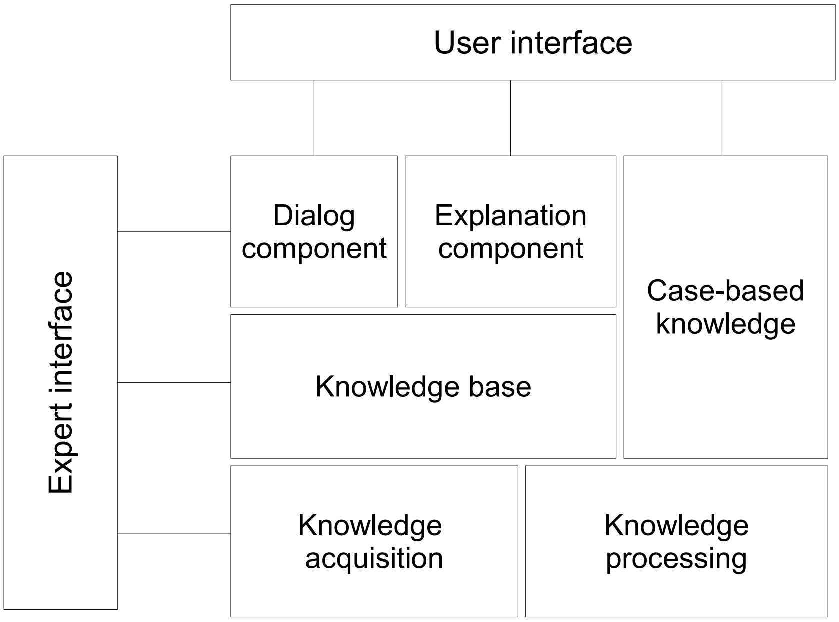


Figure 2.1: Components of a knowledge-based system [6, page 18]

In Figure 2.1 all important components of a knowledge-based system are shown. In the following they are described briefly [6, pages 16–18]:

- A knowledge base consists of all knowledge necessary for computing a solution in a specific domain. Lehmann et al. describe a knowledge base as an "implementation of a formal domain model" [7, page 67]. A knowledge base may contain rules, crisp as well as fuzzy (see [8]), facts, and vocabularies—it is a specified ontology.

- Case-based knowledge contains information about the real world situation. This knowledge may also contain rules, compared to the knowledge base these are changing rules, which are generated or adapted by experts.
- The knowledge processing component processes knowledge from the knowledge base to compute a solution and to provide the explanatory component with the path to this solution. The knowledge contained in this component is called meta knowledge—the knowledge about knowledge. This component is also called the inference component. It should be as domain-independent as possible, however this is not always possible.
- The knowledge acquisition component helps the knowledge engineer to build the knowledge base. Even if the system is already in use, the modification of knowledge contained in the knowledge base should be possible.
- The explanatory component shows how the system calculated the resulting value or decision and what data affected the decision. The end user should always be able to validate and reconstruct every single step of the underlying logic.
- The user interface constitutes the connection to the end user. Through this graphical interface the user enters data into the KBS. Inside the system a dialog component takes care of the communication with the user. It preprocesses and verifies input for the other components and also generates an output in form of advices, suggestions, reminders, etc. Especially in medical environment user interfaces must be user-friendly and easy to understand in order to gain acceptance and help optimizing routine processes. A KBS should have two user interfaces: for the expert to build the knowledge base (see knowledge acquisition) and for the end user, an interface for communication and use of the KBS.

Decision support systems in medicine

Decision support systems (DSSs) are a subgroup of knowledge-based systems. A medical DSS is every system, which was developed to assist with decisions in a medical environment. In the reference book of medical informatics [7], a DSS is defined by the following requirement:

> ... deliver a case or problem-specific response for support of the clinical decision-making process computed by using clinical data and medical knowledge ... [7, page 108]

The field of medical decision making is very specific. Physicians make decisions about clinical situations of patients based on medical knowledge every day. The basis for these decisions—details about the patient's situation—often is neither complete nor exact. Nevertheless a decision about further steps and therapy must be made as soon as possible. To make this decision medical knowledge is necessary. This knowledge however sometimes changes and develops. In antibiotic therapy bacterial resistances change over time. To be able to provide effective therapy new substances and new therapy methods are invented. It requires time and effort of physicians to keep up with all new facts and therapies because of this fast change. Therefore systems to support physicians in their process of decision making by providing and visualizing important and up-to-date information are developed.

Application of decision support systems in medicine

Various types of medical decision support systems exist [7, pages 137–138]:

- Knowledge server allow the centralized access to medical information and knowledge sources in a clinical information system. The vast amount of information and data is connected by special rules so that intelligent medical information retrieval is possible.

- Diagnostic and therapeutic expert systems are the classic field of knowledge-based and decision support systems. By combining diagnostic findings, clinical signs, risk factors, and previous illnesses a therapeutical or diagnostic decision is computed and presented to the physician in charge. The subject of this work, an application of a decision support system, falls into this category.

- Monitoring systems are systems which, based on specific knowledge, produce warnings and messages about certain events. The integration into the clinical routine is best in this category of decision support systems.

- Prognostic systems support physicians by suggesting possible future development of a patient's condition, disease, or therapy.

- Knowledge-based reference guides organize or create special information contexts.

- Tutoring systems help to obtain, improve, or refresh medical knowledge.

2.1.2 Knowledge

In order to be able to use medical knowledge in an automated process, it has to be brought into a general form (rules, database, etc.). For this purpose its various types and forms are analyzed and the different possibilities of knowledge representation are discussed. Last but not least the process of knowledge acquisition for expert systems is explained.

Knowledge in medicine can be categorized as follows [7, pages 106–107]:

- Experiential knowledge forms the basis of medical knowledge. Descriptions and definitions of diseases are composed of observations of symptoms and disease progress. Valid medical studies with correct statistical evaluations may be described as experiential knowledge. Evidence-based medicine (EBM) is based on this kind of knowledge [7, page 657].
- Vague knowledge is very common in medicine and a challenge for expert systems. Biological processes in medicine often are not measurable. In the attempt to describe these complex processes special linguistic tools have been developed to classify them. The terms do not exactly correspond with measurable values, e.g., high fever. Fuzzy modelling is a tool to handle such knowledge.
- Knowledge available in medical processes is not complete. On one hand the medical knowledge itself is changing on the other hand knowledge about the patient's situation is not completely available. Resulting from the unavailability is vagueness and the need for decisions about missing data. Records with missing data may be defined as corrupt and not used for calculation. Handling missing knowledge constitutes a part of the design of a KBS.
- Model knowledge is necessary for abstraction and analysis of biophysical and biochemical processes in the human body. To be able to predict these processes abstract and complex models have been developed.

Types of knowledge

Another abstraction of previously described knowledge categories is the classification based on the way it is formalized [9, pages 3.4–3.5]:

- Facts are descriptions of objects and their characteristics.

- Relations between facts are another important information. E.g., hand is an upper extremity. Relations can be static, temporal, or causal.
- Methodic knowledge describes process steps—how and in which order certain relations or facts occur.
- Meta knowledge is the knowledge about the knowledge base. This knowledge describes the structure of the knowledge base, the application and structure of knowledge it contains. Further it contains algorithms how a decision or solution of a given problem is achieved by using a knowledge base.

Vague knowledge in medicine

Vagueness of knowledge in medicine may result from several factors [10, pages 54–55]:

- imprecision of the measuring process
- vagueness resulting from representational or linguistic vagueness of facts
- uncertainty about occurrence of certain facts
- incompleteness of facts

These sources of vagueness must be identified in the design process of a KBS. Causes of vagueness must be eliminated by enhancing respective processes (e.g., more precise measurement) or dealt with in the system (e.g., calculate with imprecisions). There are several techniques to reduce the impact of vagueness:

- For imprecise measuring techniques measurement control techniques may be developed.
- Representational and linguistic variables in medicine can be mapped to corresponding sets of values by fuzzy logic.
- If the occurrence of facts is uncertain, a process together with the domain expert must be designed, how to handle them. Either these facts are not used for processing because of their uncertainty, or in combination with other, more certain facts, a new context is given.

- Incomplete information such as clinical signs may be calculated from existing data, if possible, or a program may demand this information from the user. Missing facts in patient data may also indicate that an information was not needed because another meanful fact was found (e.g., no urine sample because pathogen was already found in a blood sample).

Rule-based systems

A rule-based system is a kind of knowledge-based system. The term "rule-based" describes the part of the knowledge base, which contains the methodic and relational knowledge described in rules. This knowledge consists of domain-specific and common knowledge. Domain-specific knowledge contains theory as well as experience organized in logical rules. Common knowledge may contain problem solving heuristics, optimization rules, or knowledge about objects and relationships of the real world. Rules are either changed by the domain expert because of innovation in the field, or they are permanent (e.g., pregnant patient must be female). Changes, however, do not occur very often, so we call this unchanging knowledge. Rules mostly have the following syntax:

```
IF A AND/OR B THEN C
```

The main challenge in rule-based systems is to select correct rules in proper order. Two approaches in evaluating selected rules can be distinguished [9, pages 4.2–4.10]. For a given fact evaluate the rule and find the answer:

- with forward chaining we try to find a diagnosis—data driven

  ```
  IF fact THEN answer
  ```

- with backward chaining we try to verify a hypothesis—goal driven

  ```
  IF answer THEN fact
  ```

The disadvantage of a rule-based system is its static data. Rule-based systems have their knowledge coded once and more effort is needed to renew the information contained. Further they can "think" only in their given set of rules. The advantage of this approach is the possibility to verify that the rules are consistent and valid. Mostly they are created by domain experts.

Case-based reasoning systems

Case-based reasoning systems (CBRs) take their knowledge from generalizing past behaviors. For example, diagnosis and therapy outcomes of a group of patients are generalized and the so gained knowledge is applied to future cases. In systems with CBR the knowledge base does not consist of rules but of a set of cases, which contain previous experience [6, page 155]. CBR systems are based on the following assumptions [6, page 157]:

- similar problems have similar solutions—solutions of solved problems are a good basis to solve similar problems
- every problem is different, but the type of question repeats itself—after having solved one type of problem for the first time, next time being confronted with a similar problem it will be easier to solve it

Essential questions on CBR systems to be solved are [6, pages 157–200]:

- Where and how are new cases obtained?
- How is the similarity of cases determined?
- How are generalized cases saved and represented?
- How are appropriate cases retrieved for the current situation?
- How can these cases be retrieved in real time?

Every single question bears its difficulties and disadvantages with respect to this type of system. For example, the correct and reasonable functioning of such a system is determined in a large extent by the question, whether the underlying cases were representative enough to be generalized and applied to new patients, and whether the generalization reflects reality correctly. Real-world systems however must refer to real data. These data underly temporal changes and cannot be represented by rules. These changes of data and the unstructured real-world knowledge is best expressed by the knowledge retrieval method CBR [11].

2.1.3 Fuzzy logic

The theory of fuzzy logic was developed by Lotfi A. Zadeh [8] at the University of Berkeley, California, in 1965. Fuzzy logic allows to integrate knowledge, which contains vague values, into algorithmic systems. Compared to the Boolean (dual) logic, which calculates with "true" and "false", fuzzy logic operates with infinite logic values.

The logic values in Boolean logic exclude each other. To explain the basic concept of fuzzy logic we first define logic true/false values. Let m be the membership function of an element in a set. The crisp logic defines the membership of an element x to a set A as follows [6, page 420]:

$$\begin{aligned} A &= \{x \in N \mid a < x < b\} \\ m(x) &= 1 \quad if\, a < x < b \\ m(x) &= 0 \quad else \end{aligned} \tag{2.1}$$

The membership function in Boolean logic classifies x as a member of the set A if a certain condition is fulfilled, e.g., x is greater than a and less than b. If x is equal to b it is no longer a member of set A. The result of a Boolean logic membership function is defined by two values: either an element is contained in the set (true, 1) or not (false, 0). In many fields, especially in medicine, the use of true/false logic does not suffice to describe states and vague knowledge (see section 2.1.2) accurately. Fuzzy theory allows gradual membership of an element to a set. The less the condition for a membership to a function is fulfilled, the less the element is a member of the respective set. A fuzzy membership function therefore returns a real number between 0 and 1, indicating the degree of membership of an element to a set. 1 indicates full membership to a set and 0 no membership at all.

$$x \in A \;\; if \;\; 0 < m(x) \leq 1 \tag{2.2}$$

From the point of view of mathematical logic and the previous given example it does not make sense how an element of a set may be a member by 0.50. In fuzzy logic this statement would define an element, which belongs to a set partially with a membership degree of 50%. Membership values are mostly mapped to linguistic variables. These are easier to interpret and understand and thus used for further processing. In the following example an application of fuzzy logic and mapping of fuzzy membership values to words are shown.

Example: fever

In medicine when measuring and reporting fever, a patient may be defined to have "a touch of" fever. In this case we cannot say that the patient truly has fever but he also is not fully free of fever. Therefore a fuzzy function starting at 36.6°C and ending at 38.5°C is created, see Figure 2.2. The first value is known as normal temperature and the respective membership function for this temperature gives 0, meaning the patient has no fever. The second value is defined as fever. Therefore, when measuring a temperature of 38.5°C and higher the patient is defined to have fever. Now to be able to define the term "a touch of" fever we must map the result of our membership function to this linguistic term. Let us assume that "a touch of" fever means for a physician a temperature of about 37°C. As can be seen in Figure 2.2, 37.5°C has a membership value of 0.5. So, "a touch of" fever can be defined by a membership value between 0.1 and 0.4. When the membership function returns a value within this interval then the patient is said to have "a touch of" fever. Outside this interval a different fever value is defined.

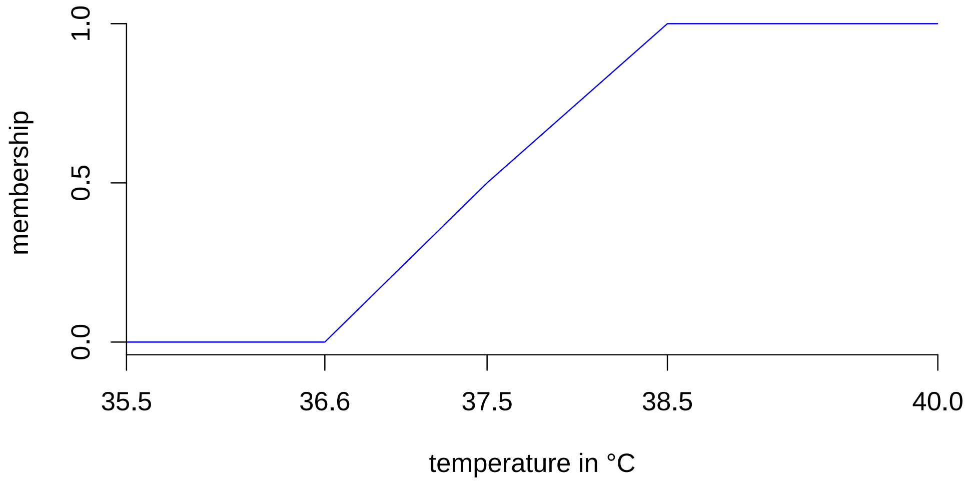


Figure 2.2: A fuzzy function to determine the degree of having fever

Fuzzy operators

In the following the basic logic operators (AND, OR, NOT) are introduced and compared to corresponding fuzzy operators. In equations 2.3 through 2.7 we use following notation: $m_{\tilde{A}}(x)$, which shows a fuzzy membership function m of the element x to a set $\tilde{A}$. This membership function defines to what extent the element belongs to the defined set.

First, the fuzzy NOT operator is described. In equation 2.3 it can be seen that this fuzzy operator has the same functionality as the logic NOT. By negating a membership function we receive opposite values: 0 instead of 1 and vice versa. However, a result of 0.5 remains the same even if negating the membership function.

$$m_{NOT\ \tilde{A}}(x) \ = \ 1-m_{\tilde{A}}(x) \tag{2.3}$$

For the next two operators the following two cases must be distinguished:

- result of one situation is assessed by two different membership functions resulting in two values
- two situations are assessed by two membership functions resulting in two values

Based on these cases we distinguish which operator is used. For the first case the min and max operators are used to compute intersection and union. An example for the first case would be to assess a membership value for the attribute fever by two membership functions. To AND relate the two resulting values, the minimal value of the two results is to be taken, see equation 2.4. This way it is ensured that each membership function returns at least the minimal value for the given attribute.

$$m_{\tilde{A}\bigcap\tilde{B}}(x) \ = \ min \ \{m_{\tilde{A}}(x),\, m_{\tilde{B}}(x)\} \tag{2.4}$$

For the OR relation of the values from the first case, the maximum of the two membership function values is returned. The reason for taking the maximum of the two values is to state that the expected value has to be below the maximum—both values can be found below the maximum.

$$m_{\tilde{A}\bigcup\tilde{B}}(x) \ = \ max \ \{m_{\tilde{A}}(x),\, m_{\tilde{B}}(x)\} \tag{2.5}$$

For the second case, two values resulting of two situations assessed by two membership functions, an other set of operators is used. For example, if we wanted to determine the degree of a specific illness of a patient, we could define for each clinical sign (fever, headache, running nose, etc.) a fuzzy function and combine the results from the respective membership functions. The fuzzy AND operation for this case, which can also be represented as the algebraic product of two fuzzy membership functions, is assessed as given in equation 2.6.

$$m_{\tilde{A}\bigcap\tilde{B}}(x,\, y) \;=\; m_{\tilde{A}}(x) \;\times\; m_{\tilde{B}}(y) \tag{2.6}$$

If we used the previous equation to assess a cold we could define the measured temperature as x and a fuzzy membership function $m_{\tilde{A}}$, the severity of headache as y and the respective fuzzy membership function $m_{\tilde{B}}$. The fuzzy membership function of the union of two sets returns the degree of membership to both of these sets at once. This way we may classify the severity of cold of a patient. In Boolean logic we would only receive two values: patient is affected or is not. We would not be able to determine how severe the disease truly is. The fuzzy OR operation for the second case is represented by the algebraic sum.

$$m_{\tilde{A}\bigcup\tilde{B}}(x,\, y) \;=\; m_{\tilde{A}}(x) \;+\; m_{\tilde{B}}(y) \;-\; m_{\tilde{A}\bigcap\tilde{B}}(x,\, y) \tag{2.7}$$

This algebraic sum defines to which extent the element x is present in set $\tilde{A}$ and y in set $\tilde{B}$. For example, if a diagnosis would be indicated by two clinical signs, which may occur separately or together, we would apply this fuzzy OR operator.

2.1.4 Data binding

The Extensible Markup Language (XML) is a language to save hierarchically structured data. This format is often used to exchange data between applications. The challenge is to intelligently read and write these data into the respective system. Various techniques have been developed for parsing of XML documents. These techniques are based on simple, textual parsing of the file, but can be distinguished by included knowledge about XML and usage. Here are some examples:

- DOM technique: creates an object tree of XML nodes [12, pages 91–121]
- SAX technique: is a parser automat which can be stopped at a specified XML node [12, pages 41–69]
- data binding: connects XML content and application content [12, pages 310–347]

Especially in Java these concepts are available and may be used without much programming efforts. For this work we decided to use the data binding technique.

> ... data binding is an XML processing technique that eliminates references to XML nodes from your code. [13, page 310]

Data binding allows applications to move XML content to object structure and vice versa. Data binding binds XML content to an existing object structure. The content of the XML file is read into the corresponding structure at one point of the program and can be accessed within the program. This requires the program's object structure to be the same as the XML structure and vice versa. Within data binding the process of parsing data is called unmarshalling. Through unmarshalling document content is filled into a given object structure. The opposite process, reading data from objects and creating an XML file, is called marshalling.

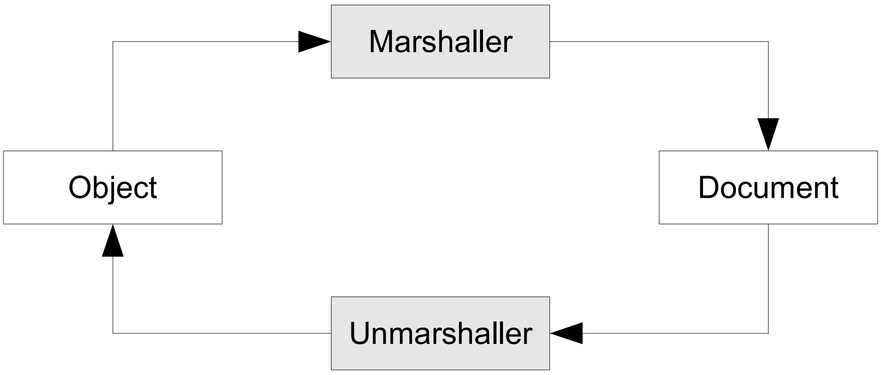


Figure 2.3: Process of data binding

To produce an XML file from a given object structure or an object structure from an XML file a schema file is necessary. Schema files define the structure of an XML file or an object structure. Schema files may be Document Type Definition (DTD) files, XML schema files, or some other schema language. We used the XML Schema Definition (XSD) language to define our XML files. The XML document must correspond with the XML schema from which the classes are compiled. These classes are then instantiated by an unmarshaller which results in objects filled with XML data. These objects are then, after using them in the program, marshalled into an XML-schema-conform document. Benefit from data binding is the direct access to XML content on program level. Data binding is to be used if the following prerequisites are fulfilled [13, page 314]:

- an XML structure can be defined in order to produce a corresponding set of objects
- the XML file is not intended to be used for mixed content
- XML files are not large documents

The first point must be given to be able to use data binding. Mixed content, such as hyperlinks, is problematic to be expressed in various frameworks. Therefore this type of content should be omitted in data binding. The last point refers to the functioning of data binding. For each XML node at least one instance of a class is created. The amount of memory determines how long it will take that large documents can be marshalled and unmarshalled. For very large documents other technologies are to be preferred.

2.2 Medical background

In this section medical terms and processes important for this thesis are introduced. First, the basic terms for this field of medicine are defined. Further the therapeutic process is described, particularly the etiology of infectious diseases, their diagnosis, and selection of an appropriate therapy. The focus is narrowed to one specific site of infection—the urinary tract. At the end of this section the prescription process is described in more detail.

2.2.1 Definition of terms

- Bactericidal effect:
  bactericidal substances kill bacteria by damaging them severely
- Bacteriostasis:
  reversible inhibition of growth and reproduction of bacteria by bacteriostatic agents
- Minimum inhibitory concentration (MIC):
  minimal amount of an antibiotic substance, which prevents pathogen growth and reproduction
- Minimum bactericidal concentration (MBC):
  smallest concentration of a substance which eliminates 99.9% of a pathogen population in a given time
- Antibiotic spectrum of activity:
  the variety of microorganisms affected by a substance
- Bioavailability:
  the fraction of the administered active substance that reaches the site of infection after having been ingested

- Spectrum of resistance:
  statistic evaluation of all sensitivity patterns in one area. It is a set of substances tested for sensitivity on a specific pathogen with respective resistance values.

2.2.2 Etiology

Etiology is the science of causes of diseases or pathologies. Pathogens and their characteristics, host risk and other disease-causing factors, as well as the definition of infection as a disease are explained.

Pathogens

> A pathogen is a microorganism, with the ability to cause an infection. [14, page 43]

Microbial pathogens are viruses, bacteria, fungi, and protozoa. Pathogens are distinguished by their anatomy, their genome, the course of reproduction, and their metabolic properties [14, page 46]. General characteristics, determining the behavior of a pathogen in a host organism, are [14, pages 43–44]:

- Contagiousness shows the ability of the infected host to spread the pathogen in its surrounding.
- Infectiousness determines the ability of a pathogen to cause an infection.
- Pathogenicity is the ability of a pathogen to cause an illness in its host. Two types of pathogenicity for microorganisms can be distinguished:
  - Facultative pathogenicity describes microorganisms, which cause an illness only under certain circumstances. These microorganisms belong to the natural, physiologic flora of the macroorganism. An opportunity for such a microorganism to become pathogenic may be reduced immunity of the macroorganism, predisposing factors or displacement from its natural site to a different tissue in the macroorganism. To cure an infection caused by facultative pathogens it is necessary to eliminate them from the infected site but preferably to leave them unharmed on their natural site.
  - Obligatory pathogenic microorganisms cause an infection to the host organism every time after colonization. These have to be eliminated from the host entirely [14, pages 43–45].

- Virulence determines the degree of ability of a pathogenic microorganism to cause an illness, thus to colonize, invade, reproduce, and cause damage to a host organism.

Bacteria

Bacteria are prokaryotic organisms varying in their size from 0.2–3.0 μm. Mostly bacteria of the size of 1–3 μm are relevant for medicine [14, page 113]. Important for diagnosis and therapy of bacterial infections is the cell membrane of bacteria. By identifying the structure of the bacterial cell wall the most effective and specific therapy can be chosen.

Bacteria are grouped into families. Each family contains several species. However each specie may show different strains. Differences between these strains are mostly slight and can be found in the genome. These genome modifications may cause for example different reactions to antiinfective treatment resulting in resistance.

Gram stain

To select correct and effective antibiotic therapy, characteristics of expected bacteria have to be found. One of them is also the gram stain of the pathogen. The gram stain is a technique to determine physical and chemical properties of the bacterial membrane. Microbiologically is the gram stain one of the most important coloring techniques of bacterial membrane. It shows the quality of the murein layer in the bacterial outer membrane. The process of gram staining requires coloring bacteria and then applying alcohol to remove the color. Bacteria with more layers of murein in their membrane stay colored blue, others become red. Blue color defines gram-positive bacteria, red gram-negative ones [14, page 288]. In Table 2.1 examples of various gram-negative and gram-positive bacteria are given.

Infection

An infection is a settlement, growth, and reproduction of microorganisms in a macroorganism, which shows defense reactions and/or signs of damage.

Infections are a prerequisite of an infectious disease but are not tantamount to an illness, since infections can be both symptomatic or asymptomatic, progressing without any damage. An infection is an interaction between host and pathogen. The development and course of the disease depend on the receptiveness (innate immunity—external barriers) and on the defense (immunity) of the macroorganism as well as on the infectious and pathogenic properties of the microorganism [14, page 3].

Table 2.1: Overview of important gram-positive and gram-negative bacteria [14, page 114]

form	**gram-positive**	**gram-negative**
cocci	Enterococcus spp Streptococcus spp Staphylococcus spp Peptostreptococcus spp	Neisseria spp Veillonella spp
rods	Corynebacteria spp Listeria spp Erysipelothrix spp Clostridia spp Bacillus spp Lactobacillus spp	Enterobacteriaceae spp (*E. coli, Enterobacter*) Proteus spp Klebsiella spp Pseudomonas spp Campylobacter spp Helicobacter spp Haemophilus spp Legionella spp Bacteroides spp Aeromonas spp
spiral		Treponema spp, Borrelia spp Leptospiral spp

Transmission of infections

Two kinds of transmission can be distinguished: direct and indirect transmission. Direct transmission requires a direct contact to the infection source.

Indirect transmission can be classified as follows [14, pages 5–7]:

- Vehicle-borne infections spread through unanimate carriers (e.g., water, food).
- Vector-borne infections spread through animate carriers.
- Air-borne infections spread through dusts or aerosols over longer distances.

After transmission the pathogen colonizes the host. This means that it settles on the inner or outer tissues of the host and cannot be removed mechanically by the host anymore. After the colonization process the invasion into the tissue of the host begins. The pathogen now reproduces in the tissue causing damage to the host by intracellular reproduction (causing lysis of cells), directly by production of toxins (metabolizing tissues), or indirectly by causing an inflammation.

2.2.3 Diagnosis

When an infection is suspected the patient mostly shows several subjective symptoms. These may specify the illness, but must be ascertained by a physician. After obtaining material from the patient it is sent to a laboratory and microbiological tests are performed. Data obtained from these tests form the so-called objective symptoms and are made up of various laboratory parameters [14, page 277]. Most of the parameters help to recognize whether the patient truly is infected. In order to find the causative pathogen a specimen of the tissue or a secretion is taken from the patient and sent to the laboratory for further testing.

Tissues and materials

Based on the suspected diagnosis and the site of infection a material probe is taken. The material may be either a secretion, a swab, a punctuation, or a tissue. The more of this material is available, the higher the probability to find the causative pathogen. We distinguish sterile and possibly contaminated materials. The probe is contaminated mostly by passage through another body region colonized with another natural flora. Materials coming from such regions have to be marked to avoid false interpretation of results. For example bladder puncture urine is a material obtained without contamination. Urine obtained in a regular way may be contaminated by other bacteria than the causative ones.

Sensitivity screening methods

First of all the specie of the microorganism is determined from the patient probe in a laboratory. Depending on the material and site of infection a quick diagnosis through the microscope may help to distinguish between viral, bacterial, and fungal infection.

Through various coloring techniques other characteristics of the organism may be determined, e.g., form, size, gram stain, cell membrane type, certain structures of bacteria (flagella, spores, etc.). After this procedure the bacterium's specie can be guessed.

Another technique is to grow culture, where the material is placed on a Petri dish with a culture medium. After 24 hours in an appropriate setting the bacteria have grown, so that they can be observed. The setting and culture medium is determined by previous suspicion of the origin of infection. The form of the bacterial growth is a further clue to estimate the specie of the found bacteria.

Another important piece of information is the resistance behavior of the microorganism coded in sensitivity patterns. Out of many techniques we would like to describe two: dilution test and agar diffusion test.

The dilution test determines the minimal bactericidal concentration of an antibiotic substance. The antibiotic substance is diluted in different concentration and the survival of microorganisms in this dilution is observed.

Another technique is the agar diffusion test or inhibiting areola test. After having grown a culture of the suspected bacteria and having estimated the most probable kind, a set of testing antibiotics on small platelets is placed on the Petri dish. For each bacterial kind a different set of antibiotics is tested. These sets have been determined over time by experience. For certain bacteria it is also known whether special resistancies could occur and therefore these antibiotics are also tested. After another 24 hours the Petri dish is analyzed by the laboratory staff.

Two cases may occur: surrounding the antibiotic platelet a bacteria-free ring has formed or no visible change in the culture happened. In the first case the "ring" area, also called the inhibiting zone, shows that the antibiotic is effective and the bacteria is sensitive to this antibiotic—a cure with this agent would be successful. The second case shows a total resistance to an antibiotic agent. To prescribe such a medicine would have no effect on the patient's condition. The precondition for the use of this method is the linear relation between the diameter of the inhibiting zone and the MBC.

However, in reality it is not always as easy as described. Rarely a little has happened on the Petri dish. Mostly some of surviving bacteria still remain in the inhibiting zone. This state is defined as intermediate sensitivity or resistance. The degree of resistance is coded in percent—how many percent of pathogens were eradicated.

The amount of growth of the pathogen is evaluated resulting in a percentage of growth value. Values of resistance, sensitivity, or intermediate sensitivity are determined by special rules based on the percentage of growth. These values are then written into the appropriate column of the antibiotic agent [14, page 323].

2.2.4 Antibiotic therapy

Therapy intends to help the macroorganism to deal with the invading microorganisms. The goal is to eliminate pathogenous microorganisms and not the macroorganism. There are various antimicrobial medications each targeted on a different group of microorganisms. Antibiotics

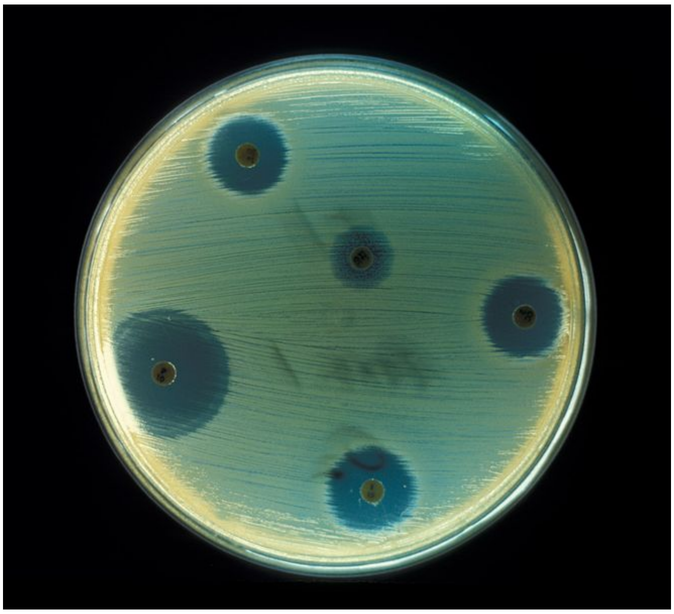


Figure 2.4: Inhibiting areola test of *Staphylococcus aureus*—antibiotics test plate (from [15])

treat only infections caused by bacteria. In this chapter antibiotics are introduced and their role in treatment of infections, the prescription process, its steps, and related occurring difficulties are discussed.

Pharmacokinetics

Pharmacokinetics is the science of the temporal change of the concentration of a substance in an organism. It also describes how an organism affects a given substance and includes following effects, which help to determine the choice of an antibiotic prescription [14, pages 313–314]:

- Resorption describes the passage of a substance through the blood or lymph circulation. In pharmacology resorption determines the route of administration of a substance. For example, if a substance is well absorbed through the intestines an oral administration is possible.
- The distribution of the antibiotic in an organism determines for which sites of infection it may be used. Not all sites of infection can be reached by the blood stream and the substance has to arrive at the infection site by other means. Whether an active substance is able to reach a special compartment in the organism is given by the distribution characteristics of this substance.

- Metabolization of a drug explains how the active substance is modified or degraded by the organism's metabolism. For example, orally administered drugs are subjected to the so called "first pass effect" in the liver, where the bioavailability is reduced.
- Elimination are all processes in the organism to decrease the concentration of a drug. Renal and liver insufficiency may prolong the duration of this process, therefore the dose has to be adjusted especially in these cases.

Mode of action

The goal of antibiotic substances is to eliminate or inhibit growth and reproduction of bacterial cells. The differences between human and bacterial cells are exploited in order to affect the right ones. There are various points which can be attacked by a substance to affect a bacterium's growth, reproduction, or life [14, page 313]:

- Cell wall synthesis
- Cell membrane
- Protein biosynthesis
- Nucleic acid synthesis
- Intermediary metabolism

Resistance

Antibiotics are powerful drugs if used wisely. Excessive use of antibiotics for non-bacterial illnesses such as the viral common cold, not taking the full prescribed dose of drugs because feeling healthy, or even excessive use of prophylactic traveler's antibiotics are examples of misuse of antibiotic drugs, which may result in developing dangerous resistances of bacteria to various kinds of antibiotics.

In antibiotic therapy resistance means further growth of bacteria after the therapeutic concentration has been reached. This results in having to choose stronger medication and thus often in higher costs too.

Three kinds of resistance can be distinguished [14, page 315]:

- Natural resistance:
  This type of resistance is the natural characteristic of a bacterium to protect itself against an unfriendly environment. This resistance is generally known and shows the reason why special bacteria can be treated only by a certain group of antibiotic substances.

- Resistance through mutation:
  Resistance in microorganisms is a natural evolutionary process. Resistance through mutation occurs 10^{-6}–10^{-9} times in a bacterial population. These mutations may lead to higher resistance towards antibiotic substances. An ongoing antibiotic treatment helps the natural process of evolution to select these resistant bacteria. This process can be compared to a farmers field, where the farmer is trying to eliminate all insects with pesticides. The evolutionary theory of selection states that all or nearly all members of a species have to be eradicated in order to prevent the development of resistance. If a small group of this species survives a special event, like the treatment with pesticides or antibiotics, it means that they have a special strategy or feature which helped them. Therefore the next generation will be less susceptible to damage if this event reoccurs [1, page 19]. Resistance traits are inheritable among bacteria since they ensure their survival.

- Transmitted resistance:
  The resistance information can also be exchanged among bacteria so that a microorganism which never was confronted with an antibiotic can acquire such a gene from another microorganism of a different kind. This type of resistance is called transmitted resistance. There are several ways how a bacterium may acquire resistance information: intake of desoxyribonucleic acid (DNA) from the medium, through injection of DNA material by a bacteriophage, or sexually transmitted from the same or different specie.

Bacteria resistant to one antibiotic agent are mostly resistant to the whole class this agent belongs to. So, for example, *Staphylococcus aureus* may become resistant to the whole group of penicillins [1, page 21].

Resistance types

Types of resistance in certain kinds of bacteria are known. It is also known which antibiotic regimen are ineffective with these resistant bacterial strains. The most common resistance types are the following:

- ESBL (extended spectrum beta lactamase): the enzyme beta-lactamase produced by these bacteria destroys the effective molecule of beta lactam antibiotic regimen
- ORSA (oxacillin resistant *Staphylococcus aureus*): formerly known as MRSA (methicillin resistant strains of *Staphylococcus aureus*)
- VRE (Vancomycin resistant enterococci): mostly found in healthcare settings

Table 2.2 summarizes bacteria that are able to develop the discussed resistance types.

Table 2.2: Resistance types and bacteria

resistance type	**pathogen name**
ESBL	*Escherichia coli* *Klebsiella pneumoniae* *Enterobacter cloacae* *Proteus mirabilis* Proteus species *Proteus vulgaris*
MRSA	*Staphylococcus aureus*
VRE	*Enterococcus faecalis* *Enterococcus faecium*

Side effects

Although antibiotics are relatively harmless to the host their side effects vary from the most common intestinal upset and diarrhea, allergic reactions, potential danger to mother and fetus in pregnancy, to rare kidney and brain damage. These side effects can be divided into three groups [14, page 319]:

- toxic effects related to the dosage of medication,
- allergic side effects,
- and biologic side effects, when due to elimination of physiologic colon flora other bacteria colonize the colon.

Especially wide range antibiotics negatively affect the natural, intestinal bacteria, necessary for digestion and the immune system, which results in previously mentioned side effects. These side effects are a price for the effective elimination of pathogenic bacteria.

Contraindications

In many cases antibiotics cannot be used because of patient's characteristics and/or conditions. There are several reasons why substances are contraindicated. The first common reason is a known allergic reaction to a substance. Allergies against penicillins and cephalosporins are very common.

Other characteristics may be patient's age or condition (pregnancy). These patients either must not receive a certain drug (fetotoxicity, etc.) or the dosage of a substance must be adjusted (according to weight for small children). Very often patients are not treated for only one disease but suffer from multimorbidity. If other diseases are treated as well the medication has to be coordinated with other prescribed substances too. Organ dysfunction is also a big issue. For elimination of antibiotics from the organism, liver and renal elimination must be functioning. Organ dysfunction either results in dosage adjustment or contraindication of a substance.

The immune system of the patient is also decisive. If an elderly catheterized patient with one or more chronic diseases acquires a bacterial infection, the therapy will be different than for a young sportive person. In the first case the infection may result even in death if not treated properly, in the latter case the immune system of the patient may manage the infection with little therapeutical support.

Antibiotics

The word "antibiotic" originates from the Greek words, "anti" and "bios", and means "against life". Nowadays this term is used to describe the treatment for bacterial infections. Antibiotic substances are used to eliminate or inhibit growth and reproduction of pathogenic bacteria and help the host organism to deal with this threat.

Antibiotics can be classified in many ways. The most common differentiation is between bacteriostatic and bactericidal substances. The explanation of this two effects is given in section 2.2.1. It is important to know which substance affects a bacterium in which way, because substances with different effects must not be administered in combination [16, page 198].

Antibiotic substances are also grouped by their characteristics. Members of such a group have similar chemical aspects. These large groups have subgroups, which contain antibiotics with larger similarity.

During the prescription process it is necessary to know the membership of each antibiotic substance. This knowledge helps to exclude substances if an allergy to one group exists. If a resistance to one member of a group occurs, others of this same group might be ineffective as well, etc.

One of the characteristics of an antibiotic is the range of effect. Narrow-range substances affect specific bacterial strains, broad-range substances have a wider scope.

Combinations of antibiotics are often used to broaden the antibiotic spectrum of activity, to reduce the probability of selecting a bacterium with mutational resistance and sometimes also because of their synergistic and additive effects [14, page 321].

Antibiotic therapy

We distinguish three kinds of antibiotic therapies:

- Prophylactic therapy is used in perioperative care to prevent infections after surgery. Antibiotics are administered before or shortly after the surgery without the patient showing any symptoms of infection.
- In the case of therapeutic antibiotic prescription, the antibiotic agent is selected when all relevant information, especially causative microorganism and its resistance behavior, is available. The process of drug selection follows taking a patient probe, isolating the causative pathogen, and generating the pathogens antibiogram. Because of this knowledge specific narrow-spectrum antibiotics may be used.

  The main disadvantage of this approach is the late onset of therapy. To receive all microbiology data including the antibiograms, at least 48 hours are needed. Thus for acute infections this approach is unsuitable. Another disadvantage is the possibility of laboratory mistakes and consequently misinterpretation of the patient's situation and selection of a wrong therapy. However, this type of therapy and the use of narrow-spectrum, targeted antibiotics is necessary in treatment of subacute and chronic infections to expose the patient to as little side effects as possible and to reduce the risk of further resistance development.

- Empiric, untargeted, or calculated antibiotic therapy is the most common approach in nowadays prescribing of antibiotics. It is used every time a patient needs a therapy as soon as possible and the microbiology information is not yet available. Because of the lack of information or evidence a broad-spectrum antibiotic is often chosen to cover as many pathogens as possible.

2.2.5 Urinary tract infections

Urinary tract infections (UTIs) are diseases of the urinary tract caused by pathogens, mostly bacteria. Causative bacteria are mostly endogenous bacteria ascending through the urinary tract. These bacteria usually come from the patient's own colon flora [14, page 401] and cause an infection of the urethra or bladder.

Classification of urinary tract infections

There are many ways how to classify UTIs. The following ones are some concepts of common classifications:

- by anatomical localization: upper and lower UTI [16].
- by symptoms: pyelonephritis, cystitis, urethritis, asymptomatic bacteriuria (preliminary stage to UTI)
- by etiology: obstructive and non obstructive (obstruction in urine drain, concrements, ureteral stenosis, ureterocel, hypertrophy of prostata, congenital urine drain obstructions: hydronephrosis, megaurethra, urethra valves, other anomalies)
- by host-risk factors: uncomplicated and complicated with following risk factors:
  - obstructions in urine drain, which results in a high residual urine level
  - pregnancy
  - male sex
  - immune suppression
  - catheterization
  - paralysis (sclerosis multiplex)

  - diabetes mellitus

- by therapy response: responder and non-responder (whether a patient responds to a single-dose therapy or not)

Most of these classifications interact, most of the time it is necessary to combine them. For example, a common combination of these classification systems is to combine the anatomical and etiological classification.

Etiology

In 60–80% UTIs are caused by *Escherichia coli*. Enterococci, Proteus, Klebsiella, Enterobacter and *Pseudomonas aeruginosa* have each a 5% share [17, page 546]. The occurrence of *Pseudomonas aeruginosa*, Klebsiella spp and Proteus spp pathogens may sometimes indicate an urine-drain obstruction [16, page 197]. These bacteria are the most common causes for a UTI, however other bacteria may also occur.

Clinical signs and symptoms

Clinical signs and symptoms of UTIs differ based on localization of the infection. Symptoms of cystitis are:

- pollakiuria (increased urinary frequency and/or urgency)
- dysuria (painful urination or burning feeling in urethra)
- hematuria (blood in urine)
- nocturia (nightly urination)
- suprapubic tenderness

The difference to pyelonephritis is given by the occurrence of the following symptoms:

- fever ($> 38°C$)
- flank pain

Less specific signs are nausea, confusion (caused by high ammoniac levels), and fever.

Objective symptoms are assessed through a laboratory urine test. To determine these objective symptoms a midstream urine sample has to be taken from the patient. More than 10^5 bacteria in urine indicate a relevant bacteriuria. Furthermore, granulocytes occurring more than $20/\mu l$ are a pathologic sign [17, page 544]. Erythrocytes, leukocytes, and granulocytes can be determined by reagent strips.

Host risk factors

Complicated UTIs are defined by the presence of host risk factors. These factors and their influence on the UTI are described below.

- Pregnancy
  Attributed to the hormonal dilatation of the urethra and/or compression by the enlarged uterus, a urinary obstruction of the upper urinary tract may occur [16, page 391]. During pregnancy even asymptomatic bacteriurias, bacterial presence proven in urine, must be treated to prevent a development of a serious UTI.

- Catheterization
  Catheterization weakens local immunity by disrupting the inner lining of the bladder, which prevents bacteria to adhere. Voiding through catheter also disables the beneficial flushing effect of urination [18, page 30]. Another problem is the catheter itself, which may be contaminated by unproper handling. Patients with the need for catheterization mostly are immunocompromised, suffer from other diseases, and already receive treatment as well.

- Metabolic disorders—diabetes mellitus
  Diabetics are more susceptible to renal disorders than patients without diabetes [19]. Changed microflora, high glucose levels in urine, diabetic neuropathy—all symptoms of diabetes mellitus—increase the risk of bacteria settling in the urinary tract and result in UTIs.

- Urinary obstructions
  Any kind of urinary obstruction leads to urinary retention, which allows bacteria to multiply in urine. The urinary obstruction must be treated first to allow a free passage of urine.

- Gender specifics
  Due to the shorter female urethra women are more prone to UTIs than men. However, if a man must be treated because of a UTI, it bears mostly more complications than in women. UTI in men are mostly a consequence of another disease, which must be treated at first.
- Renal insufficiency
  Antibiotic substances are mostly eliminated through the kidney. Patients with renal insufficiency must therefore have their dosage of medication adjusted.
- Liver insufficiency
  Orally administered drugs pass the liver, where the bioavailability is reduced. Also in the process of elimination may liver insufficiency prolong the duration of this process, therefore the dosage must be adjusted especially in these cases.

Therapy

Therapy for each kind of UTI must include the following points [19]:

- eliminate reasons for urinary obstruction and other complicating factors, if possible
- generous fluid intake
- antispasmodic and antipyretic medication
- adjusting medication of metabolic disorders (e.g., diabetes mellitus)
- removal of noxae

Uncomplicated UTIs do not always need antibiotic treatment. Sufficient fluid intake however is critical (more than 3 litres). In severe cases of uncomplicated UTI one-day, or three-days antibiotic regimen have to be prescribed [17, page 548]. This type of therapy shows several advantages [16]:

- no or little side effects
- less damage to intestinal flora
- smaller probability for resistance development
- higher patient compliance
- faster discovery of complications

- lower costs

Complicated UTIs require prolonged antibiotic treatment also because of the possible occurrence of resistant bacteria.

2.2.6 Prescription process

Treat the patient and his disease not the microbiological finding. [14, page 319]

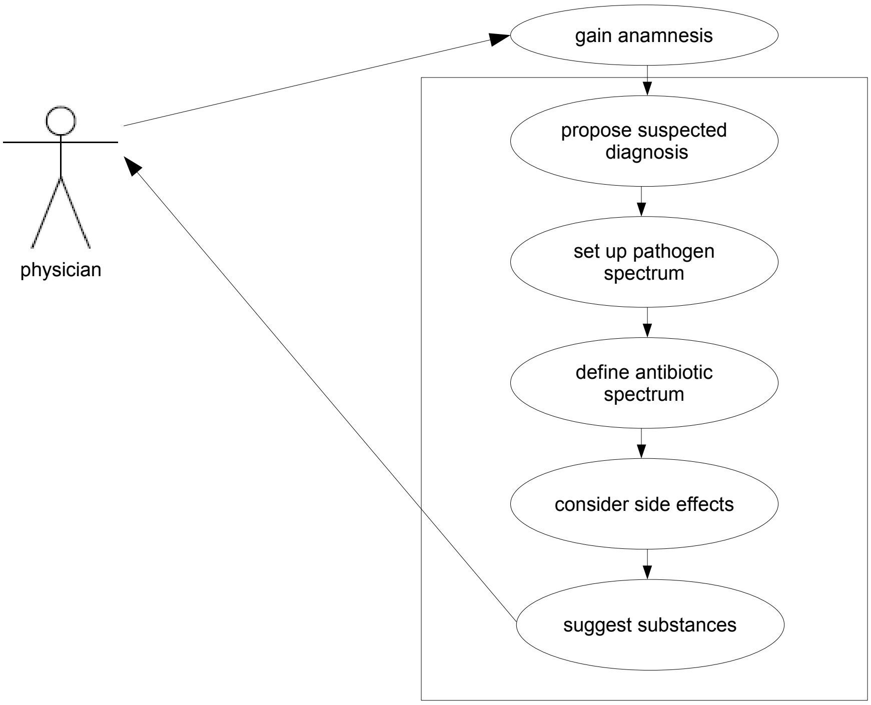


Figure 2.5: Workflow of the prescription process in unified modeling language (UML) notation

The textbook about medical microbiology and infectiology by Miksits et al. [14, page 320] summarizes the steps of the calculated therapy prescription process in five questions (see also Figure 2.5):

1. Based on anamnesis and clinical findings: is an infection suspected (suspected diagnosis)? Are therefore antibiotics indicated as treatment?

2. Which pathogen is most likely to occur (pathogen spectrum)?

3. Which antibiotic substances will be the most effective ones with respect to the local resistance situation (spectrum of effectiveness, spectrum of resistance)?
4. Are there any contraindications based on patient's characteristics (pharmacokinetics, side effects, allergies)?
5. What is the best way of administration to reach the infection and to achieve the minimal inhibitory concentration as soon as possible (dosage, application form)?

A more detailed view on the prescription process can be found in the book on antibiotic therapy by Wolfgang Stille et al. [17, page 391]. According to this, the following items must be considered during the prescription process in order to choose a correct therapy:

- clinical situation of the patient
- found microorganisms or typical organisms for the suspected disease and their resistances
- patient's past medical history and history of present illnesses
- properties of antibiotic agents
- least drug intake for patient as possible
- clinical experience and expert recommendations
- epidemiology as a whole, concerning local circumstances
- economical aspects and costs

The choice of antibiotic agents, as well as their dosage and the chance of success are determined by these items. However, often only incomplete information is available to the prescribing physician when the patient needs a therapy as soon as possible. Paradoxically the most important information from laboratory results, the causative microorganisms, are available ony 24 to 48 hours after sending patient probes to the lab.

In order to ensure best possible prescription and therapy, physicians should achieve the following four aims:

> ... maximize effectiveness, minimize risk, minimize cost, and respect patient's choice ... [20]

When trying to maximize effectiveness of therapy, physicians face a lack of information about the resistance behavior and the kind of pathogen as discussed before. The patient mostly shows specific or less specific symptoms. Based on these symptoms an expected pathogen spectrum may be estimated, e.g., *Escherichia coli* causes 80% of all UTIs.

To minimize risk the patient's other diseases, treatments, pregnancy, as well as allergies have to be considered.

Another problem physicians face is antibiotic resistance. For a long time this was a topic only in patients, who acquired an infection in healthcare facilities, where some bacterial strains had developed certain resistance against a group of antibiotics, e.g., MRSA. Nowadays also community-acquired infections may show resistances. For example, the increasing incidence of community-acquired methicillin resistant *Staphylococcus aureus* (cMRSA) is becoming a threat [21].

The physician receives information about the pathogen and possible resistance only after 24 to 48 hours. To cope with resistances, stronger and broader antibiotic agents are prescribed. These antibiotics usually have a higher toxicity and are more expensive than narrow-spectrum, specific antibiotics. The use and dosage of antibiotics has to be well considered because it may also lead to a development of resistance toward these substances if not used wisely. The balance between finding the most effective and risk-minimizing therapy has to be found.

Cost of medication is becoming a big issue in health care. Beside the primary responsibility for the patient, physicians also have to meet with wishes of healthcare providers and hospitals as their employers. Data to find the best and most cost-effective medication is not always available to the prescribing physician.

2.3 State of the art

Tools in the area of monitoring of nosocomial infections have been developed all over the world. For example the PTAH system [22] of the Slovenian research group, MONI [5] from the Vienna General Hospital in Austria, and commercial ones like PathFinder by Vecna [23]. However, decision support for antibiotic prescription can be found in the ICONS [24] consultant developed in Rostock, Germany, in a part of the HELP system [25] developed in Salt Lake City, Utah, USA, in a product of an Italian researcher team named MERCURIO [26] at the University Ferrara and in the TREAT [27] system developed at the University of Tel Aviv in Israel. The latter two systems have been finished recently thus not much information is available. We will describe the HELP system, which relies on a logic- and data-based approach, and the ICONS system which relies on a case-based approach.

2.3.1 Logic- and data-based approach

The hospital information system HELP (Health Evaluation through Logical Processing) was developed at the 500 bed LDS (Latter-Day Saints) hospital in Salt Lake City, Utah, USA. This system was developed in the 1970's and has been enhanced ever since as it became part of daily routine. The HELP system provides, besides a fully integrated patient medical record, various decision making capabilities for physicians in order to improve care. Since 1983 this application monitors antibiotic treatment and assists in infection control. The HELP system contains applications covering the following areas [25]:

- monitoring, predicting, and surveillance of hospital-acquired infections
- monitoring, surveillance, and assistance in antibiotic treatment
- monitoring of adverse drug events
- monitoring of duration of prophylactic antibiotic treatment
- isolation programs
- reporting of respiratory therapy infections

Automated antibiotic consultant

The application for monitoring antibiotic treatment is called automated antibiotic consultant, which consists of three major program parts [28]:

- extraction program—extracts data from the patient data management system (PDMS)
- data analysis program—analyzes extracted data
- physician interface program—visualizes the analysis for the physician and suggests a therapy

Data preprocessing
To be able to extract knowledge from the PDMS patient data are reduced to the most important information. Through stepwise logistic regression [28] six patient variables out of 23 were identified to represent a patient and the most probable pathogen:

- site of infection: blood, a wound, the lower respiratory tract, an abscess, and urine,

- patient's status: inpatient or outpatient,
- mode of transmission: community- or hospital-acquired infection (according to Centers for Disease Control and Prevention (CDC) definitions [29]),
- patient's hospital service: relevant treatment and diagnostic procedures, covered by the insurance,
- patient's sex,
- and age (in two categories: younger than or equal 50 years and older than 50 years)

Extraction program
This part of the consultant scans the database for records with an indicated infection. The output of this program is a file with patient data. This file called "infection file" consists of records of all patients from the PDMS for whom culture and sensitivity tests have been performed. It is further limited to patients with infections on the most common sites. To avoid redundant patient records only the first result for a patient with a certain pathogen at a given site of infection is recorded. Monthly the PDMS is scanned this way for new patient records. Entries older than five years are removed from the infection file.

Data analysis program
The purpose of this part of the program is to detect all occurring combinations of patient-specific variables in the infection file. The outcome of this program are three files: the six month, the five year, and the antibiogram file. The six month file contains records collected in the last six months, the five year file analogously contains data of the past five years. The antibiogram file provides sensitivity patterns of single isolated pathogens. This program is activated once a month after the update of the infection file. Each record in the five year and six month files contains the five most probable pathogens and the five most active antibiotic agents for a special combination of patient-related variables. To select the most probable pathogens and appropriate antibiotics on the basis of sensitivity patterns, two levels of logic are applied by this program [28]:

- first level of logic:
  Calculates probabilities of occurrence for all pathogens in the respective combination of the six patient-specific variables. From previous sensitivity tests antibiotic agents, which would cover all found pathogens for a given combination, are determined.

- second level of logic:
  This logic level uses rules, which were written by infectious disease specialists. These rules have precedence over the first level logic. They represent certain knowledge about antibiotic regimen—when to apply which antibiotic therapy.

Physician interface
The interface for the end user is called "antibiotic assistant". The physician interface is not a graphical user interface. The output appears as text on a command line. The interaction happens through keyboard entry. This program part provides four different information options [28]:

- patient-specific option:
  Here the physician identifies a specific patient and selects the suspected site of infection. The program retrieves missing relevant patient information of the current patient from the PDMS. Based on the information of the current patient a similar record from the six month and five year file is selected. Here a third level of logic is applied concerning the treatment-specific variables (allergies, organ dysfunctions, ability of oral intake, costs, etc.). From this data an antibiotic regimen is suggested.

- site-specific option:
  This option does not provide therapy suggestions for specific patients but for a certain site of infection.

- automated antibiograms:
  This option is an information module for cases, where the pathogen is known but its sensitivity patterns are still unavailable. Physicians may view ward-specific or hospital-wide antibiograms. These are presented in form of a ranking list beginning with the antibiotic regimen to which the pathogen is most sensitive. Further it provides information about the frequency each antibiotic has been tested with this organism and the average cost of the antibiotic agent per day.

- monographs:
  This option concerns the information about the antibiotic substances currently used at the hospital. It displays all commonly requested pharmaceutical information by physicians on antibiotic agents, for example, defined application use, generic and brand names, dosage recommendation, ways of administration, dilutions, infusion times, average cost per day, as well as drug allergy, and drug incompatibility [28, 30]. These monographs are regularly updated by the local pharmacy.

Logic levels

The knowledge in the antibiotic consultant is on one hand extracted from previously collected data (files of cases with most probable pathogens and antibiotic agents) and on the other hand knowledge from experts—which is not present in the data (use of antibiotics, contraindications) —has been codified into rules. This knowledge can be found on the previously mentioned logic levels:

- first level of logic:
  For every combination of patient-related variables the probability for each pathogen is computed and based on the sensitivity data the coverage of the pathogen spectrum is determined. In addition several susceptibility patterns have been built in, which are not routinely tested on sensitivity because of stable resistance behavior.
- second level of logic:
  The appropriate use of antibiotics is encoded. This level has precedence over the first level. Here the decision is made when to suggest a monotherapy and when and how to combine monotherapies.
- third level of logic:
  Treatment specific variables (known allergies, organ dysfunction, possibility of oral administration, costs) also determine the suggestion of therapy.

Therapy suggestion process

The single steps of the therapy suggestion process are:

1. retrieve patient-related information from the PDMS
2. retrieve similar records from five year and six month files
3. compute a therapy suggestion

Retrieval of similar records

To allow a conclusion from retrospective data, the retrieved group of records must be representative in number. For the retrieval process rules have been developed, which drop the selection variables age and sex, if unsufficient record counts (less than 15) are found [28]. Further if the six month file does not provide a representative sample size records from the five year file are used.

Computation of therapy

Based on retrieved similar retrospective records to the current patient and domain-specific knowledge the actual therapy is computed.

Next step is to select the five most active antibiotics from all retrieved records, excluding contraindicated ones. For this purpose the pathogen spectrum, a list ranked by probabilities of pathogen occurrences based upon the retrieved records, is computed. The logic determines the highest probability for each antibiotic regimen to be active against most probable pathogens of the patient. This probability is called coverage and is given in percent. The selection rules operate with coverage and cost of antibiotics, which is represented by the $ sign:

if highest coverage $> 90\%$
select least expensive within 6% and $\geq 90\%$ of coverage

A = 97% and cost = $$$
B = 92% and cost = $
⇒ select B

if highest coverage is between 80% and 90%
select least expensive within 3% and $\geq 80\%$

A = 89% and cost = $$$
B = 87% and cost = $
⇒ select B

if highest coverage is $< 80\%$
select least expensive with highest coverage

A = 79% and cost = $$$
B = 79% and cost = $$
C = 75% and cost = $
⇒ select B

These three rules concern the selection of monotherapies. If the program cannot find five antibiotics covering more than 80% of the expected pathogen spectrum, combinations of antibiotics are selected. Probabilities of gram-negative, gram-positive, and anaerobic pathogens determine this choice [28].

Evaluation of the system

A pilot study of this system was performed at a shock/trauma/respiratory intensive care unit (ICU). First physicians were asked to prescribe antibiotics as usual. Next the same task was performed with the use of the decision support tool. These two processes were monitored and the results evaluated. A difference in a time-and-motion study, measuring the amount of time used for retrieval of patient-specific information, was shown. The greatest advantage of this tool was discovered to be a fast information retrieval [31].

The automated antibiotic consultant was used two times a day on average. Most of the time the monograph and antibiogram option was used, followed by the patient-specific and the site-specific option. Importance of information about the ongoing sensitivity situation in the prescription process was demonstrated.

Reliability of the consultant was shown when in 94% of cases the program selected a regimen which covered the real pathogen spectrum. In 6% one or more pathogens were not sensitive to the regimen [28]. Physicians also reported a learning effect while using the system. They used knowledge from previous work with the antibiotic consultant also for other patients. 66% of physicians also stated that the information from the tool influenced their decision. However, they complained about having information only on the five most common sites of infection, as especially for the less common sites help would be necessary.

Strengths and weaknesses

- Strengths:
  The knowledge in this system contains expert knowledge from medical experts encoded in rules and knowledge about the ongoing situation at a hospital or ward from patient data. Through this feature, the system is able to adapt to a local environment. Sensitivity patterns have been built in for pathogens, which are not routinely tested on their sensitivity—because of constant susceptibility patterns to certain antibiotics despite exposure. Physicians must have information on how to deal with these organisms, therefore the information about sensitivity patterns and therapy recommendations has been included.

- Weaknesses:
  Only the five most frequent pathogens are taken into account. The overall view on the situation of occuring pathogens is not available. A graphical user interface is, especially in medical environment, faster to use than a textual frontend.

2.3.2 Case-based approach

At the University of Rostock in Germany a case-based decision support system, named ICONS, for antibiotic therapy was developed [24]. The target application field of this system are ICU patients, who have developed a bacterial infection as an additional complication. The main task of this tool is to advise physicians in cases, where no microbiology results are available. ICONS tries to help to provide best care and best prescribing strategy for every patient, independent of the knowledge of the physician on duty. Its decisions, based on medical background knowledge, form a calculated therapy. Patient information, like site of infection and patient's contraindications, as well as ward-specific information like the current resistance situation, allow to compute an expected pathogen spectrum and to present a suitable antibiotic therapy suggestion covering the whole spectrum. The antibiotic advisor also suggests an individualized dose of antibiotics based on patient's weight and age. For immunocompromised patients it uses an one-compartment pharmacokinetic model to compute the individual dose.

The ICONS system consists of a main module for antibiotics therapy decision support, a resistance surveillance module, and a module for computing individualized therapy dose.

Physician view

First of all the physician provides patient data in order to receive a suggestion for antibiotic therapy [24]:

- site of infection
- age, height, and weight of the patient
- select contraindications for the patient
- patient group (community-acquired, postoperative, hospital-acquired, immune deficiency)

Based on this information the program displays suggestions of antibiotics for this specific case. There is also a possibility to view other therapies. For example, unsuccessful therapies from previous cases, or monotherapies. The program allows physicians to create also their own therapies, which were not suggested by the program. For designing new therapies there is a need for additional information like the calculated expected pathogen spectrum, contraindications, and side effects of antibiotics; all this information is provided by the program. After choosing the final therapy, information about the initial doses, administration forms, defined daily doses, administration guidelines, and other pharmaceutical data are presented.

Calculating therapy suggestions

The first step in order to suggest a therapy is to determine the expected pathogen spectrum of the current patient. The authors of this system use the establish-refine inference: generate a general list and then refine. The first list consists of all possible pathogens, which are probable for this specific case. A second list of antibiotics is produced by refining the first one. The constraints for the refinement are patient's contraindications (antibiotic allergies, reduced organ functions, specific diagnoses, special blood diseases, pregnancy, and patient's age [32]), and site of infection.

Then in this second list antibiotics are presented, which, based on the resistance data, cover the whole calculated pathogen spectrum. For less difficult cases monotherapies, therapies consisting of only one antibiotic regimen, may be found. For more complicated cases, however, monotherapies are not advisable. Then antibiotics are combined considering their synergistic and additive effects through a set of rules. However, to produce an output of advisable therapies, each of this combinations of antibiotics has to cover the whole expected pathogen spectrum.

Case-based reasoning

The authors of this system decided to use CBR in order to speed up the solution process. Subjective knowledge for this domain is contained in cases. CBR methods solve the problem of fast changing knowledge through updating the already existing fund of knowledge with new cases. This technique consists of two major tasks: retrieval of the most similar cases and adaptation of these selected cases to fit the current case. For the first task many effective algorithms already exist, for the second task however, so far no general methods have been developed and thus this task stays domain-dependent. In the above therapy calculation process CBR methods are used to find all similar cases from previous data based on the parameters "patient group" and "site of infection".

Prototypes and prototype trees

A prototype is a generalization and summary of a set of cases. A prototype represents a typical diagnosis or disease situation. For an antibiotic consultant system this means, that a prototype can be seen as a typical antibiotic treatment of a diagnosis. In this system attributes of prototypes are contraindications of patients, which do not generate a solution but rather constrain a list of possible therapies [32]. Prototypes are classified in a hierarchical categorization tree—the prototype tree. There is always one prototype tree for one patient group and an affected organ. This system uses 20 different organ groups and has 5 defined patient groups. These trees are created as soon as new records occur in the system. From each prototype a specific pathogen spectrum corresponding to a typical antibiotic treatment can be deduced from background knowledge. Prototypes within prototype trees differ by their attributes, which are the patient's contraindications. Every prototype contains beside its cases and alternative prototypes also one empty case, which is displayed, when no adaptable case or prototype is found.

Generating prototypes

Prototypes are used to structure the case base, keep changing knowledge up-to-date and to erase redundant cases. Two threshold parameters are used [33]:

- minimum frequency: how often a contraindication has to occur in cases to be included into a prototype
- number of cases: how many cases are needed to fill a prototype. The lower this threshold is set the more prototypes will be created.

Cases with similar parameters are stored in one prototype until the "number of cases" is reached. Then the prototype is filled. Now contraindications occurring with more than "minimum frequency" are included into the prototype's description. Cases, which do not have additional contraindications compared to the prototype are erased. If a new contraindication is added to a prototype's description or if the frequency of occurrence of a certain contraindication changes, the contraindications and the antibiotic therapies are recomputed. A new alternative prototype is created, which is connected to the prototype. This new alternative prototype "waits" for enough cases to occur to become filled and thus to become a real prototype. Cases of the alternative prototype have to have at least one contraindication in common which is not included in the main prototype.

The idea of these two parameters is that they determine the behaviour of the system: how many and how often prototypes are released or how often new contraindications are added to already existing prototypes.

Retrieval of cases

This task consists of finding the appropriate prototype tree based on site of infection and patient group. The retrieval process consists of three successive steps. At first, an appropriate prototype tree based on the two parameters is found.

Now tree-hash retrieval algorithms are applied to nominal-value contraindications (the algorithm of Slotter, Henke, and King [32]) and to integer-value contraindications (the similarity measure of Tversky [32]).

Last but not least the adaptability criterion is applied, to ensure that the found case or prototype does not have more contraindications than the current case. Otherwise it would cut down the therapy solution set unnecessarily. This criterion guarantees the adaptability of the prototype to the current case [33].

Adaptation

In this system three cases of adaptation occur:

- adapting retrieved cases to current case (defined by site of infection and patient group),
- adaptation criterion to ensure that no cases with more contraindications than the current case are retrieved. This adaptation reduces the first set of advisable therapies by the contraindications of the current patient.
- The last kind of adaptation occurs when sensitivity information has changed. In this case prototypes are recalculated with new records.

Evaluation of the system

The evaluation of the system's user-friendliness and user-machine interaction was assessed through a questionnaire. First, experienced physicians were given certain fictional cases requiring an antibiotic therapy and were supposed to solve them on their own.

After a short training on the ICONS system this group of physicians was asked to solve those cases with the system. Many physicians agreed with the suggested therapies, but a few disagreed completely [34]. However, the general idea of trying to find an expected pathogen spectrum and to match an antibiotic therapy, which covers the whole spectrum based on local resistance data, was approved by all physicians as very good. A retrospective test with microbiological results of 2 years proved a coverage of therapies recommended by physicians and by ICONS of 70%. In 18% of all cases ICONS recommended a different therapy than physicians.

Strengths and weaknesses

- Strengths:
  This system inherits knowledge about unsuccessful therapies. These are therapies, which for various reasons did not show the expected therapeutic effect. The authors used CBR to speed up the whole process of finding an appropriate therapy. They compute synergistic and additive effects of antibiotics when computing the second antibiotics list. In the calculation of empiric therapy site-specific information about the current resistance situation is integrated [34]. Another important information, the cost of a treatment regimen, is also displayed. This system uses weight and pharmacokinetic calculations to determine the correct dose of a recommended antibiotic regimen.

- Weaknesses:
  Trying to apply the system at another site showed a decrease in correctness. Used antibiotics, site-specific resistance and the regimen of combinations of antibiotics had to be updated or totally changed [34]. A negatives list, a list of antibiotics, which are dangerous for the patient and not to be administered for some reason or another, is not displayed although data is available. A CBR system's performance strongly depends of its retrieval and similarity measures. If the system's complexity grows, these algorithms may not be sufficient anymore.

3 Material and methods

In this chapter material and methods, which were used to design and implement the system are described. The goal of this chapter is to enable the reader to reproduce the results gained.

First, description and analysis of retrospective data used for this thesis are given. Then algorithms, which were developed based upon the data analysis, are discussed. Further knowledge application and representation are explained. How recommendations are generated and resistances calculated is described at the end of this chapter.

3.1 Problem definition

Our task was to build a prototype of a decision support system to assist physicians in the management of antibiotics therapy by providing an estimated pathogen spectrum and suggesting antibiotics suitable for treatment.

During the prescription process of antibiotic regimen the physician faces several problems. The absence of information about the etiology of infection as well as its resistance behavior do not allow a precise prescription. Therefore empiric therapy, based on clinical signs only, is most often used and adjusted after the missing knowledge becomes available. The goal is to design and implement a decision support system by taking retrospective data into consideration. From the retrospective data relevant knowledge for the current patient should be extracted to replace temporarily the unavailable information about the pathogen and its resistance behavior. Furthermore antibiotic-specific knowledge will be used to generate a list of recommended antibiotic regimen.

From the described problem following goals emerge:

- correctness: information from retrospective data must be shown correctly
- explanatory component: the physician must be able to re-enact recommendations

- data security: use of anonymized patient data
- specialized medical field: diagnosis of UTIs
- knowledge-based system: separate knowledge base from knowledge processing
- user friendliness: create a user-friendly interface for the application
- interactivity: processing of retrospective data must be in real time

3.2 Data

The application is based on retrospective data, which were provided by the Hygiene Department of the Vienna General Hospital. These data are usually used for monitoring the development of nosocomial infections in the hospital.

3.2.1 Description

The database consists of patient records, each row representing a record of one patient, for whom a specimen has been taken. A description of all columns follows:

Original columns:

- first name
- last name
- date of birth
- age
- protocol number of laboratory finding
- date of specimen taken
- ward or department
- kind of specimen
- name of found pathogen
- sensitivity pattern: columns for all antibiotics tested

The first three columns are the patient's identification (the patient's first and last name and date of birth). Additionally, for the identification of the specimen, an identification number (protocol number), the date of taking the specimen as well as the kind of specimen taken are contained. The patient's age is computed based on the date of birth. Another column in this table is the ward or department column. The most important part of this table however is the found pathogen and its sensitivity pattern. The sensitivity pattern may contain either empty fields, or values of sensitivity (resistant, sensitive, intermediate, and not indicated).

Three columns have been added later:

- gender,
- specimen category, and
- ward category.

The patient's sex has been added later based on the first names. The columns ward category and specimen category classify wards and specimen into coarser categories.

3.2.2 Preprocessing

Each laboratory finding of a patient is recorded automatically. Therefore the received data contains also corrupted records, follow-up isolates, and duplicate findings. These have to be eliminated not to distort the statistics. Two kinds of data cleaning within the program are distinguished: institution-specific data screening and general data screening. Institution-specific screening eliminates documentation errors. Next the data of each record is screened for plausibility in the course of general screening. In this screening values are checked, if they fulfill certain criteria e.g., age must not be higher than a defined highest age of 120 years. In case of a typing mistake, when a date of birth or age value has a higher value, the record is skipped. A valid record also must contain a value in the "found pathogen" field. Next, the sensitivity patterns are checked. A value of the sensitivity pattern must be one out of five defined values (resistant, sensitive, intermediate, not indicated, or none (without a value)). In case of a different value the particular record is skipped. To obtain as much knowledge as possible from this retrospective data the sensitivity pattern must contain values that are different to none. This assumption is based on the belief that each pathogen, if tested, is at least fully sensitive to all tested antibiotic regimen. If sensitivity values are missing, it must be assumed, that either a documentation error occurred, the pathogen has not been tested, or the sensitivity pattern was not recorded. Thus records with "empty" sensitivity patterns are not used for further computation.

From the records following data is needed: date when the specimen was taken, patient age, patient sex, kind of specimen, specimen category, ward and ward category, found pathogen and the respective sensitivity pattern.

Follow-up isolates occur when from a patient the same kind of specimen is taken repeatedly at different times and the pathogen and its sensitivity behavior do not change. These records—the later taken, thus younger specimen—are erased from the data. A change in sensitivity behavior is defined as a difference in the sensitivity value of a tested antibiotic or a different set of tested antibiotics.

Duplicate records must not be used for further computation. They are specified by same patient data, same kind of specimen, same pathogen, and same sensitivity pattern and would cause higher numbers in pathogen counts. Duplicate records may be follow-up isolates or documentation errors.

3.2.3 Statistics

Retrospective data used for this master thesis are produced by a program for monitoring of nosocomial infections, named MONI, at the Vienna General Hospital. By analyzing the retrospective database the concepts of this application were developed. In this section the statistics, upon which these concepts are based, are shown. Only data from the emergency station and ward of the years 2006 and 2007 were used. From the total of 989 records we filtered 180 with empty sensitivity patterns.

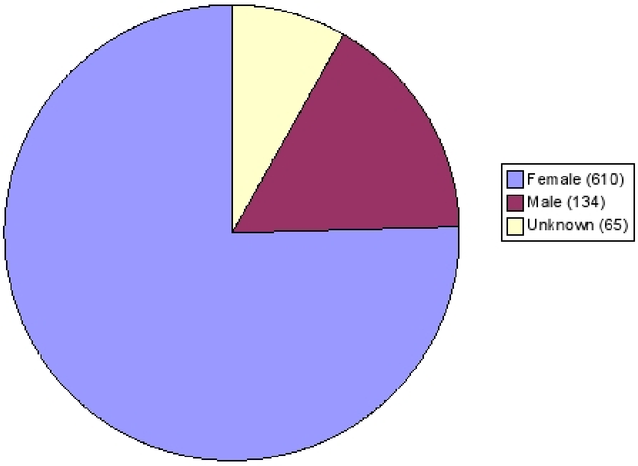


Figure 3.1: Sex distribution of patients in the retrospective database

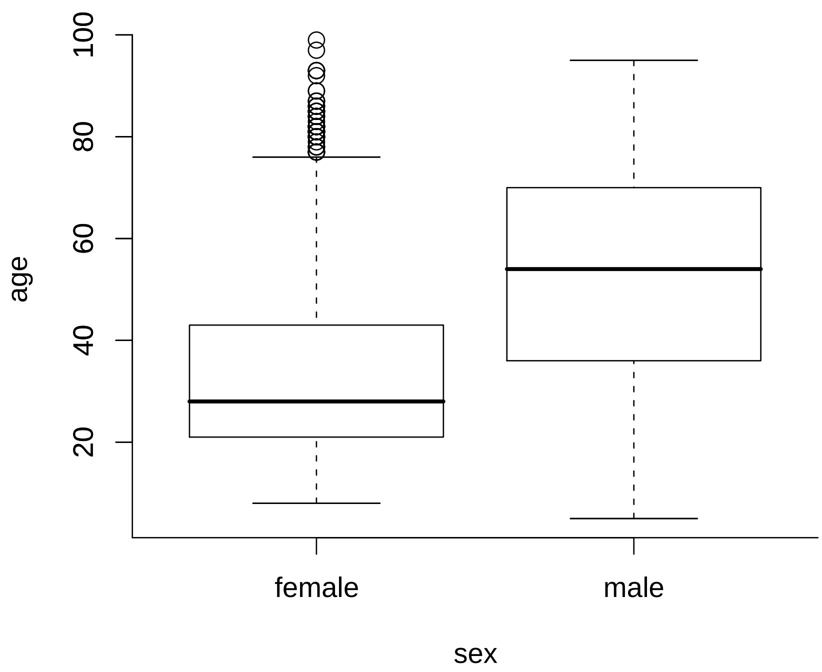


Figure 3.2: Sex-related age distribution of patients in the retrospective database

In Figure 3.1 the sex distribution of patients in the database is shown. This value, sex, was not originally contained in the data. Because it was added later, based on first names, some records were not assigned a sex because of a foreign first name. We do not use these records for further computation. If we added them to either group of sex, we would possibly falsify the statistic. Women were four times more often medicated at the emergency station and ward than men.

Age and sex distribution of patients

In medical literature it is known, that at an emergency station more young women with a UTI go to see a doctor than men. While men acquire a UTI as a side effect of an obstruction in urine drain caused often by a prostate disease or catheterization being already fairly advanced in years, women often develop a UTI in an earlier age during pregnancy or due to sexual activity (see also Figure 3.2).

We statistically analyzed these facts and performed tests to find out, whether these age distributions are significantly different. The question to test is whether the average age of the group of female and male patients is the same. For this test we used the previously mentioned data with 610 records of female and 134 of male patients.

$$H_0 : \mu_{female} - \mu_{male} = 0 \tag{3.1}$$

We performed a t-test with the level of significance being 0.05. The mean age in the group of female patients was 35.6 and in the group of male patients 53.1 years. The resulting p-value was less than 2.2^{-16}. The zero hypothesis was rejected at the level of significance being 0.05. The alternative hypothesis, average ages of female and male group are not equal, was accepted (see Appendix C).

Pathogen spectrum: influence of sex

By analyzing pathogen spectra we found that the factor sex influences the distribution of pathogens. For this reason we show a table with "female" and "male" pathogen spectra, which are counts of pathogens found in female or male patients.

Table 3.1: Pathogen spectra of female and male patients

pathogen name	**female patients**		**male patients**	
	count	in %	count	in %
Escherichia coli	535	87.7%	79	59.0%
Enterococcus faecalis	23	3.8%	21	15.7%
Klebsiella pneumoniae	15	2.5%	7	5.2%
Proteus mirabilis	12	2.0%	6	4.9%
Escherichia coli ESBL	6	1.0%	1	0.8%
Citrobacter koseri	5	0.8%	2	1.5%
Staphylococcus aureus	4	0.7%	1	0.8%
Pseudomonas aeruginosa	3	0.5%	3	2.2%
α-hemolytic streptococcus	2	0.3%	1	0.8%
Enterobacter cloacae	1	0.2%	3	2.2%
Enterobacter aerogenes	0		2	1.5%
Morganella morganii	0		2	1.5%
⋮	⋮	⋮	⋮	⋮

In Table 3.1 the distribution of pathogens among men and women is slightly different. Women have a large share of *Escherichia coli* while men have two large groups of pathogens when considering the percentage of these two groups: *Escherichia coli* and *Enterococcus faecalis*. Is this difference in *Escherichia coli* occurrence statistically significant? We decided to test this hypothesis with a binomial test at the significance level 0.05. We used the following data: the count of *Escherichia coli* pathogens found in male patients (79), number of all male patients (134) and the female occurrence of this pathogen (87.7%, see Table 3.1). With this input (see Appendix C) we could reject the zero hypothesis and accept the alternative hypothesis, which states that the occurrence of *Escherichia coli* pathogens in male patients is less than the occurrence in female patients at the significance level of 0.5%.

The result of this statistical test shows that pathogen spectra are sex-specific, thus the variable sex is important for the selection of an appropriate patient group to compute the expected pathogen spectrum for the current patient.

Pathogen spectrum: influence of age

Another factor influencing the distribution of pathogens in patients is age. Since we have shown in the previous section the influence of the variable sex, age groups were compared only within the respective sex. First we compared two groups of female patients. The first group, aged 30 to 39 years, comprised 110 records, the second group (50 to 59 years old) comprised 37 records.

Table 3.2: Comparison of pathogen spectra of female patients: age groups 30–39 and 50–59

pathogen name	**female 30–39**		**female 50–59**	
	count	in %	count	in %
Escherichia coli	101	91.8%	32	86.5%
Enterococcus faecalis	3	2.7%	0	
Citrobacter koseri	2	1.8%	0	
Enterococcus faecium	1	0.9%	0	
Proteus mirabilis	1	0.9%	3	8.1%
Staphylococcus aureus	1	0.9%	0	
Enterobacter cloacae	1	0.9%	0	
Klebsiella pneumoniae	0		2	5.4%

By comparing occurrence of various pathogens no significant difference was found. The order of pathogens is different. However, this fact may be attributed to coincidence.

Next we compared two groups of male patients. The younger group aged 30 to 39 years comprised 18 records. The older group, aged 60 to 69 years, comprised 23 records.

By comparing these male age groups it could be shown that the respective pathogen spectra are not identical. The occurrence of some pathogens differs significantly. The main difference

Table 3.3: Comparison of pathogen spectra of male patients: age groups 30–39 and 60–69

pathogen name	**male 30–39**		**male 60–69**	
	count	in %	count	in %
Escherichia coli	16	88.9%	11	47.8%
Enterococcus faecalis	1	5.6%	5	21.7%
Klebsiella pneumoniae	1	5.6%	0	
Pseudomonas aeruginosa	0		2	8.7%
Klebsiella oxytoca	0		2	8.7%
Proteus mirabilis	0		2	8.7%
Staphylococcus aureus MRSA	0		1	4.4%

between the spectra lies in the fact, that the pathogen spectrum of the younger group is classified as simple by our rules, while the pathogen spectrum of the older group is already classified as complicated.

These comparisons show the difference between age groups. For female groups no big difference could be shown, however a change in pathogen spectra could be detected. The impact of coincidence on this change cannot be rejected based upon the small sample size of the older female age group.

Based upon these results we cannot decide whether the variable age is an important factor in female patients. However, for male patients the situation is different. We were able to show a change in the severity of the pathogen spectra for the two male age groups: the younger group represents a simple case, the older group a complicated one. This change is very important as it indicates a shift towards complicated cases, which must be treated differently than simple cases. For male patients the variable age is necessary.

We could neither reject nor prove the influence of age on pathogen distribution in women, but we have shown its importance with respect to male patients. Therefore we will consider this variable in the group selection process for the current patient.

3.3 Condensing retrospective data

To establish a correct prescription for the current patient, the following data is of greatest importance for the physician: the pathogen and its sensitivity pattern as well as patient information like sex, age, kind of specimen, and ward.

The information that can be gained from the retrospective data, is an expected pathogen spectrum and the respective behavior of pathogens specific for the current patient. The identifying patient variables in the retrospective data are:

- sex
- age
- kind of specimen
- ward or department

These variables correspond with the variables used in the HELP system [28]. Based on them records as similar as possible to our patient are selected and the respective pathogens counted to make up the expected pathogen spectrum.

3.3.1 Condensing data

We would like to avoid reading all data every time when retrieving an expected pathogen spectrum. Therefore all records are grouped by taking three out of the four defining variables into consideration: sex, specimen category, and ward or department category. We defined the process of finding similar data for the current patient by starting at a large group of records, which we specify by dropping records which do not correspond with the abovementioned three defining variables.

The forming of groups was done according to specimen category instead of kind of specimen. This coarser grouping allows us to decide later how specific specimen are to be handled. The same method was applied to the ward or department column.

The value age needed some more thought. Our first idea was to already group data to certain age groups at this stage. The reasons we decided not to, were the speed of the grouping process and the loss of dynamic retrieving of records based on age. By introducing age groups they would have to be computed twice. The first time in the condensing process—assigning each record to an age group—the second time for the current patient to specify from which age group similar records should be selected. Also the number of condensed groups would be far larger depending on the number of age groups used. Therefore we do not include age as a variable in the grouping process and consider it only in the next processing step.

Compared to the number of records we obtained a much smaller number of groups. The number of groups is determined by the number of values of each of the defining variables. There are two possible values for the sex column, two in our case for the category of specimen column (blood and urine) and also two for the ward category column (emergency ward or department, or other). Thus we receive at maximum 8 different groups.

These groups are based on changing data. It is not sufficient to build these groups only once. How often records are grouped depends on the speed the data change. Similar systems build such data excerpts on a monthly basis. In our system the physician may decide himself how often a grouping is done.

3.3.2 Group content

> The goal of cluster analysis is to partition the observations into groups ("clusters")
> so that the pairwise dissimilarities between those assigned to the same cluster tend
> to be smaller than those in different clusters. [35]

The first measure to speed up the process of finding the expected pathogen spectrum was to group records. The second measure was to define labels for these groups. Each group is made up of records of a certain combination of the three defining variables (see 3.3.1). This combination makes up the label of the group, by which the search for the appropriate record group is run. To the outside world a group may be seen as a black box labeled by a combination of these defining variables. Inside the group all necessary information of all records is stored. Names of patients, protocol numbers, and dates of birth are no longer relevant for further processes. The structure inside the group is not a regular table structure as in the original data. While in the original data table single records are read horizontally, in a group a record can be found by reading vertically. This way it is easier to return column values without processing all records. For better understanding: single records are of no use for the expected pathogen spectrum, rather it is the statistic over a group of records which makes up the necessary information. Based on these thoughts and to enhance performance this structure was introduced. The content of a group is all information not included in the label but contained in the records or generated during data processing. A group is made up of lists of the following data:

- patient ages
- kinds of specimen
- dates when the specimen were taken

- wards
- sensitivity patterns
- pathogens
- timestamps

The list of timestamps is generated by the program. All other lists are built from data contained in the records.

Data provided by a ward's information system may be several years old. The timestamp allows to quickly find a record in a certain time interval without having to compare the date of each record with the actual date. Five time intervals have been specified: six months, one year, three years, six years, and more than six years. In the group every case is assigned a timestamp. We assume that this application has access to current data. Therefore the timestamp is computed from the current day backwards. The initial data we received were not up-to-date data therefore the timestamp was computed from the most recent date backwards. Data used to generate an expected pathogen spectrum are within the one-year interval, data for resistancies within the six-month interval. Further optimization was done with respect to the storage of sensitivity patterns. Sensitivity patterns are defined by four different values as mentioned before. These values were initially recorded as strings. It was highly inefficient to parse these strings every time they needed to be evaluated. Also comparing string values is less performant than comparing numbers. Therefore the values of the sensitivity patterns were assigned to numbers as follows:

- 1 – not indicated
- 2 – intermediate
- 3 – sensitive
- 4 – resistant

When displaying results these numbers are changed back into strings. These groups are written to an XML file and read every time the program is started.

3.4 Patient group selection

For our current patient we would like to estimate the most probable expected pathogen spectrum. We know the aforementioned variables of the patient. From the retrospective records we select those which were taken at the same hospital ward. Further we limit these records to those with the same kind of specimen taken and sex of the current patient. If this group were homogenous in the occurrence of pathogens we would be able to omit the age variable. As already shown in section 3.2.3 this is not the case. Thus we try to constrict this group of records further to an appropriate age group.

3.4.1 Dynamic age groups

The size of such a group—the amount of in this way selected cases—varies. While some groups are very poorly populated—particulary the tails of the age curve—others are very large. Let us imagine the age projected on the x axis and the y axis specifying the number of cases with this age to show how many patients have been treated at the respective age of the current patient before. This number though might not be representative. It also could be a local minimum. Therefore the respective age is spread to a surrounding age group. There are several possibilities to obtain this group. The following available variables could help to obtain an appropriate age group representing age-specific data:

- size of the condensed group
- minimal group age
- maximal group age

By setting or calculating the following variables we obtain an age group:

- upper and lower age limit
- minimum width of group
- minimum number of cases

Our first approach was to define the age limits. Setting the lower and upper age limit was based on the number of cases available between these two threshold values. As the minimum number of cases was per default 50, the age limits were set to those values, where the number of cases found was closest to and at least 50. This method allows us to have dynamic thresholds of the age group. The advantage of this method is that the group is stretched in poorly populated sections and narrowed in highly populated ones. However, we had to define a minimal group size which is not dynamic. Especially in groups with fewer records it may be disadvantageous when the minimal group size is bigger than or equal to the actual group size. In this case the age group takes up all cases and may stretch across the whole age scale.

Let us assume the number of cases is uniformly distributed over all ages. This assumption is a simplification of a very complicated reality. For these purposes however it will satisfy our needs. So, for a patient group size (g) of 200 records we would have 2 cases in each age slot (each age slot represents a timespan of a year; e.g., from 0 to 1 year olds) if our maximum age (a_{max}) is 100 years. Our first approach was to set a minimum age group size (g_{min}). We set it to 70. According to equation 3.2 we calculate based on g_{min} the span of the group ($span$) in which this number of records should be found.

$$span = (g_{min} \times a_{max}) \ / \ g \tag{3.2}$$

If we go on calculating with this default value we receive the results shown in Table 3.4. In our defined minimum age group we have 35 age slots. The number of age slots tells us how broad our final age group could become. The maximum age group width ($width_{max}$) is computed based on equation 3.3.

$$width_{max} = \lceil (span - 1) \ / \ 2 \rceil \tag{3.3}$$

To receive the amount of age slots to be added to/substracted from the current patient's age, the span of the group is diminished by one, and the result divided by two. The ceiling function rounds up to the smallest integer greater or equal this value. Table 3.4 comprises two real situations found in the retrospective data.

Table 3.4: Calculation of dynamic age groups with defined minimum age group size = 70

	time period	**cases**	**cases/age slot**	**span**	**max width**
model situation		200	2	35	+/- 17
male, 33, urine, emergency ward	3 years	178	1.78	39.33	+/- 20
male, 33, urine, emergency ward	1 year	74	0.74	94.59	+/- 47

By means of this example the disadvantage of this approach in case not enough records make up one patient group becomes visible. Especially in the last row of the table the age limits are useless: the lower age limit is -14 and the upper age limit is 80.

By computing with a given minimum size of a retrieved group and the case number being smaller or equal this minimum size, we would finally obtain all records of this patient group without a specified age limit in the retrieved group.

A more efficient approach is to define the maximum group width ($width_{max}$) and to compute the age group span *span* on the basis of equation 3.4.

$$span = width_{max} \times 2 + 1 \tag{3.4}$$

Furthermore the minimum age group size g_{min} is computed by considering the patient group size g, the maximum age a_{max} (100 years), and the age group span *span*, see equation 3.5.

$$g_{min} = \lceil (g/a_{max}) \times span \rceil \tag{3.5}$$

The age group span is the maximal allowed range in which similar records are to be retrieved for the current patient. We approximately determine the number of records found in one age slot, regarding the distribution of the age of patients. This approximation is a very rough one, since we assume an equal distribution of patients in every age. For parts of the real age distribution with lower patient counts we obtain stretched retrieved groups, for higher counts more narrow groups. This record count per age slot is multiplied by the computed span of an age group and rounded up to the smallest integer greater or equal this value.

Table 3.5: Calculation of dynamic age groups with a defined maximum width of +/- 15

	time period	cases	cases/ age slot	mingroup size
model situation		200	2	62
male, 33, urine, emergency ward	3 years	178	1.78	56
male, 33, urine, emergency ward	1 year	74	0.74	23

The implementation of the group selection is described in section 4.4.2.

Alternative methods

On the way to the method described above we considered other methods to determine plausible age limits for a patient group. The first method that a technician would suggest, is to use some kind of clustering method. Beside the problem of defining the correct local minima/maxima, these groups strongly depend on the granularity of the age distribution. We compared groups generated by a clustering method with those proposed by the physician and they were not identical, not even similar.

The method of determining groups by physicians depends on the suspected diagnosis and the physician's experience. Especially in the field of UTI age groups emerge based on risk groups and risk factors. For example, it is generally known that young, sexually active women have more often uncomplicated cystitis than other women [18, pages 31–32]. Therefore it would be possible to determine age limits based on medical expert knowledge. However, a major disadvantage in this method is that age limits in real world change and these changes would have to be updated in the program manually. We decided not to use this method because it is not dynamic and therefore does not visualize possible pattern changes in data.

3.4.2 Statistical view of data

During the prescription process the information about the local bacterial situation as well as the response to antibiotic treatment may alter the final decision. With nowadays patient information systems this knowledge needs to be condensed to be understandable at the first glance and be of use. Other station-specific statistics such as counts of probes taken, numbers of sex or age groups, pathogen counts, and their sensitivity patterns may be compiled. For the prescription process the important information lies in the specimen-specific pathogen counts and sensitivity patterns.

1. patient-specific expected pathogen spectrum
2. accumulated sensitivity patterns
3. resistance change detection

Patient-specific expected pathogen spectrum
This view computes the possible pathogens for the current patient. In the retrospective data the values sex, age, specimen category, and ward category define a patient's situation with a given suspected location of infection at a given ward. As the pathogen situation does not change very quickly, records which are at maximum one year old are taken for this view. The pathogens of these records are counted and their occurrence in this group is displayed as a percentage value.

Accumulated sensitivity patterns

The second view is an accumulated sensitivity pattern for a pathogen. Here for each antimicrobial agent tested, resistance reactions are averaged. The result is a table of antibiotics tested for a certain pathogen and its average sensitivity and resistance values. This gives a quick overview of the state of the pathogen in the current patient group. As the sensitivity behavior of these pathogens might change quicker than their occurrence, here only data which are at maximum half a year old are used.

Resistance change detection

Each pathogen has a sensitivity pattern. This sensitivity pattern contains valuable data: against which antimicrobial agents has the pathogen been tested, what was its reaction. Pathogens belonging to one strain do not always have the same resistance behavior. Some bacteria have special resistance strains, which can be found by checking the resistance of these bacteria to certain antibiotics. If values in the sensitivity pattern begin to show resistance on these antibiotic substances, a new strain of resistant bacteria may be emerging.

3.4.3 Interpretation of statistical results

The patient-specific expected pathogen spectrum shows the possible occurrence of pathogens for a certain patient group. The interpretation of the spectrum is important for the prescription process. Patient groups in which one pathogen occurs very often have to be treated differently than groups where all pathogens are equally distributed. In the first case the therapy can be assessed based on the knowledge about the leading pathogen. In the second case the therapy should cover the whole spectrum to ensure full coverage. Next in the accumulated sensitivity patterns the physician sees what on average may be expected of this pathogen. It is necessary to know, that not all specimen of one pathogen have been tested on the same set of antibiotics. Therefore a count of tested pathogens for each antibiotic is added as a column.

3.5 Spectrum analysis

For each patient a special, individual therapy is necessary. Therefore, to include this diversity, we analyzed computed pathogen spectra with the help of a physician and found two different situations. The first situation (see Table 3.6) reflects a simple case of UTI. The UTI is mostly uncomplicated and one pathogen in the expected pathogen spectrum occurs in more than 50% of all cases. Next presumption for such a situation is that the share of the second most occurring pathogen is less than a third of the percentage of the first pathogen. Then the therapy is chosen based on the average sensitivity pattern of this most occurring, also called the leading, pathogen.

The second situation (see Table 3.7) reflects a patient with a complicated infection. These are mostly UTIs with complicating factors and an almost equally distributed pathogen spectrum. An equally distributed pathogen spectrum states that many pathogens are likely to occur and therefore a combination therapy considering all of these pathogens is indicated. To compute a therapy of a combination of antibiotic regimen complicated algorithms are required to involve synergistic and antagonistic effects. Therefore we decided to reduce the decision support in this situation to common recommendations for UTI treatment found in literature.

Table 3.6: Expected pathogen spectrum for a young woman of 25 years having a UTI treated at the emergency station

pathogen	**number of cases**	**in percent**
Escherichia coli	73	80.2%
Enterococcus faecalis	7	7.7%
Proteus mirabilis	4	4.4%
Klebsiella pneumoniae	2	2.2%
Pseudomonas aeruginosa	1	1.1%
Staphylococcus saprophyticus	1	1.1%
β-hemolytic streptococcus gr.B	1	1.1%
Citrobacter koseri	1	1.1%
Enterococcus faecium	1	1.1%

Table 3.7: Expected pathogen spectrum for an 88-year-old male patient at the emergency station with suspected site of infection being the urinary tract

pathogen	**number of cases**	**in percent**
Escherichia coli	24	32.9%
Enterococcus faecalis	15	20.6%
Pseudomonas aeruginosa	6	8.2%
Klebsiella oxytoca	6	8.2%
Klebsiella pneumoniae	4	5.5%
Staphylococci koag. neg. (not saprophyticus)	3	4.1%
Morganella morganii	2	2.7%
Citrobacter freundii	2	2.7%
⋮	⋮	⋮

To be able to differentiate between simple and complicated cases special fuzzy rules were developed. The rules themselves are listed in Appendix E.

3.5.1 First pathogen rule

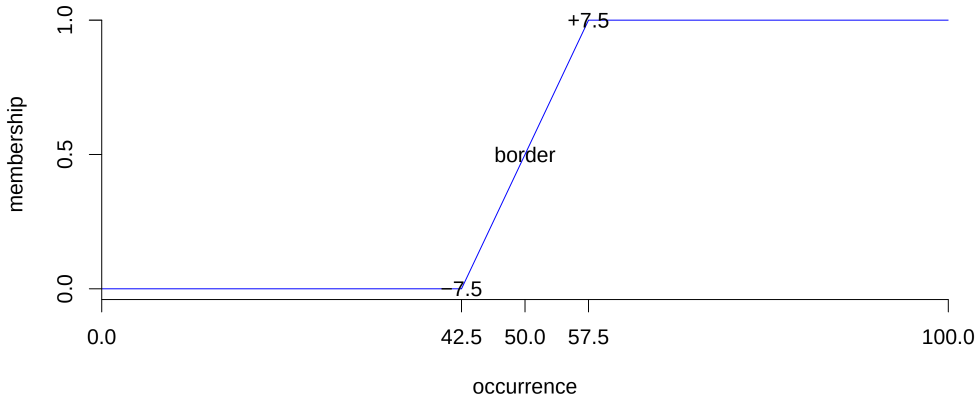


Figure 3.3: Fuzzy function to determine the significance of occurrence of the first pathogen

The "first pathogen rule" tries to find out whether the occurrence of the first pathogen is significant. The first pathogen is significant when its occurrence in the expected pathogen spectrum is more than 50%. We used a fuzzy function, see Figure 3.3, with a border at 50% and a width of 15.

3.5.2 Ratio rule

The first pathogen rule however is not sufficient. If we found two pathogens, each occurring in 50% of cases the first rule would be true, the spectrum nevertheless would be equally distributed, indicating a complicated patient. Therefore we introduced a second rule based on the ratio of the occurrences of the first two pathogens in the pathogen spectrum. This is also a fuzzy rule but an inverted one, see Figure 3.4.

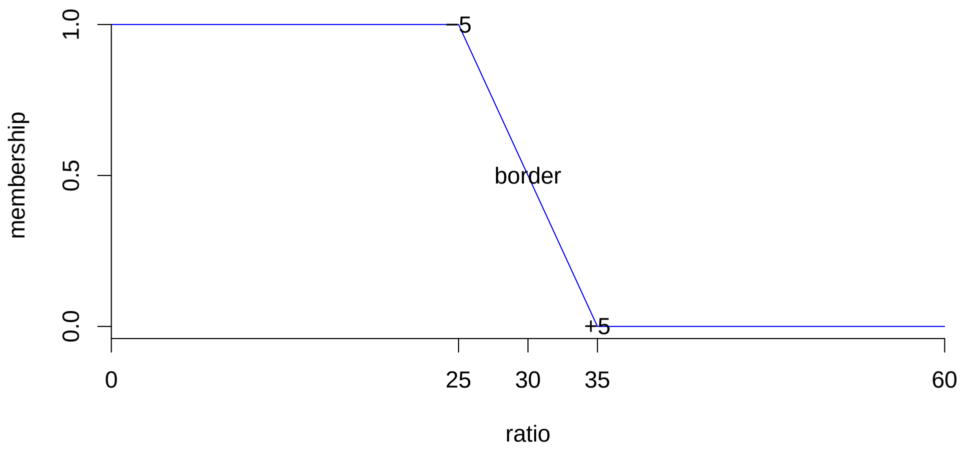


Figure 3.4: Fuzzy function determining whether the ratio of the first and second pathogen is less than 30%.

3.5.3 Decision on the situation at hand

To classify a simple case the rules as specified in equation 3.6 are run. $N[i]$ defines the pathogen occurrence at the i^{th} position in the expected pathogen spectrum. Occurrences of pathogens are defined as a value between 0 and 1. This equation is basically the summary of the previously discussed rules.

$$\begin{gathered} N[i] \quad \{N[i] \in R | 0 \leq N[i] \leq 1\} \\ F_{simple} = (N[1] >_{fuzzy} 0.5) \bigcap_{fuzzy} (N[2]/N[1] <_{fuzzy} 0.3) \\ F_{simple} \in [0,\, 1] \ \textit{whereas} \ [0,\, 1] = \{x \in R | 0 \leq x \leq 1\} \end{gathered} \tag{3.6}$$

Applying these membership functions to the pathogen spectrum we receive two values between 0 and 1. Two values, because we apply each rule separately. To decide the complexity of the case's situation we fuzzy-AND relate the two membership results and receive c, a value between zero and one.

The final result is then determined as given in equation 3.7:

$$c \in [0,1] \qquad \begin{aligned} c &\leq 0.25 \rightarrow \mathit{complicated\,situation} \\ c &> 0.5 \quad \rightarrow \mathit{simple\,situation} \end{aligned} \tag{3.7}$$

If the value is higher than 0.5 it is classified as simple situation with a significant leading pathogen. If the result is below 0.25 it represents a complicated situation with equally distributed pathogens. Results between 0.25 and 0.5 have to be classified by the user.

3.6 Knowledge acquisition and representation

This section describes how knowledge is stored and applied. The first part explains the knowledge database used by the application. The second part shows how the expert knowledge has been represented by decision trees and rules.

3.6.1 Knowledge database

The knowledge about antibiotic substances and their use is very complex as described in section 2.2.4. To compute recommendations for prescription all this knowledge had to be brought in a structured form. The development of the database was a lengthy process. At the beginning only names of antibiotic substances were included, later pathogen and diagnostic information were added. We defined the following fields of knowledge:

- antibiotics
- pathogens
- allergies and contraindications
- diagnoses
- resistance types
- application forms

Each field is represented by one or more tables in the knowledge database (see Appendix D). Hierarchies for pathogens and antibiotic substances needed to be modeled in these tables. If

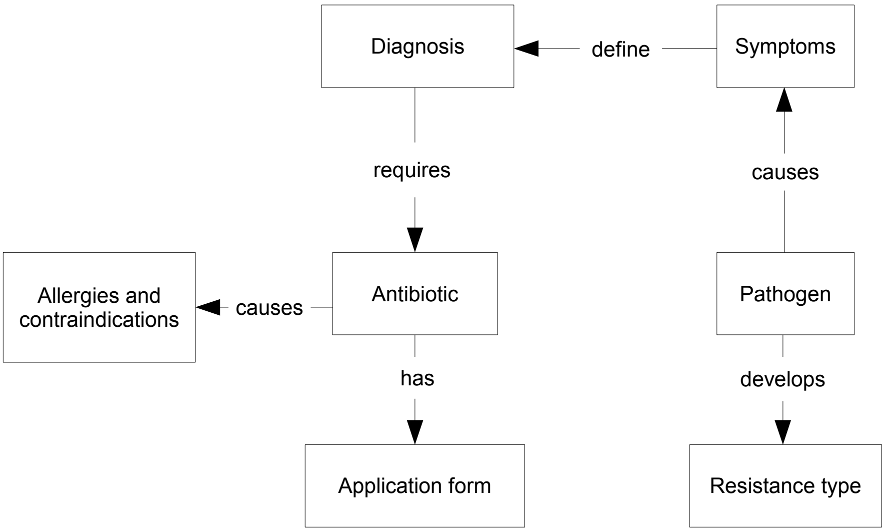


Figure 3.5: Model of relationships between the fields of knowledge covered by the application

clinical signs are given, the correct diagnosis is obtained according to the rules presented in the decision trees in Appendix A. In Figure 3.5 the relationships between these knowledge fields are shown.

Complementary to the entities shown in Figure 3.5 we needed to add tables with institution-specific names of pathogens and antibiotics. Antibiotic substances often have commercial names which are used in sensitivity patterns. Therefore these commercial names are translated to general regimen names and used for further processing. Pathogens raise a similar problem. The abbreviations of pathogen names are various and defined by the institution. To ensure comparability of these pathogens their names were translated to general pathogen names. Furthermore the following information was added to the database:

- categorization of pathogens by gram behavior
- resistance development—name of effect (ORSA, ESBL, VRE, etc.) and which substances must be tested
- pathogen and antibiotic connection—for which pathogen, which antibiotic treatment is necessary
- categorization of antibiotics—multiple names and antibiotic group name
- contraindication list and recommended other treatment

- commonly recommended therapies for UTI
- allergy effects

All this knowledge must be saved in an editable form. This knowledge is stored in tables, which are related to each other. Advantage of creating a database is easier understanding of both the structure and relationships and no need for a special syntax knowledge.

Pathogen knowledge

For this application extensive information about pathogens is necessary. We categorized pathogens into groups based on a pathogen classification defined in a textbook on microbiology [36, pages 94–97]. This knowledge contains the pathogen name, its gram behavior (see section 2.2.2), and the name of its resistance development. Antibiotics are used differently for every pathogen, as they may be reserve substances, antibiotics for extremely complicated cases, or substances which are ineffective. Therefore for each pathogen these antibiotics must be specified.

Antibiotic knowledge

Antibiotics often have various active agent names which function synonymously. The substances must therefore be classified into groups. In the generally accepted classification more than one hierarchy exists. Known antibiotic substances are assigned to these commonly known antibiotic groups. If a test of a pathogen shows a developing resistance towards a substance of a certain group, other antibiotics belonging to the same group will be ineffective as well. We also defined general, not institution-specific names for each substance. These general names are assigned one or more synonyms for one antibiotic regimen. We defined general names of antibiotic substances according to names found in a guide on antibiotics [37].

Each antibiotic may be contraindicated due to the patient's condition. Here special conditions like pregnancy, allergies to antibiotic substances, or other accompanied illness which may be dealt with differently, are included.

3.6.2 Expert knowledge

In the previous section we discussed which knowledge is necessary for the process of prescribing antibiotics. Even though we tried to record as much knowledge as possible in our database, not all knowledge may be saved in tabular form. Decisions based on symptoms or patient conditions are represented best in the form of rules. In the following sections we will describe how we acquired this knowledge, how rules emerged from the written knowledge, and the syntax of the resulting rules.

Finding rules

In medicine a lot of knowledge is presented in unstructured form, not suitable for being processed by computers. For this reason we tried to find structures in the knowledge found in literature concerning the prescription of therapy for UTIs. The following decision trees have been designed (see Appendix A):

- diagnosing a UTI based on patient's clinical signs
- obstruction decision tree
- decision on the existence of a complication
- duration of therapy
- application form

Based on these trees we created sets of rules. These rules are processed for each current patient and influence the selection of the antibiotic substances.

Rules

For this application we transformed expert knowledge into rules. These rules have input and output parameters. Based upon the input parameters, the output parameter is returned. The input parameters are "and" combined.

$$op(output)\ :-\ op_1(input_1)\ \wedge\ ...\ \wedge\ op_n(input_n) \tag{3.8}$$

The left hand side of the $:-$ assignment operator represents the output. The right hand side is the body of the rule. Each input parameter may have an operator function. This is a mathematical notation of our rule design:

$$output = operator(\bigcap_{i=1}^{N} operator[i](input[i])\ ) \tag{3.9}$$

Basically the output parameter receives the result of the "and" combined input parameters (see equation 3.9). Each input variable is first computed with the given operator. Results of these input variables combined with their operators are then AND related. This result is then computed with the operator of the output variable. This value represents the output of this rule. Output variables may become further input variables for other rules. We based our rule concept on the *closed world assumption* [6, page 286]. This is the assumption of the value of unknown variables, which according to this theory are false. Resulting from this assumption each variable, which is either unknown or not filled, is automatically assigned the value false. The conclusion of the previous assumptions is the necessity to write rules in the correct order. If variable A is used as input to a rule, then its value must be assessed before this rule comes to action.

Fuzzy operators

During knowledge acquisition the need for fuzzy operators emerged. Various parameters such as age or occurrence of the first pathogen cannot be defined in a crisp form. Fuzzy operators are defined by their name, border, type of the fuzzy function, and width of the slope. They also may be inverted. Three types of fuzzy function may be chosen: Gaussian curve, linear, and trapezoid function, which is symmetric to the defined border.

```
<fuzzy name="fuzzyFunctionName" [invert="true"]>
  <function var="functionType"/>
  <border var="functionBorder"/>
  <width var="functionWidth"/>
</fuzzy>
```

Fuzzy operators are used in the same way as other operators like e.g., "not".

```
<rule>
<in op="fuzzyFunctionName" var="input"/>
<out var="output"/>
</rule>
```

In this case the application receives the result of the fuzzy operator applied to the given value. The input variable must be a real number. The output variable then receives the membership value of the input variable compared to the border of the fuzzy function. The "invert" parameter of the fuzzy function states whether the function starts at zero or at one. If the "invert" parameter is defined as true then the function starts at one.

3.7 Generating recommendations

Recommendations may be presented in various ways. We decided to present a list of antibiotic regimen with a label indicating the application's recommendation. Based on the computed expected pathogen spectrum or on the patient's diagnosis we generate lists of antibiotics. All substances in the list of antibiotics undergo a labeling process. Each substance receives one label depending on patient's allergies, contraindications, found resistances, or indicated therapies:

- contraindicated – red label
- not indicated – orange label
- neutral – blue label
- indicated – green label
- unknown – white label

3.7.1 Labeling process

If the antibiotic list was built for a complicated case, it mostly contains antibiotic regimen according to given clinical signs and the resulting diagnosis. The initial status of these substances is "indicated"—green. Now the patient's special conditions and allergies are taken into account and antibiotics, which would worsen these conditions or endanger the patient are labeled accordingly. If a condition or allergy leads to a new label of an antibiotic substance, a message is generated containing the name of the condition or allergy. The labeled antibiotic list and for each antibiotic also the reason for its label are presented.

For the simple case the antibiotic list is generated based on recommended therapies for the leading pathogen of the computed expected pathogen spectrum. The further process is the same as for the complicated case, except here antibiotics start with the label "unknown". The reason for this initialization of substances is described in the section on the positive knowledge problem.

3.7.2 Positive knowledge problem

Knowledge about antibiotic regimen used for a certain diagnosis may be found in medical literature. This knowledge explicitly states which regimen are to be used. These statements we translate into "indicated" labels in the recommendation list, because explicitely these antibiotic substances are declared to be effective for just this diagnosis.

On the other hand, for the simple case we assess all antibiotic regimen for a certain pathogen, not for a diagnosis. While these therapies are known to be effective against this pathogen in medical literature, this knowledge does not state that this therapy will be appropriate for UTIs too. Therefore, antibiotics which were not excluded based on allergies and conditions are labeled unknown. To label these antibiotics as indicated is not possible, because knowledge which would indicate these therapies is missing.

3.8 Reporting development of resistance

Additional to the information about expected pathogen spectra and recommendation of antibiotics the information about a possible development of resistance is visualized by the application. We decided to tackle only ORSA, VRE, and ESBL resistance development. For each of these resistance types we defined in a table which antibiotic regimen are typically ineffective. The expected pathogen spectrum shows for each pathogen its tested antibiotic substances. The percentage of resistant and sensitive bacteria found is reported.

If the pathogen could possibly develop a resistance of a known resistance type its tested substances are checked. The next approach has three steps, beginning at the specific antibiotic substance and ending at the general antibiotic group.

The ineffective antibiotic regimen for the expected resistance type is selected and following conditions are checked:

1. if the found antibiotic has been tested: is the resistance value greater than 0?
2. else if any member of the found antibiotics subgroup has been tested: is the resistance value greater than 0?
3. else if any member of the found antibiotics group has been tested: is the resistance value greater than 0?

If any antibiotic is found to be resistant a warning is shown on the screen. The difficulty of this method lies in the necessity to translate antibiotic substances from general antibiotic names to institution-specific names, which complicates the process even more. The steps of the translation process are given below:

- find the general name for the institution-specific substance mapped explicitely
- if the first step failed, then find among the institution-specific ones all antibiotics belonging to the same subgroup as the general antibiotic.

4 Implementation

In this chapter the technical implementation of the application is discussed. The environment used for the development of this application is described and the single steps of the program are explained in detail.

4.1 System description

The application consists of two active parts: condensing and knowledge processing. These components process data and knowledge. Further the inactive parts, a knowledge database and a rules file in XML format, make up the knowledge base of the system.

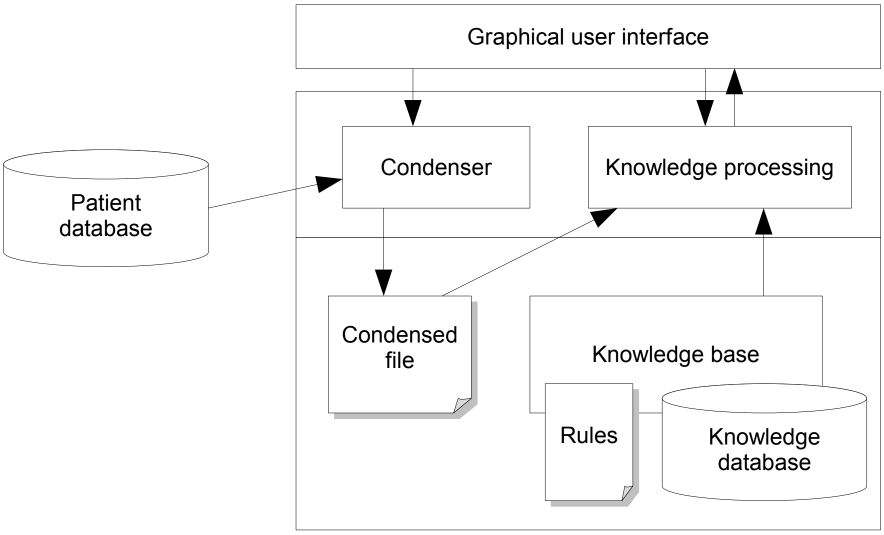


Figure 4.1: Components of the application

The condenser component is triggered by a menu in the user interface to process new data. This component processes data from the retrospective patient database to create a file of condensed patient records in XML format. The component also checks the validity of records. It expects a set of data in comma separated value (CSV) format and transforms it to an XML file of a given structure. The second and main part of the application is the knowledge processing part. After validating the input from the user interface an appropriate condensed case is extracted from the condensed file. Next the connection to the knowledge database is established and rules are evaluated. All results are then collected and shown at the user interface.

4.1.1 Development environment

For the development of this application we had to choose an appropriate system and programming language. To keep the application as independent of the operating system as possible, we have chosen the platform independent language, Java 1.6. The programming environment has been Eclipse 3.3.

4.1.2 Database system

First of all tables and their relations were designed in entity relationship diagrams. We chose to use the JDBC (Java Database Connectivity) standard as interface to the database. As database we used the open source project from Apache named Derby. For easier handling of the database we further used the program SQuirreL$^{\text{TM}}$, a program with a graphical user interface to view the database's structure. The easiest way to fill tables in this program is to import the respective data in CSV format. This process consists of the following steps:

- start Derby server
- start SQuirreL and open database
- import prepared CSV files

4.1.3 Rules and data file

To save condensed data and rules we decided to use the XML format. The reason for this choice was the availability of XML parsers for Java and the structurability of this format. For each of these XML files (condensed and rules file) we first had to determine the structure of the document. Based on this structure XML schema definition files for these documents were written. Since we fulfilled the prerequisites to use data binding to connect the XML contents with the application (see section 2.1.4), we decided to use JAXB (Java architecture for XML binding). From these structure files we generated Java objects. These we use for reading XML content and, in case of condensing data, also for writing to XML files.

XML syntax of rules

Expert rules are stored in an XML file. We tried to design the syntax of the rules to be self-explanatory. A rule in XML syntax is given as follows:

```
<rule>
  <in [op = "operator1"] var = "input1"/>
  ...
  <in [op = "operatorN"] var = "inputN"/>

    <out [op = "operator"] var = "output"/>
</rule>
```

A rule is defined by a rule beginning and end tag. Each input or output parameter may contain an operator, for example, "not", or a defined fuzzy operator. Allowed types for input and output parameters are Boolean logic values (true/false) and real numbers. Variables are defined as follows:

```
<var name="variableName" type=("boolean"|"real")/>
```

Each variable has a name which is referred to in rules. Supported types of variables are Boolean logic values and real numbers.

Java architecture for XML binding

We used JAXB to define the association between XML documents and Java objects. Java objects are created from data of an XML file in one go, and thus can be dealt with immediately. This process of transferring XML file information into Java objects is called marshalling [13, page 310]. The reverse process of writing Java object content into XML is called unmarshalling. Through these two processes the enduser receives objects without having to write a parser and a writer for the XML file. JAXB is a higher-level application platform interface (API) which reduces programming effort by providing functions for parsing and serializing of Java objects. By defining the structure of the XML file in the XML schema language, which defines the grammar of the XML file, a script builds a corresponding Java class structure. This way even after changing the XML structure the Java classes may be adapted easily. By generating these Java classes automatically getter and setter methods are created to obtain the content of the instances.

4.2 Program start

When starting the application automatic processes are started without user interaction. These actions control data, read system settings and files.

- At program start the date of the CSV file is checked. If the date of the CSV file holding the retrospective data does not match the date of the condensed XML file, the condensing process starts automatically. The CSV file may also be changed while the program is running and the condenser started manually from the program's user interface.
- By default the application language is set to the operating system language, but may be changed within a respective menu in the application. A change affords the application to be started anew. Currently, German and English are available.
- From the XML rules file and the condensed XML file respective Java objects are instantiated.

4.3 Condensing data

We received patient data from the Hygiene Department of the Vienna General Hospital stored in CSV format. CSV format is used for structured data such as tables. To extract data for the condensed file the files are parsed by the parser, the information is condensed and the XML file written. Further the XML content is unmarshalled (see section 2.1.4) into Java objects. If the user starts this process manually, all objects are generated anew.

4.3.1 Condensed file

First we defined the structure of this condensed file using the extended stylesheet language (XSL). It emerged from the given database of patients and from the needs. Based on three describing parameters—sex, specimen category, and ward category—condensed prototypes are generated. These describing parameters make up the prototype identification key. We define a prototype as a group of patient records which have the same three defining parameters. The number of condensed prototypes can be computed as follows:

$$number\,of\,sex\,values\ *\ number\,of\,specimen\,category\,values\ *\ number\,of\,ward\,category\,values \tag{4.1}$$

All cases are stored within one XML file. The structure of one prototype in the XML file is defined by the:

- number of records included
- identification key consisting of three describing parameters (sex, specimen category, and ward category)
- list of sensitivity patterns
- list of pathogens
- list of ages
- list of wards or departments
- list of specimen
- list of timestamps
- list of dates, when the specimen probes were taken

Most of these lists are self-explanatory. The timestamp list is used for internal help. For calculating resistance values, records of the last 6 months are used, for pathogen spectra, records of the last year are used. To speed up performance time when extracting data from the XML file, every record is labeled according to the date of origin of the specimen probe. Four kinds of labels exist: 6 months, 1 year, 3 years, and 6 years. In XML list elements are separated by whitespace, thus whitespace in list elements should be omitted. However, for this use and textual output whitespace is also necessary. To make sure words are separated correctly all whitespace have been replaced by the special character "@".

4.3.2 Performance

The duration of condensing depends on the size of the underlying database. The largest database processed was a database inheriting data from a period of seven years with a size of 24 MB. To condense this database it took the program one second and an XML file of 9 MB was produced.

4.4 Finding recommendations

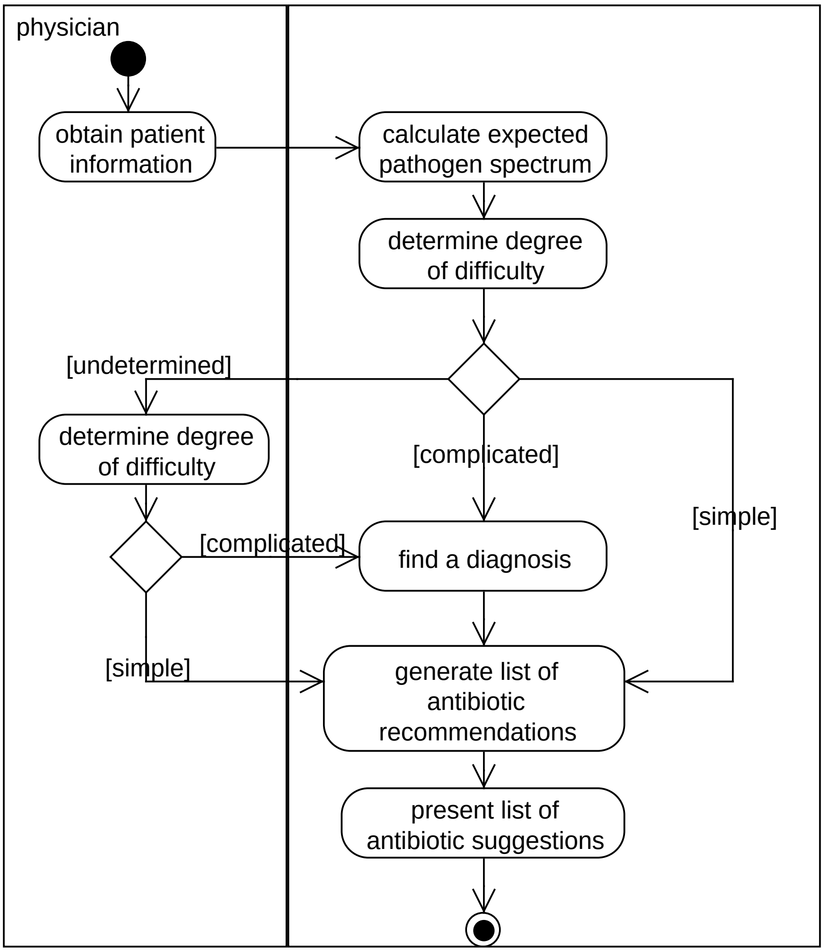

physician
obtain patient information
calculate expected pathogen spectrum
determine degree of difficulty
[undetermined]
determine degree of difficulty
[complicated]
[simple]
[complicated]
find a diagnosis
[simple]
generate list of antibiotic recommendations
present list of antibiotic suggestions

Figure 4.2: Workflow chart of the application

After typing in data the end of input must be confirmed by clicking the "OK" button. At this point no further data input for the current patient is possible. To unlock the input pane the button "new" must be clicked. "New" indicates that data of a new patient will be entered into the form and all tables as well as background objects are erased and generated for the new situation. These are the coarse steps of the program, which will be explained in detail:

- import proper condensed case
- filter records of imported condensed case
- summarize pathogens
- generate list of antibiotics
- update graphical user interface

4.4.1 Import condensed case

In view of this step, data from the condensed file have already been loaded at program start. For the current patient the identification key is determined. The identification key consists of values of sex, suspected site of infection, which corresponds to the specimen category, and ward or station, corresponding with the ward category. Based on this key an appropriate prototype is retrieved.

4.4.2 Filter records of imported condensed case

The imported condensed case contains many records. But not all of these records are of interest, thus two filter processes are introduced:

- filter records by timestamp
- filter by age (see section 3.4)

Filter by timestamp

The timestamp was intended for performance enhancement. This concept was introduced while working with a database holding data of the last eight years. In a healthcare facility historic data are important, however, with respect to mapping the current resistance situation eight year old data are entirely outdated. Together with a physician it has been decided to only use data of the past six months for the evaluation of the current resistance situation. To compute expected pathogen spectra, data of the last year are used. The respective timestamp is set for each record based on the difference between the actual date and the date when the probe was taken (e.g., if the difference is less than or equal to 6 months, the record receives the label "6M"). Four labels were introduced:

- 6M (six months),
- 1Y (one year),
- 3Y (three years),
- 6Y (six years),

although the last two labels are not used in the application yet. In summary, this filter removes all records from the condensed case, which do not belong to a given period of time.

Filter by age

While in section 3.4 it is specified how the age group is computed, here it is described how this procedure is implemented. At first, the upper and lower age limits are assessed. These represent the maximum width of the group, which has been defined.

Starting with the patient's age the number of cases in the respective age slot is assessed and compared with the minimum group size (see equation 3.5). If the number of cases in this age slot is less than the minimum group size, then the adjacent age slots are added. Now we assess the number of cases in these three age slots—patient's age minus one year, patient's age, and patient's age plus one year. Each time the number of cases is less than the minimum group size the next two age slots—one to the left and one to the right—are added and the number of records is assessed again. The process is stopped as soon as the number of in this way found records is equal or higher than the minimum group size, or the upper and lower age limits are reached.

4.4.3 Summaries

Summaries within this application are basically descriptive statistics for antibiotic regimen or pathogens. These statistics are then used to find antibiotic recommendations.

Pathogens

Following data is extracted for each pathogen from all records:

- name of pathogen
- occurrence
- averaged sensitivity pattern over all records containing this pathogen
- occurrence of resistant test results

Antibiotics

This view reverses the previously collected knowledge about pathogens. This is the view from the side of the substance showing pathogens being resistant to it and to what extent.

4.4.4 Generating antibiotic recommendations

To produce a list of antibiotic substances recommended for therapy, it first must be decided on the degree of difficulty of the situation. This decision is based on the distribution of pathogens in the calculated pathogen spectrum, which is done according to section 3.5. After obtaining values from the first rule (first pathogen's ratio is above 50%) and from the quotient rule (second pathogen's share is less than a third of the first pathogen's percentage) these two values are multiplied. Each case with a result less than or equal to the quarter mark (≤ 0.25) is defined to be a complicated case, and, when the result is higher than 0.50, it is defined to be a simple case. Results lying inbetween these two limits need to be interpreted by a physician. After having determined the respective degree of difficulty, the program generates a basic list of antibiotics as follows:

- simple case:
  For the simple case a therapy for the first, leading pathogen is proposed. From the knowledge database all antibiotics identified as treatment for this pathogen are added to the list.
- complicated case:
  In this case a therapy according to the entered symptoms or diagnosis is searched for. In both cases all antibiotic regimen suitable for treatment of the defined diagnosis are given.
  - if a diagnosis is entered, further given symptoms are ignored. Assuming, that the

physician knows the accurate diagnosis, the diagnosis is not compared with possibly further entered symptoms.

- if only symptoms are given, the diagnosis is determined by the program's rules.

This initial list contains antibiotic regimen indicated for the current patient. Further these substances are restricted due to allergies, contraindications, and resistances found in retrospective sensitivity patterns. Restricting in the context of the application means labeling the respective antibiotic (see section 3.7). In case of allergies, for example, the updated label is always "contraindicated", forbidding the prescription of this antibiotic. Other contraindications have an information about the degree of forbiddance. After the label has changed a message is generated with the name of the cause for the new label and extra information, if available in the database.

- if the patient is allergic to any antibiotic substances:
  For all known allergies all regimen are identified, which could cause an allergic reaction. With this list of "nono" substances we restrict the initial list. First, the list is reduced by these substance names, after that by the antibiotic subgroup, and at last by the antibiotic group too.

- if the patient has contraindications other than allergies:
  For all occurring contraindications all substances are listed which endanger the patient or could worsen his condition. By this list of substances the recommendation list is restricted.

- resistances are computed:
  Each substance in the regimen list is assigned its institution-specific name. With this name the sensitivity patterns are searched through. If a resistance value occurs for a pathogen, the pathogen's name and resistance value is displayed.

At the end of this process the recommendation list with updated labels is presented on the user interface.

4.4.5 Graphical user interface

The graphical user interface of the application contains the most important information on the first screen. In Figure 4.3 the schema of the user interface is shown. On the left-hand side—via the input form—information about the current patient is entered. It is divided into four sections: basic patient data, program information panel, additional patient data, and buttons. On the right-hand side the main output screen of the application is located. This part is multilayered and

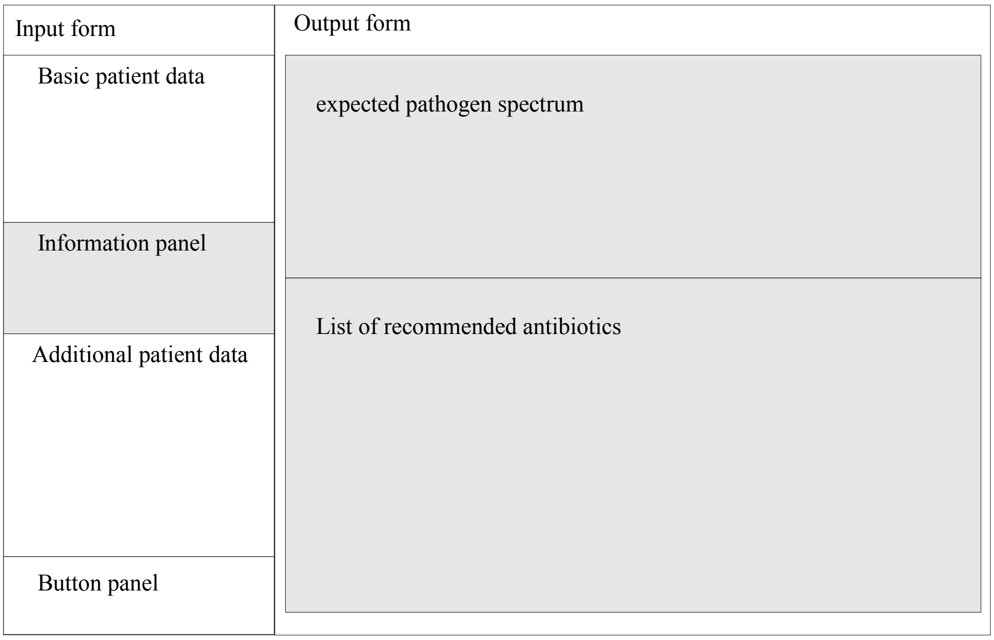


Figure 4.3: Schema of the graphical user interface

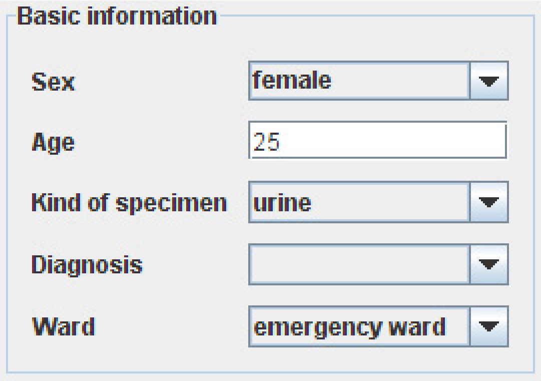
visualizes information generated by the application such as the expected pathogen spectrum, the list of recommended antibiotics, pathogen resistance information as well as an detailed explanation for the generated recommendations. All boxes of gray color are filled by the application.

Input form

The input form on the left hand side of the graphical user interface consists of four sections, as described above.

Basic patient data, see Figure 4.4, are the four variables defining a patient: sex, age, kind of specimen, or site of infection, and ward. Additionally the diagnosis, if known, may be specified here. The input of a diagnosis is optional for simple cases, obligatory for complicated ones. If the exact diagnosis for complicated cases is not known, then symptoms must be entered. Selectable values in the field diagnosis are only loaded after the kind of specimen is selected.

To start the calculation the button "OK" must be clicked, see Figure 4.5. This way the data is submitted and subsequent the whole input form is locked—no further data input is possible. For a new calculation, the button "New" has to be clicked. The input form is initialized for new input.



Basic information

Sex	female
Age	25
Kind of specimen	urine
Diagnosis	
Ward	emergency ward

Figure 4.4: Input of basic patient data

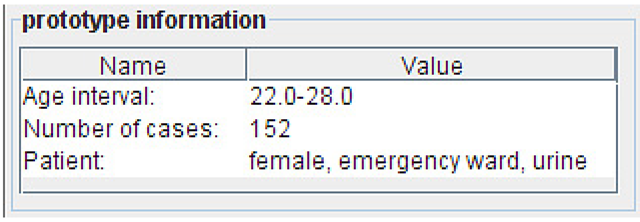
Figure 4.5: "OK" button to confirm calculation start and "New" button to begin the input of another patient

After the calculation of the expected pathogen spectrum the details of the group selection are displayed within the information panel (see Figure 4.6). These are made up of the age interval of the selected age group, the number of records contained in this group, and the identification of the respective prototype selected: sex, ward category, and kind of specimen.

In Figure 4.7 the input options for additional patient conditions, contraindications, allergies, and clinical signs are shown. The content of these panels depends on the sex of the patient, and the kind of specimen. Based on these values, for example, the check box pregnancy is enabled or disabled, or the UTI panel is replaced by a panel for clinical signs closely related to another kind of specimen.

Output screen

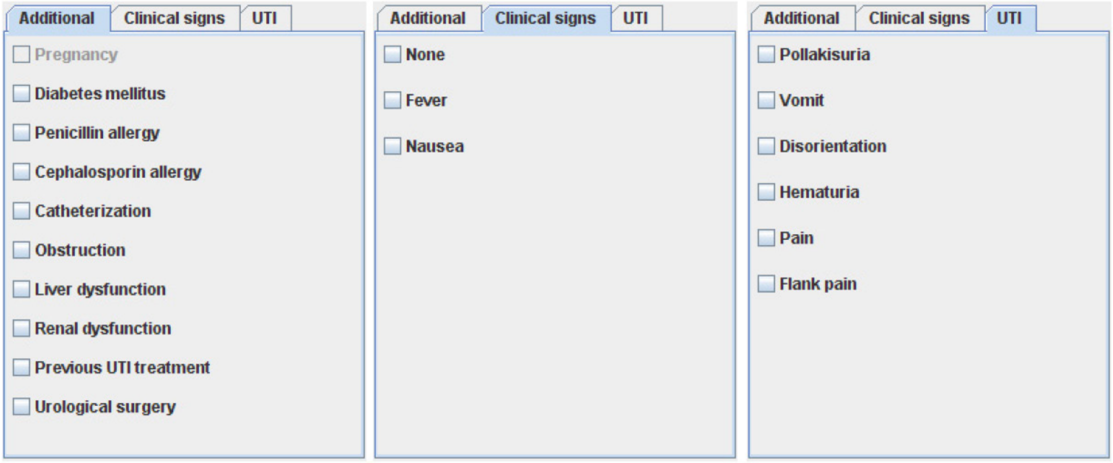
The output screen shows the main results of the application. A corresponding screenshot is shown in Figure 4.8. To be able to present as much information as possible but to give the most important information first, we designed this form multilayered. On the first tab the expected pathogen spectrum as well as the list of antibiotic recommendations are presented.



prototype information

Name	Value
Age interval:	22.0-28.0
Number of cases:	152
Patient:	female, emergency ward, urine

Figure 4.6: Prototype information with group selection details



□ Pregnancy
□ Diabetes mellitus
□ Penicillin allergy
□ Cephalosporin allergy
□ Catheterization
□ Obstruction
□ Liver dysfunction
□ Renal dysfunction
□ Previous UTI treatment
□ Urological surgery

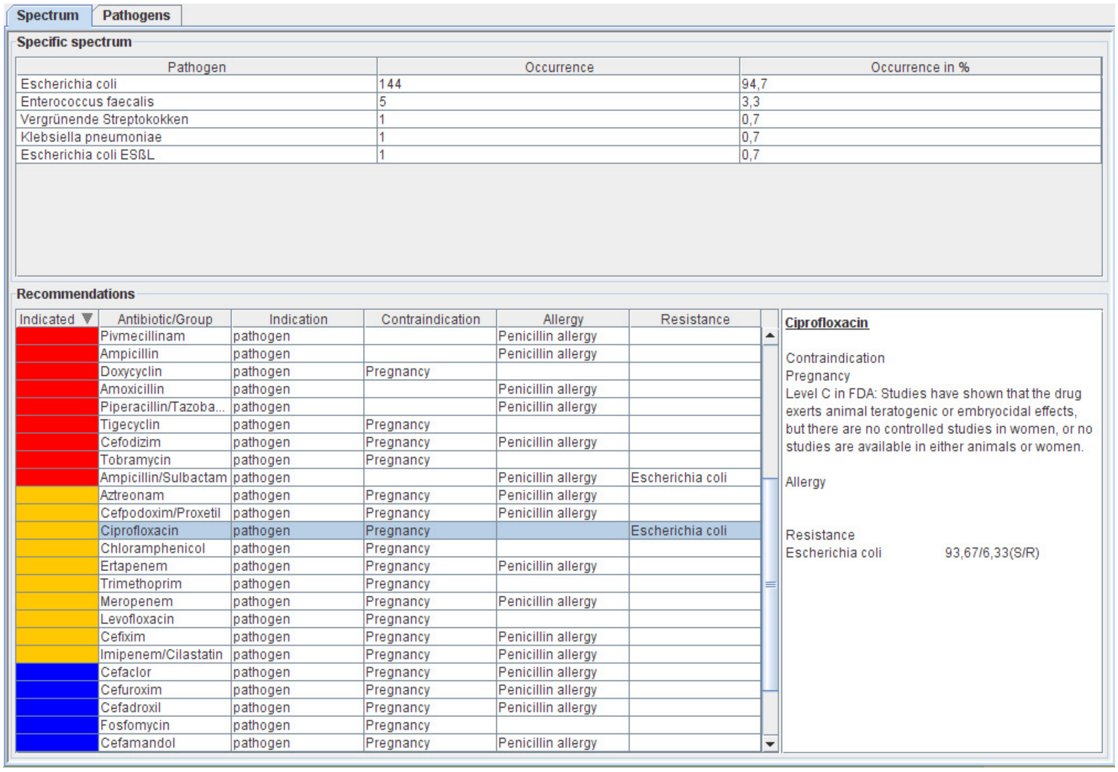
□ None
□ Fever
□ Nausea

□ Pollakisuria
□ Vomit
□ Disorientation
□ Hematuria
□ Pain
□ Flank pain

Figure 4.7: Input options for additional patient information, clinical signs, and specific UTI symptoms

The table with the expected pathogen spectrum contains three columns: pathogen name, occurrence, and occurrence in percent. Values of occurrence state how many pathogens of the respective kind have been found in the group of similar patients. Occurrence in percent gives the percentage share of the respective pathogen. The list of antibiotic recommendations is based on the statistical view described in section 3.4.2. This table contains six columns: indicated status, name of antibiotic or group of antibiotics, indication, contraindication, allergy, and resistance. The indication status of each antibiotic is calculated from patient's conditions, contraindications, and allergies (see also section 3.7). The column indication states upon which basis this antibiotic substance was added to the list of recommendations. The column contraindication contains for each regimen all contraindications, which cause a different indication status, e.g., pregnancy.

Each row in the list of recommendations can be selected per mouse click. More detailed information about the given facts is then presented next to the list of recommendations in a panel on the right-hand side. For example, a detailed explanation why the antibiotic regimen should not be prescribed is given.



Specific spectrum

Pathogen	Occurrence	Occurrence in %
Escherichia coli	144	94,7
Enterococcus faecalis	5	3,3
Vergrünende Streptokokken	1	0,7
Klebsiella pneumoniae	1	0,7
Escherichia coli ESßL	1	0,7

Recommendations

Indicated ▼	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
🟥	Pivmecillinam	pathogen		Penicillin allergy	
🟥	Ampicillin	pathogen		Penicillin allergy	
🟥	Doxycyclin	pathogen	Pregnancy		
🟥	Amoxicillin	pathogen		Penicillin allergy	
🟥	Piperacillin/Tazoba...	pathogen		Penicillin allergy	
🟥	Tigecyclin	pathogen	Pregnancy		
🟥	Cefodizim	pathogen	Pregnancy	Penicillin allergy	
🟥	Tobramycin	pathogen	Pregnancy		
🟥	Ampicillin/Sulbactam	pathogen		Penicillin allergy	Escherichia coli
🟨	Aztreonam	pathogen	Pregnancy	Penicillin allergy	
🟨	Cefpodoxim/Proxetil	pathogen	Pregnancy	Penicillin allergy	
🟨	Ciprofloxacin	pathogen	Pregnancy		Escherichia coli
🟨	Chloramphenicol	pathogen	Pregnancy		
🟨	Ertapenem	pathogen	Pregnancy	Penicillin allergy	
🟨	Trimethoprim	pathogen	Pregnancy		
🟨	Meropenem	pathogen	Pregnancy	Penicillin allergy	
🟨	Levofloxacin	pathogen	Pregnancy		
🟨	Cefixim	pathogen	Pregnancy	Penicillin allergy	
🟨	Imipenem/Cilastatin	pathogen	Pregnancy	Penicillin allergy	
🟦	Cefaclor	pathogen	Pregnancy	Penicillin allergy	
🟦	Cefuroxim	pathogen	Pregnancy	Penicillin allergy	
🟦	Cefadroxil	pathogen	Pregnancy	Penicillin allergy	
🟦	Fosfomycin	pathogen	Pregnancy		
🟦	Cefamandol	pathogen	Pregnancy	Penicillin allergy	

Ciprofloxacin

Contraindication
Pregnancy
Level C in FDA: Studies have shown that the drug exerts animal teratogenic or embryocidal effects, but there are no controlled studies in women, or no studies are available in either animals or women.

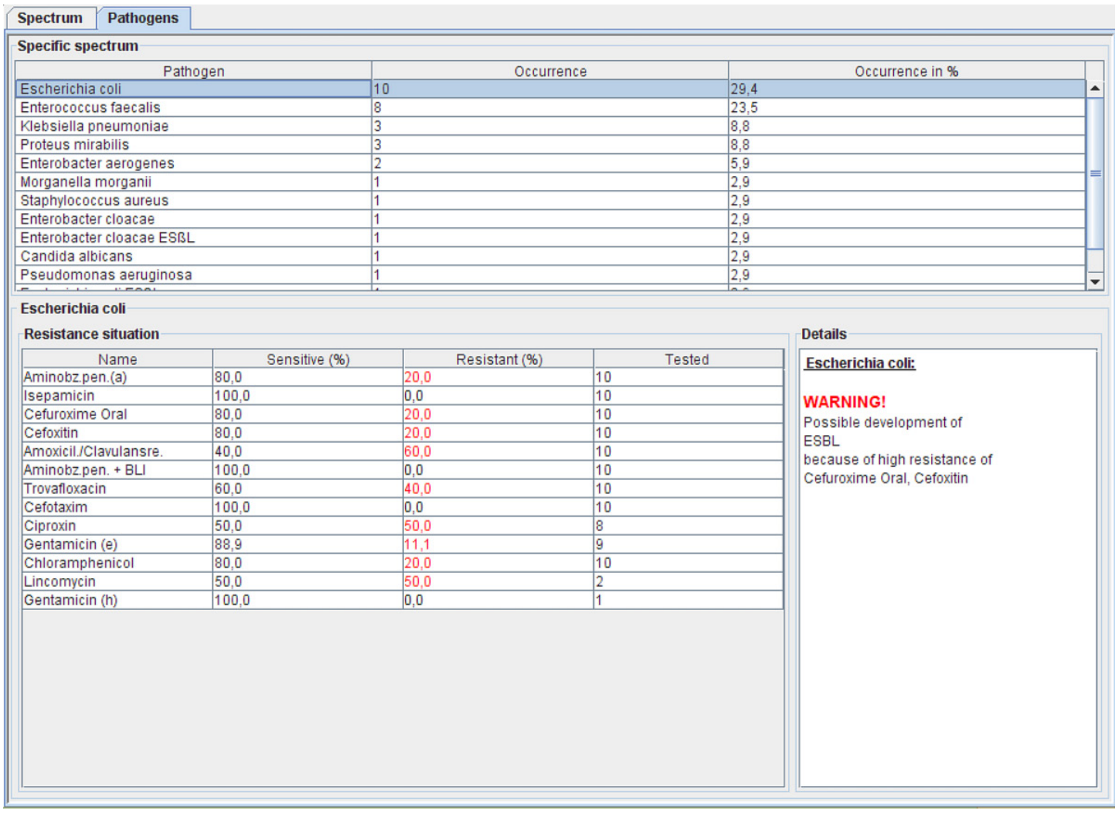
Allergy

Resistance
Escherichia coli 93,67/6,33(S/R)

Figure 4.8: Screenshot of the output screen with detailed spectrum information

On the second tab, named "Pathogens", the expected pathogen spectrum is shown once again. Here each row can be selected per mouse click again, the average sensitivity pattern appears in the lower part of the screen (see section 3.4.2).

The table "resistance situation" contains four columns: name of antibiotic substance, occurrence of sensitive pathogens, occurrence of resistant pathogens, and the number of records with this pathogen that have been tested with this substance. For pathogens where resistance development is possible, the respective antibiotic groups are checked for resistances, see section 3.4.2. If a resistance occurs, a warning message appears next to the resistance situation table on the right side. This message warns of a possible development of a certain resistance due to a present antibiotic resistance.



Specific spectrum

Pathogen	Occurrence	Occurrence in %
Escherichia coli	10	29,4
Enterococcus faecalis	8	23,5
Klebsiella pneumoniae	3	8,8
Proteus mirabilis	3	8,8
Enterobacter aerogenes	2	5,9
Morganella morganii	1	2,9
Staphylococcus aureus	1	2,9
Enterobacter cloacae	1	2,9
Enterobacter cloacae ESßL	1	2,9
Candida albicans	1	2,9
Pseudomonas aeruginosa	1	2,9

Escherichia coli

Resistance situation

Name	Sensitive (%)	Resistant (%)	Tested
Aminobz.pen.(a)	80,0	20,0	10
Isepamicin	100,0	0,0	10
Cefuroxime Oral	80,0	20,0	10
Cefoxitin	80,0	20,0	10
Amoxicil./Clavulansre.	40,0	60,0	10
Aminobz.pen. + BLI	100,0	0,0	10
Trovafloxacin	60,0	40,0	10
Cefotaxim	100,0	0,0	10
Ciproxin	50,0	50,0	8
Gentamicin (e)	88,9	11,1	9
Chloramphenicol	80,0	20,0	10
Lincomycin	50,0	50,0	2
Gentamicin (h)	100,0	0,0	1

Details

Escherichia coli:

WARNING!
Possible development of
ESBL
because of high resistance of
Cefuroxime Oral, Cefoxitin

Figure 4.9: Screenshot of the output screen with detailed pathogen information

5 Results and discussion

In this chapter the application is evaluated by comparing the automatically generated lists of antibiotic substances with recommendations given in medical literature. After the evaluation a short statement of what has been done and proven follows. Finally some ideas for further development of this application are given.

5.1 Evaluation

The goal of this work was to build a decision support system for the management of antibiotics therapy. The resulting application is not yet to be used routinely in a clinical setting as it is only a prototype. Its evaluation in daily routine is thus impossible, however to be able to decide on the success of this application an evaluation is needed. Therefore five test cases were designed. In this chapter each of these cases is described and the computed recommendations are compared with those found in medical literature [17, pages 543–556], [19]. Each test case represents a typical situation for UTIs. For each test case the following questions, which made up our evaluation, were answered:

- Does the pathogen spectrum correspond with the expected spectrum in literature?
- Does the computed list of antibiotics contain antibiotics recommended by medical literature?
- Are prohibited antibiotic regimen marked so?
- Are the reasons for contraindicating substances correct?

Each of the following subchapters describing a test case has the same outline:

- short description of the test case
- expected results in literature

- patient information available to the application
- computed recommendations
- comparison

5.1.1 Test case 1: young, female patient with acute, uncomplicated UTI

This test case shows an otherwise healthy, female patient having an acute, uncomplicated UTI.

Expected results in literature

In medical literature the expected pathogen spectrum for this and similar cases is defined as follows [19, page 79]:

- *Escherichia coli*
- *Proteus mirabilis*
- *Klebsiella pneumoniae*
- *Enterobacteria*
- Staphylococcus spp

Following regimen are recommended as initial therapy for this case:

- parenteral: Cephalosporin Group 2 or 3a, Fluoroquinolons (Ciprofloxacin, Levofloxacin), Aminopenicillin/Beta lactamase inhibitors (BLI), Aminoglycosides
- oral: Fluoroquinolons, oral Cephalosporin Group 2 or 3, Aminopenicillin, BLI

Uncomplicated acute UTI can be treated with a single-dose therapy of the following regimen [17, page 548]:

- Levofloxacin
- Ciprofloxacin
- Oral cephalosporins

- Ceftriaxon
- Gentamicin

Input information

The following information about the patient was fed into the application:

- age: 25
- sex: female
- station: emergency station
- no complications
- no allergies
- suspected site of infection: urine
- suspected diagnosis: not specified

Computed recommendations

The application computed the following pathogen spectrum for this case:

Table 5.1: Computed pathogen spectrum

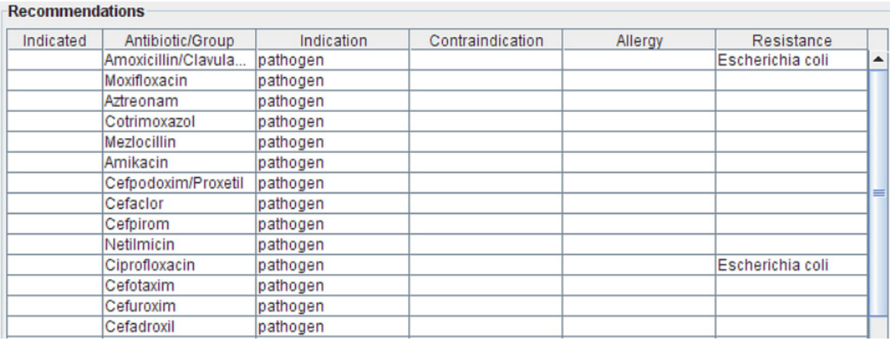
pathogen	**count**	**occurrence (%)**
Escherichia coli	144	94.7%
Enterococcus faecalis	5	3.3%
Hemolytic streptococci	1	0.7%
Klebsiella pneumoniae	1	0.7%
⋮	⋮	⋮

For the leading pathogen *Escherichia coli* three antibiotic regimen were found to be less effective, see Table 5.2.

Table 5.2: Antibiotic substances and their resistances

antibiotic substance	sensitive (%)	resistant (%)
Amoxicillin with clavulanate acid	68.1%	31.9%
Ciprofloxacin	93.7%	6.3%
Ampicillin Sulbactam	68.1%	31.9%

The following antibiotic regimen found for therapy of the leading pathogen, *Escherichia coli*, were listed in the recommendations list.



Recommendations

Indicated	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
	Amoxicillin/Clavula...	pathogen			Escherichia coli
	Moxifloxacin	pathogen			
	Aztreonam	pathogen			
	Cotrimoxazol	pathogen			
	Mezlocillin	pathogen			
	Amikacin	pathogen			
	Cefpodoxim/Proxetil	pathogen			
	Cefaclor	pathogen			
	Cefpirom	pathogen			
	Netilmicin	pathogen			
	Ciprofloxacin	pathogen			Escherichia coli
	Cefotaxim	pathogen			
	Cefuroxim	pathogen			
	Cefadroxil	pathogen			

Figure 5.1: Recommended antibiotic regimen list for a female, 25-year-old patient with UTI

Evaluation

Because the computed pathogen spectrum satisfies the defined rules for the respective case classification, this case was correctly classified as "simple". Therefore the resulting antibiotic regimen list is based on the therapy of the leading pathogen—*Escherichia coli*. The computed pathogen spectrum from the retrospective patient database shows much similarity with the suspected spectrum published in medical literature. One pathogen was not found at all in this group of patients. This might on one hand imply that the group was not representative, which is because of the availability of 152 records in the retrospective database less the case. On the other hand it might indicate a change in the suspected pathogen spectrum and the etiology of UTI in young, female patients. All antibiotic substances recommended by medical literature are included in the generated list. Because the only criterium for selection of antibiotics for this list was the leading pathogen, no ranking of the antibiotic regimen, as shown in Figure 5.1, is given. All of these antibiotic substances are recommended for therapy of the leading pathogen. Since this test case does neither have contraindications nor allergies no antibiotic regimen are marked as contraindicated.

5.1.2 Test case 2: young, female patient with acute UTI and complications

This test case is similar to the previous one, however deserves more attention, since it is a complicated case. Another common situation in UTI treatment is shown: a young, female, pregnant patient having an acute UTI. The suspected diagnosis is pyelonephritis.

Expected results in literature

For this test case, especially because of pregnancy, following antibiotic regimen are forbidden [17, page 550]:

- Aminoglycosides
- Doxycyclin
- Cotrimoxazol
- Nitrofurantoin
- Fluoroquinolons

Recommended antibiotic substances for this test case are [17, page 550]:

- Amoxicillin without Clavulanate acid
- Oral cephalosporins
- parenteral: Beta lactamase antibiotics (Cefuroxim, Ceftriaxon, Mezlocillin)

Input information

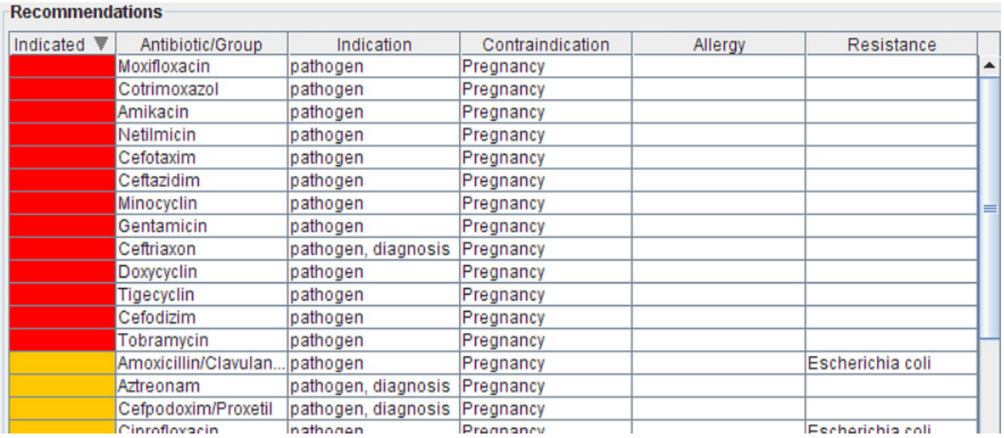
The following information about the patient was fed into the application:

- age: 25
- sex: female
- station: emergency station
- complications: pregnancy
- suspected site of infection: urine

- suspected diagnosis: pyelonephritis
- no allergies

Computed recommendations

Since the parameters necessary for computing the expected pathogen spectrum (age, sex, suspected site of infection, and ward) did not change the computed one is the same as for test case number one, however the recommended antibiotic therapy is totally different from the former one. Figure 5.2, shows the list of contraindicated antibiotic substances for this case marked in red. These substances are contraindicated based on the Federal Drug Association (FDA) classification, see Appendix B.



Recommendations

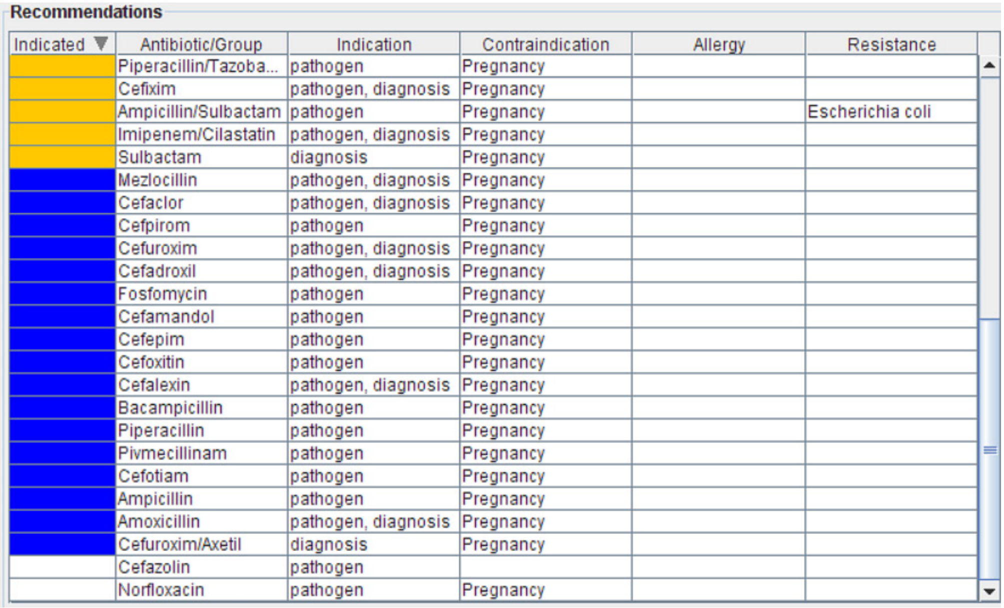
Indicated ▼	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
🟥	Moxifloxacin	pathogen	Pregnancy		
🟥	Cotrimoxazol	pathogen	Pregnancy		
🟥	Amikacin	pathogen	Pregnancy		
🟥	Netilmicin	pathogen	Pregnancy		
🟥	Cefotaxim	pathogen	Pregnancy		
🟥	Ceftazidim	pathogen	Pregnancy		
🟥	Minocyclin	pathogen	Pregnancy		
🟥	Gentamicin	pathogen	Pregnancy		
🟥	Ceftriaxon	pathogen, diagnosis	Pregnancy		
🟥	Doxycyclin	pathogen	Pregnancy		
🟥	Tigecyclin	pathogen	Pregnancy		
🟥	Cefodizim	pathogen	Pregnancy		
🟥	Tobramycin	pathogen	Pregnancy		
🟨	Amoxicillin/Clavulan...	pathogen	Pregnancy		Escherichia coli
🟨	Aztreonam	pathogen, diagnosis	Pregnancy		
🟨	Cefpodoxim/Proxetil	pathogen, diagnosis	Pregnancy		
🟨	Ciprofloxacin	pathogen	Pregnancy		Escherichia coli

Figure 5.2: Screenshot of the list of contraindicated substances

The remaining list of recommendations however does not contain any indicated antibiotic substances. Almost all antibiotic substances have an FDA classification of B to X (see Appendix B) and therefore are not allowed to be indicated. Antibiotic substances, which were selected as therapy of the leading pathogen, must not be shown as indicated, because this pathogen is a too general criterium. This problem is explained in section 3.7.2.

Evaluation

Based on the pathogen spectrum this test case is classified as "simple". Because of the input of a diagnosis also recommendations for the therapy of pyelonephritis are added to the list of antibiotic substances. This list now covers the therapy of both pyelonephritis and the leading



Recommendations

Indicated	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
🟨	Piperacillin/Tazoba...	pathogen	Pregnancy		
🟨	Cefixim	pathogen, diagnosis	Pregnancy		
🟨	Ampicillin/Sulbactam	pathogen	Pregnancy		Escherichia coli
🟨	Imipenem/Cilastatin	pathogen, diagnosis	Pregnancy		
🟨	Sulbactam	diagnosis	Pregnancy		
🟦	Mezlocillin	pathogen, diagnosis	Pregnancy		
🟦	Cefaclor	pathogen, diagnosis	Pregnancy		
🟦	Cefpirom	pathogen	Pregnancy		
🟦	Cefuroxim	pathogen, diagnosis	Pregnancy		
🟦	Cefadroxil	pathogen, diagnosis	Pregnancy		
🟦	Fosfomycin	pathogen	Pregnancy		
🟦	Cefamandol	pathogen	Pregnancy		
🟦	Cefepim	pathogen	Pregnancy		
🟦	Cefoxitin	pathogen	Pregnancy		
🟦	Cefalexin	pathogen, diagnosis	Pregnancy		
🟦	Bacampicillin	pathogen	Pregnancy		
🟦	Piperacillin	pathogen	Pregnancy		
🟦	Pivmecillinam	pathogen	Pregnancy		
🟦	Cefotiam	pathogen	Pregnancy		
🟦	Ampicillin	pathogen	Pregnancy		
🟦	Amoxicillin	pathogen, diagnosis	Pregnancy		
🟦	Cefuroxim/Axetil	diagnosis	Pregnancy		
	Cefazolin	pathogen			
	Norfloxacin	pathogen	Pregnancy		

Figure 5.3: Screenshot of the list of recommended substances

pathogen. Recommended antibiotic treatment by literature is mostly in the B category of the FDA. Only Ceftriaxon, recommended by medical literature, is prohibited since its FDA classification is X. Regimen prohibited by literature because of pregnancy are marked either prohibited or contraindicated, based on the FDA classification.

5.1.3 Test case 3: male patient with acute UTI

This test case shows a male patient having an acute UTI.

Expected results in literature

This test case represents a complicated UTI because of the complication factor of male sex [19, page 79]. The expected pathogen spectrum is supposed to include following bacteria [17, page 548]:

- *Escherichia coli*
- *Klebsiella species*
- *Enterobacter cloacae*

- *Enterobacter species*
- *Pseudomonas aeruginosa*
- Enterococcus spp
- Staphylococcus spp

A therapy duration of more than 20 days is recommended, indicating following substances:

- Fluoroquinolons
- Cotrimoxazol
- Amoxicillin
- Oral cephalosporins

Input information

The following information about the patient was fed into the application:

- age: 33
- sex: male
- station: emergency station
- complications: none specified
- suspected diagnosis: none specified
- suspected site of infection: urine
- no allergies

Computed recommendations

The expected pathogen spectrum based on retrospective data is shown in Table 5.3.

The calculated recommendation includes again antibiotic substances for the therapy of the leading pathogen—*Escherichia coli*. However, resistances were found in key therapy regimen: Ciprofloxacin (Fluorquinolone) and Amoxicillin.

Table 5.3: Computed pathogen spectrum

pathogen	**count**	**occurrence (%)**
Escherichia coli	31	81.6%
Klebsiella pneumoniae	2	5.3%
Enterococcus faecalis	2	5.3%
Proteus mirabilis	1	2.6%
Hemolytic streptococcus	1	2.6%
Citrobacter koseri	1	2.6%

Table 5.4: Antibiotic substances and their resistances

antibiotic substance	**sensitive (%)**	**resistant (%)**
Amoxicillin with clavulanate acid	53.3%	46.7%
Ciprofloxacin	95.0%	5.0%
Ampicillin Sulbactam	53.3%	46.7%

Evaluation

The expected pathogen spectra match to a large extent. There may be various reasons why the match is not perfect. As discussed in the evaluation of the first test case, this may show a change in the suspected pathogen spectrum for this patient group or may be due to outliers. The computed list of recommendations contains all recommended antibiotic regimen. While medical literature recommends these antibiotic substances, our application informs about resistances which occurred in some of these regimen, see Table 5.4.

5.1.4 Test case 4: 75-year-old, male patient with acute pyelonephritis

The group of elder male patients with diabetes mellitus is a significant one. According to [19, page 80], this group must receive a special antibiotic treatment.

Expected results

The suspected pathogen spectrum for this test case is shown below [19, page 80]:

- *Escherichia coli*
- Proteus spp

- *Klebsiella pneumoniae*
- *Staphylococcus aureus*
- Enterococcus spp

Therapy recommended by literature for complicated UTI includes:

- Trimethoprim/Sulfamethoxazol
- Trimpethoprim/Sulfamerazin

According to [19, page 80] fluoroquinolons are suitable for therapy of UTI in diabetes mellitus patients. However, another source [38] found a higher risk of hyperglycemia for diabetes mellitus patients being treated with fluoroquinolons. For this argumentation we chose not to indicate fluoroquinolons as well as Aminopenicillines and Nitrofurantoins for patients with diabetes mellitus.

Recommended therapy for acute pyelonephritis includes the following substances [17]:

- Cotrimoxazol
- Cephalosporins (Cefixim, Cefpodoxim)
- Ciprofloxacin
- there are resistancies to Amoxicillin
- complicated UTI requires parenteral therapy with: Cephalosporins, Piperacillin/Tazobactam, Fluoroquinolons

Input information

The following information about the patient was fed into the application:

- age: 75
- sex: male
- complications: diabetes mellitus
- station: emergency station
- no allergies

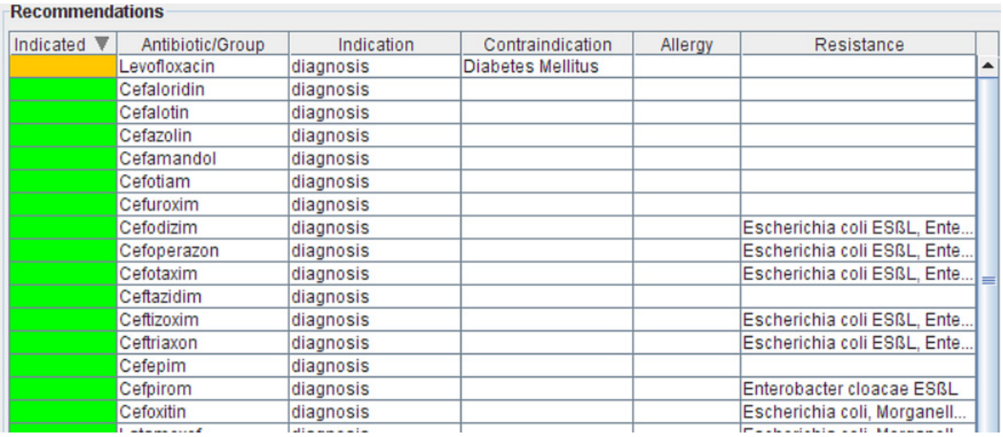
- suspected site of infection: urine
- suspected diagnosis: acute pyelonephritis

Computed recommendations

The computed expected pathogen spectrum is shown in Table 5.5.

Table 5.5: Computed pathogen spectrum

pathogen	count	occurrence (%)
Escherichia coli	31	81.6%
Enterococcus faecalis	2	5.3%
Proteus mirabilis	2	5.3%
Pseudomonas aeruginosa	1	2.6%
Enterobacter aerogenes	1	2.6%
Staphylococcus aureus	1	2.6%



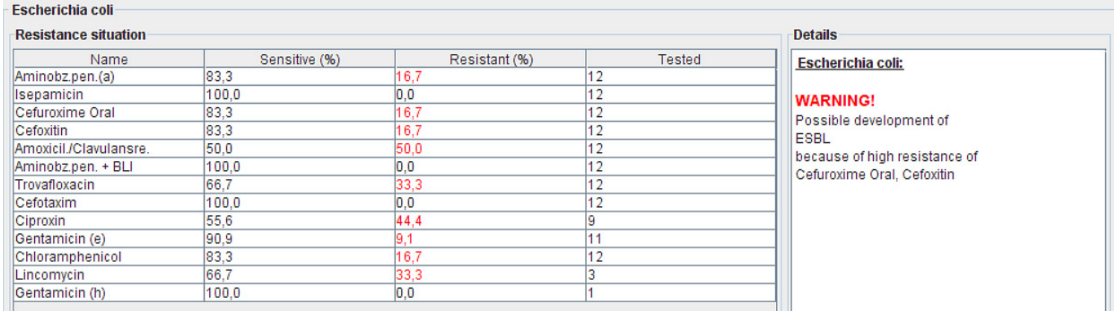
Recommendations

Indicated ▼	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
🟧	Levofloxacin	diagnosis	Diabetes Mellitus		
🟩	Cefaloridin	diagnosis			
🟩	Cefalotin	diagnosis			
🟩	Cefazolin	diagnosis			
🟩	Cefamandol	diagnosis			
🟩	Cefotiam	diagnosis			
🟩	Cefuroxim	diagnosis			
🟩	Cefodizim	diagnosis			Escherichia coli ESßL, Ente...
🟩	Cefoperazon	diagnosis			Escherichia coli ESßL, Ente...
🟩	Cefotaxim	diagnosis			Escherichia coli ESßL, Ente...
🟩	Ceftazidim	diagnosis			
🟩	Ceftizoxim	diagnosis			Escherichia coli ESßL, Ente...
🟩	Ceftriaxon	diagnosis			Escherichia coli ESßL, Ente...
🟩	Cefepim	diagnosis			
🟩	Cefpirom	diagnosis			Enterobacter cloacae ESßL
🟩	Cefoxitin	diagnosis			Escherichia coli, Morganell...

Figure 5.4: Screenshot of the list of recommended substances

Evaluation

The recommendation list was based on the entered diagnosis, since this test case is classified as "complicated". Therefore this recommendation represents the therapy for the diagnosis "acute pyelonephritis". This test case shows a complicated pyelonephritis. All substances recommended by medical literature were contained in our list. Antibiotic regimen contraindicated for diabetes mellitus patients were marked as contraindicated. In this test case multiple resistance developments occur. For *Escherichia coli* a possible development of ESBL was detected. Based on this information the physician may decide which regimen to prescribe.



Escherichia coli

Resistance situation

Name	Sensitive (%)	Resistant (%)	Tested
Aminobz.pen.(a)	83,3	16,7	12
Isepamicin	100,0	0,0	12
Cefuroxime Oral	83,3	16,7	12
Cefoxitin	83,3	16,7	12
Amoxicil./Clavulansre.	50,0	50,0	12
Aminobz.pen. + BLI	100,0	0,0	12
Trovafloxacin	66,7	33,3	12
Cefotaxim	100,0	0,0	12
Ciproxin	55,6	44,4	9
Gentamicin (e)	90,9	9,1	11
Chloramphenicol	83,3	16,7	12
Lincomycin	66,7	33,3	3
Gentamicin (h)	100,0	0,0	1

Details

Escherichia coli:

WARNING!
Possible development of
ESBL
because of high resistance of
Cefuroxime Oral, Cefoxitin

Figure 5.5: Resistance development of *Escherichia coli*

5.1.5 Test case 5: female patient with symptoms of cystitis

The underlying disease of this test case is only described by symptoms and clinical signs. Thus the application must first determine the correct diagnosis and then compute a therapy.

Expected results in literature

In this test case the patient suffers from acute cystitis. The diagnosis must be assessed based on clinical signs of the patient. The typical pathogen spectrum is the same as for test case number one. Following antibiotic substances are recommended for therapy:

- Ciprofloxacin
- Levofloxacin
- Cotrimoxazol
- Oral cephalosporins
- Amoxicillin
- Fosfomycin
- Ceftriaxon
- Aminoglycoside

Input information

The following information about the patient was fed into the application: Our test case is defined as:

- age: 55
- sex: female
- station: emergency station
- no complications
- no allergies
- suspected site of infection: urine
- symptoms: no fever, no flank pain, blood in urine, pain

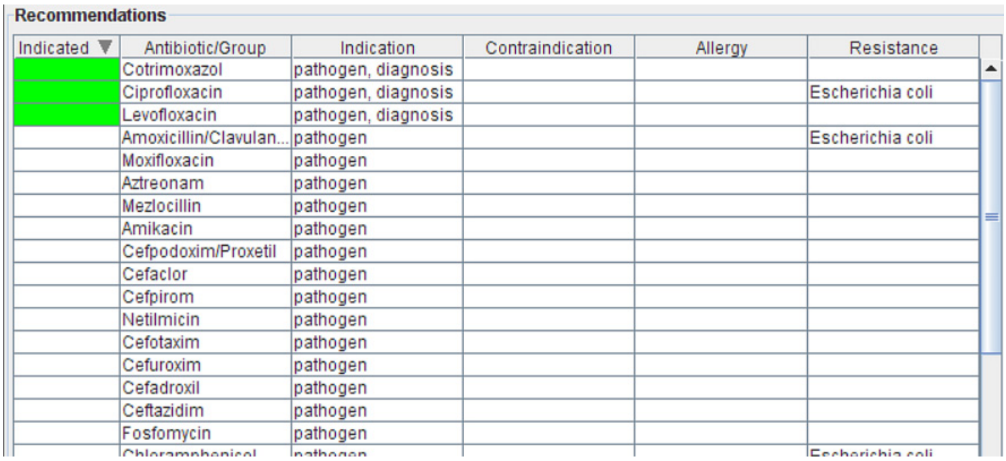
Computed recommendations

The application correctly defined the diagnosis to be acute cystitis. An excerpt of the expected pathogen spectrum in this case is shown below:

Table 5.6: Computed pathogen spectrum

pathogen	**count**	**occurrence (%)**
Escherichia coli	124	84.4%
Klebsiella pneumoniae	7	4.8%
Proteus mirabilis	3	2.0%
Enterococcus faecalis	3	2.0%
Escherichia coli ESBL	3	2.0%
Citrobacter koseri	2	1.4%
⋮	⋮	⋮

The antibiotic recommendation list contains three substances indicated for cystitis and others for therapy of the leading pathogen.



Recommendations

Indicated ▼	Antibiotic/Group	Indication	Contraindication	Allergy	Resistance
🟩	Cotrimoxazol	pathogen, diagnosis			
🟩	Ciprofloxacin	pathogen, diagnosis			Escherichia coli
🟩	Levofloxacin	pathogen, diagnosis			
	Amoxicillin/Clavulan...	pathogen			Escherichia coli
	Moxifloxacin	pathogen			
	Aztreonam	pathogen			
	Mezlocillin	pathogen			
	Amikacin	pathogen			
	Cefpodoxim/Proxetil	pathogen			
	Cefaclor	pathogen			
	Cefpirom	pathogen			
	Netilmicin	pathogen			
	Cefotaxim	pathogen			
	Cefuroxim	pathogen			
	Cefadroxil	pathogen			
	Ceftazidim	pathogen			
	Fosfomycin	pathogen			
	Chloramphenicol	pathogen			Escherichia coli

Figure 5.6: Screenshot of the list of substances with recommended and unknown labels

Evaluation

The computed pathogen spectrum does not exactly match the pathogens given in medical literature, as in the previous test cases. The computed recommended therapy covers first three regimen recommended by medical literature. These first three substances are marked in the knowledge base as treatment for acute cystitis. Other antibiotics recommended by medical literature are also contained in the list, however as therapy of the leading pathogen.

5.2 Conclusion

Our application allows the prescribing physician to compare his patient with similar retrospective cases. Based on the infection site and probable causative pathogens the possible severity of the current patient's situation is estimated and antibiotic substances for therapy are suggested. By considering possible side effects, contraindications, allergies, resistance information, and comorbidity a patient-specific solution is presented. Furthermore the physician is assisted in recognizing new resistant bacterial strains. Decisions are mostly based on retrospective information about pathogen occurrences. The potential pathogen spectra computed by our program correspond with pathogen distributions for UTI reported in literature. Even for cases with complications the spectra are plausible when compared with medical literature. Our segregation model safely distinguishes complicated from simple cases with the help of a set of rules using fuzzy logic. Our knowledge database and rules provide valid diagnoses and antibiotic suggestions. These results correlate with findings in specialized medical literature. The indication or contraindication label of the antibiotic substances is correctly determined by our knowledge database of antibiotics. The application is robust and processes all data in real time.

5.2.1 Comparison with similar systems

The comparison with other systems described in section "State of the art" is rather difficult. Each of these systems, including our application, is based on different data and implemented to use as much information from the underlying patient database as possible. None of these systems is easily portable to another hospital site because of the necessary adaptation to the local specific patient database. We selected our concept following the HELP system. We also generate files with condensed information from the retrospective database. Compared to other systems the following new features were implemented in our application:

- a graphical user interface for user interaction,
- rules for a given site of infection (UTIs) were extracted from medical literature,
- statistical data like the expected pathogen spectrum are presented, and
- a possible development of resistance in pathogens is identified.

5.2.2 Future plans

This application has shown the feasibility of a decision support system in the area of antibiotic prescriptions. With this application we generated antibiotic recommendations only for the field of UTIs. What lies at hand is to enlarge the knowledge base and add knowledge about other infection sites. Contraindication and allergy lists were filled to satisfy our purposes, but also may be supplemented. Here are some examples of possible extensions of this system:

- Algorithms for dose calculation: at the moment the information about the intake form is computed by the application, but is not output, because the computation of dosage is very complex and not yet implemented. Dose calculation requires the implementation of complex algorithms for each substance. Further patient information will be necessary such as weight and renal clearance.
- Reevaluation of initial therapy: At this point the application gives only recommendations about the initial therapy. Another feature could be the reevaluation of the initial therapy by considering cross- and co-resistances.

- Compute combinations of antibiotic substances for therapy
- Include information on pharmaceutic availability and cost to the recommendation
- Extend recommendations to other sites of infection
- Show trends in resistance development with cluster analysing sensitivity patterns
- Show other trends of antibiotic usage, changing pathogen spectra, etc.
- Include pharmaceutical information for each substance

6 Bibliography

[1] Cannon G. *Teufelskreis: wenn Antibiotika krank machen*. vgs Verlagsgesellschaft, Köln, 1994.

[2] Madani TA. Epidemiology and clinical features of methicillin-resistant staphylococcus aureus in the university hospital, Jeddah, Saudi Arabia. *The Canadian Journal of Infectious Diseases*, 13(4):245–250, 2002.

[3] Mullet CJ, Thomas JG, Smith CL, Sarwari AR, and Khakoo RA. Computerized antimicrobial decision support: an offline evaluation of a database-driven empiric antimicrobial guidance program in hospitalized patients with a bloodstream infection. *International Journal of Medical Informatics*, 73:455–460, 2004.

[4] Pestotnik SL. Expert clinical decision support systems to enhance antimicrobial stewardship programs. *Pharmacotherapy*, 25(8):1116–1125, 2005.

[5] Chizzali-Bonfadin C, Adlassnig K-P, and Koller W. Moni: an intelligent database and monitoring system for surveillance of nosocomial infections. In Greenes RA, Peterson HE, and Pratti DE, editors, *MEDINFO'95: Part 2*, page 1684. International Medical Informatics Association, Healthcare Computing and Communications Canada, Edmonton, Alberta, Canada, 1995.

[6] Kern-Isberner G and Beierle C. *Methoden wissensbasierter Systeme*. Vieweg, Braunschweig, 2003.

[7] Lehmann TM and Meyer zu Bexten E. *Handbuch der Medizinischen Informatik*. Hanser, München, 2002.

[8] Zadeh LA. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[9] Egly U, Eiter T, and Tompits H. *Skriptum zur Lehrveranstaltung "Einführung in Wissensbasierte Systeme"*. Institut für Informationssysteme, Abteilung Wissensbasierte Systeme, Technische Universität Wien, Wien, 2002.

[10] Adlassnig KP, Rappelsberger A, and Seising R. *Skriptum zur Lehrveranstaltung "Computerunterstützte Diagnoseverfahren"*. Institut für Medizinische Experten- und Wissensbasierte Systeme, Medizinische Universität Wien, Wien, 2005.

[11] Wikipedia. Case-based reasoning. `http://en.wikipedia.org/w/index.php?title=Case-based_reasoning&oldid=188969678`, 2008. [Accessed: 17 February 2008].

[12] Niedermeier S and Scholz M. *Java and XML*. Galileo Computing, Bonn, 2006.

[13] Edelson J and McLaughlin BD. *Java & XML*. O'Reilly, Beijing, third edition, 2007.

[14] Miksits K and Hahn H. *Basiswissen medizinische Mikrobiologie und Infektiologie*. Springer, Berlin, 1999.

[15] CDC and Stalons D. Staphylococcus aureus–antibiotics test plate. `http://en.wikipedia.org/wiki/Image:Staphylococcus_aureus_%28AB_Test%29.jpg`, 1972. [Accessed: 25 May 2008].

[16] Rübben H, Schulze H, and Sökeland J. *Urologie*. Georg Thieme Verlag, Stuttgart, 13. Auflage, 2004.

[17] Stille W, Brodt HR, Groll AH, and Just-Nübling G. *Antibiotika-Therapie – Klinik und Praxis der antiinfektiösen Behandlung*. Schattauer, Stuttgart, 2005.

[18] Djavan B. *Urologie und urologische Chirurgie*. facultas Universitätsverlag, Wien, 2006.

[19] Vogel F, Bodmann KF, and Paul Ehrlich Association. Empfehlungen zur kalkulierten parenteralen Initialtherapie bakterieller Erkrankungen bei Erwachsenen – Recommendations for calculated parenteral initial therapy of bacterial diseases in adults. *Chemotherapie Journal*, 13(2):46–105, 2004.

[20] Barber N. Education and debate: What constitutes good prescribing? *British Medical Journal*, 310:923–925, 1995.

[21] Kipp F, Friedrich AW, Becker K, and von Eiff C. Bedrohliche Zunahme Methicillin-resistenter Staphylococcus-aureus-Stämme: "Community-acquired MRSA". *Deutsches Aerzteblatt*, 102:1068–1073, 2004.

[22] Bohanec M, Rems M, Slavec S, and Urh B. Ptah: A system for supporting nosocomial infection therapy. In Lavrac N, Keravnou E, and Zupan B, editors, *Intelligent Data Analysis in Medicine and Pharmacology*, pages 99–111. Kluwer Academic Publishers, Boston,

1997.

[23] Vecna Robotics. http://www.vecnarobotics.com/medical/qcpathfinder.shtml, 2008. [Accessed: 06 August 2008].

[24] Schmidt R, Steffen D, and Gierl L. Evaluierung fallbasierter Antibiotikatherapie. In Jäckel A, editor, *Telemedizinführer Deutschland*, pages 185–190. Medizin Forum AG, Bad Nauheim, 2000.

[25] Evans RS. The help system: A review of clinical applications in infectious diseases and antibiotic use. *M.D. Computing*, 8:282–288, 1991.

[26] Lamma E, Mello P, Nanetti A, Riguzzi F, Storari S, and Valastro G. Artificial intelligence techniques for monitoring dangerous infections. *IEEE Transactions on Information Technology in Biomedicine*, 10(1):143–155, 2006.

[27] Paul M, Andreassen S, Tacconelli E, Nielsen AD, Almanasreh N, Frank U, Cauda R, and Leibovici L. Improving empirical antibiotic treatment using TREAT, a computerized decision support system: cluster randomized trial. *Journal of Antimicrobial Chemotherapy*, 58(6):1238–1245, 2006.

[28] Evans RS, Pestotnik SL, Classen DC, and Burke JP. Development of an automated antibiotic consultant. *M.D. Computing*, 10:17–22, 1993.

[29] Centers for Disease Control and Prevention. http://www.cdc.gov/, 2008. [Accessed: 06 August 2008].

[30] Evans RS, Classen DC, and Pestotnik SL. A decision support tool for antibiotic therapy. In Gardner RM, editor, *Nineteenth Annual Symposium on Computer Application in Medical Care*, pages 651–655. American Medical Informatics Association (Amia), Philadelphia: Hanley and Belfus, 1995.

[31] Evans RS, Pestotnik SL, Classen DC, Clemmer TP, Weaver LK, Orme JF, Lloyd JF, and Burke JP. A computer-assisted management program for antibiotics and other antiinfective agents. *New England Journal of Medicine*, 338(4):232–238, 1998.

[32] Schmidt R, Pollwein B, and Gierl L. Case-based reasoning for antibiotics therapy advice. In Althoff K-D, Bergmann R, and Branting K, editors, *Proceedings of the Third International Conference on Case-Based Reasoning and Development*, pages 550–559. Springer-Verlag, London, UK, 1999.

[33] Schmidt R and Gierl L. Case-based reasoning for antibiotics therapy advice: an investigation of retrieval algorithms and prototypes. *Artificial Intelligence in Medicine*, 23:171–186, 2001.

[34] Gierl L, Steffen D, Ihracky D, and Schmidt R. Methods, architecture, evaluation and usability of a case-based antibiotics advisor. *Computer Methods and Programs in Biomedicine*, 72:139–154, 2003.

[35] Hastie T, Tibshirani R, and Friedman J. *The Elements of Statistical Learning*. Springer, New York, 2001.

[36] Schlegel HG. *Allgemeine Mikrobiologie*. Georg Thieme Verlag, Stuttgart, 1992.

[37] Thalhammer F. *Antibiotika und Antiinfektiva*. Bayer, Wien, 10/2006.

[38] Brown DA and Mehlhorn AJ. Safety concerns with fluoroquinolons. *The Annals of Pharmacotherapy*, 41(11):1859–1866, 2007.

[39] Food and Drug Administration. FDA Drug Bulletin. `http://idahodur.isu.edu/leaflets/2005/Pregnancy%20Educational%20Leaflet%202005.pdf`, 1982. [Accessed: 17 August 2008].

[40] R environment for statistical computing. `http://www.r-project.org/`, 2008. [Accessed: 25 September 2008].

Appendix A: Decision trees

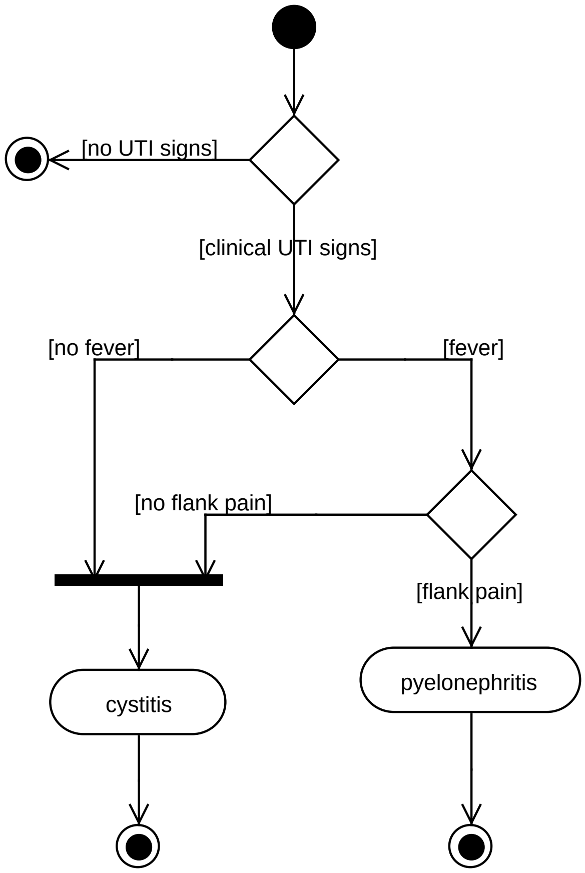


Figure A.1: Diagnosing a UTI based on patient's clinical signs

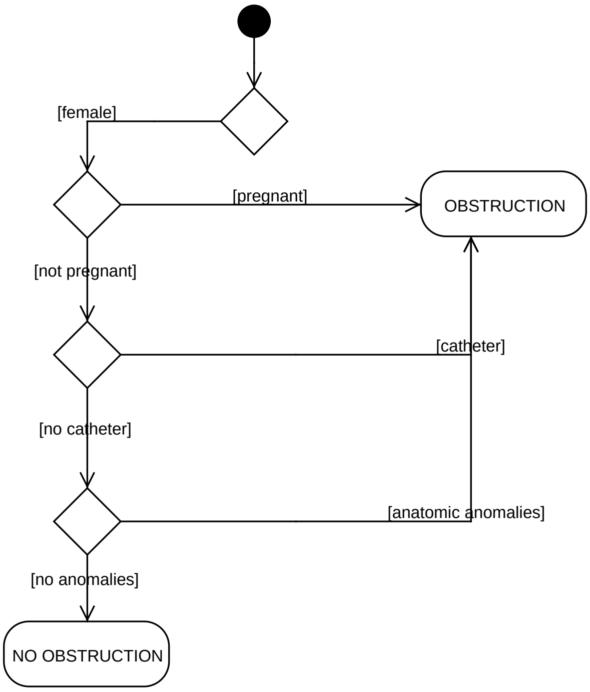


Figure A.2: Determining whether an obstruction exists

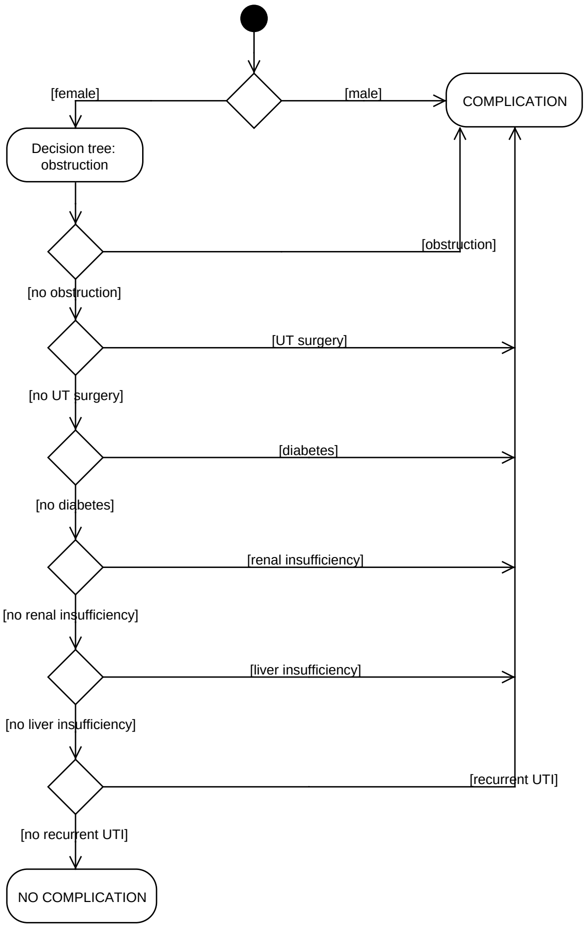


Figure A.3: Decision on complication based on patient's history and conditions

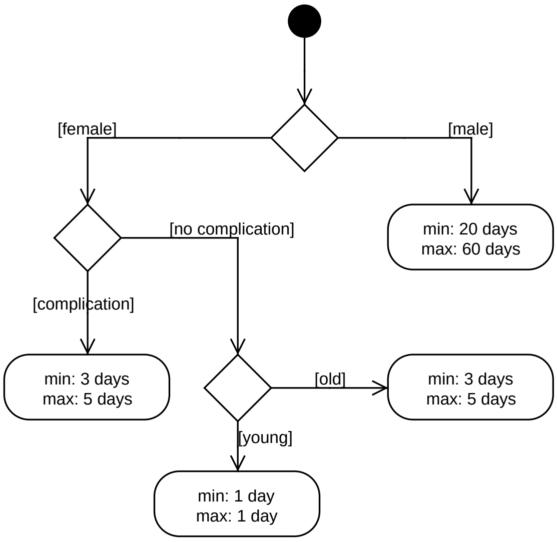


Figure A.4: Decision on the duration of therapy

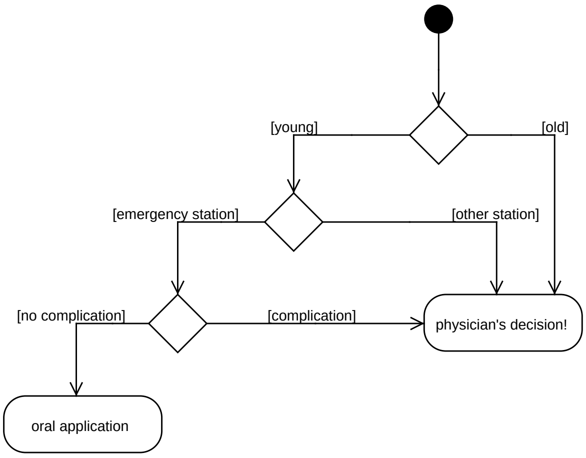


Figure A.5: Decision on the application form for the patient

Appendix B: FDA pregnancy category definitions

The FDA classifies antibiotic substances with respect to pregnancy into five categories reproduced from the FDA Drug Bulletin published in 1982 [39, pages 24–25].

- A – Controlled studies in women fail to demonstrate a risk to the fetus in the first trimester, and the possibility of fetal harm appears remote.
- B – Animal studies do not indicate a risk to the fetus and there are no controlled human studies, or animal studies do show an adverse effect on the fetus but well-controlled studies in pregnant women have failed to demonstrate a risk to the fetus.
- C – Studies have shown that the drug exerts animal teratogenic or embryocidal effects, but there are no controlled studies in women, or no studies are available in either animals or women.
- D – Positive evidence of human fetal risk exists, but benefits in certain situations (e.g., life-threatening situations or serious diseases for which safer drugs cannot be used or are ineffective) may make use of the drug acceptable despite its risks.
- X – Studies in animals or humans have demonstrated fetal abnormalities or there is evidence of fetal risk based on human experience, or both, and the risk clearly outweighs any possible benefit.

Appendix C: Statistical tests

For all statistical tests the program R [40] was used. The test input and test results are shown here. Data for these tests were taken from the retrospective database from the Hygiene Department of the Vienna General Hospital.

C.1 t-test: relationship between age and sex

Input:

```
t.test(femaleMaleData$Alter~femaleMaleData$Geschlecht.name, alternative="greater")
```

Results:

```
Result:
        Welch Two Sample t-test

data:  femaleMaleData$Alter by femaleMaleData$Geschlecht.name
t = -9.9244, df = 234.215, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -20.97373 -14.02577
sample estimates:
mean in group female   mean in group male
            35.55479             53.05455
```

C.2 Binomial test: influence of sex on expected pathogen spectrum

Input:

```
binom.test(79,134,p=0.877,alternative="less",conf.level=0.95)
```

Results:

```
     Exact binomial test

data:  79 and 134
number of successes = 79, number of trials = 134, p-value < 2.2e-16
alternative hypothesis: true probability of success is less than 0.877
95 percent confidence interval:
      0.0000000 0.6611397
sample estimates:
probability of success
             0.5895522
```

Appendix D: Knowledge database

In this appendix the knowledge database is shown in detail. Each table's create statement and its content are shown. How each table is used is described briefly. In Figure D.1 the entity relationship diagram of the whole database is shown.

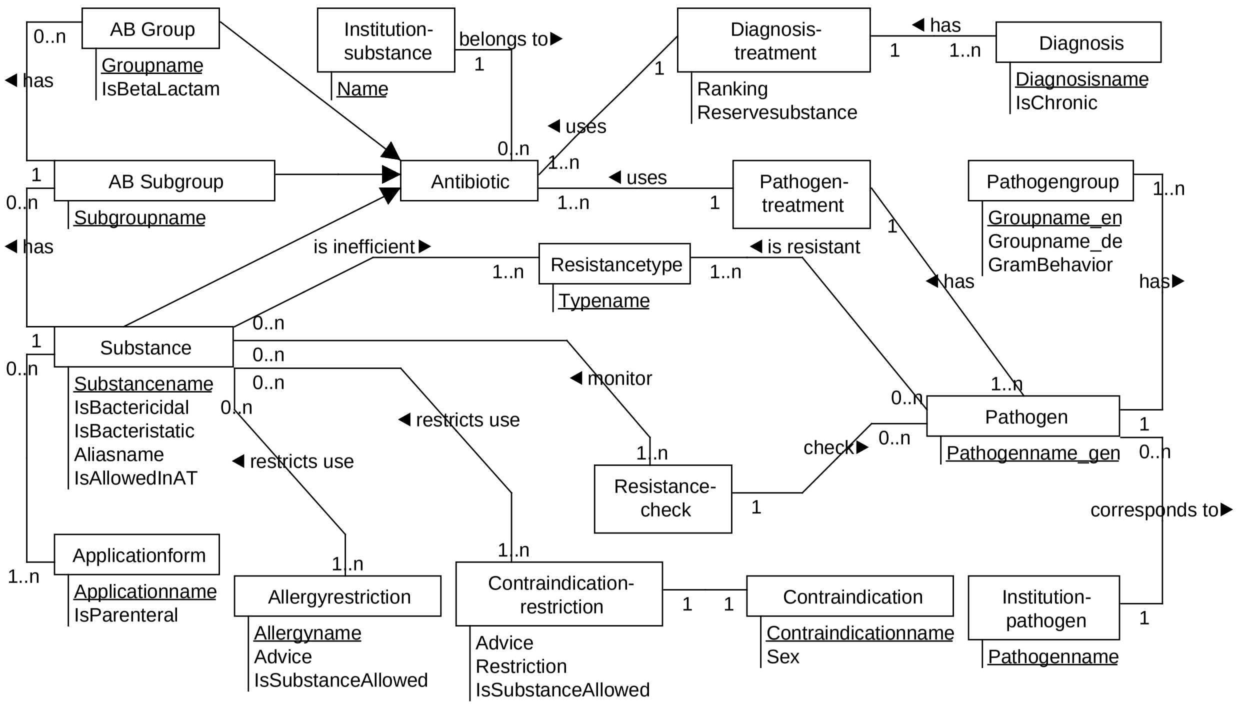


Figure D.1: Entity relation diagram of the underlying knowledge base

D.1 Antibiotic group

Table D.1 represents groups of antibiotics. The name of a group is the primary key. The column "betaLactam" may contain either true (T) or false (F), specifying if the group of antibiotics belongs to betalactam antibiotics, or not. This table is used to find general antibiotic names (see section 3.6.1). Treatment, allergies, or contraindications are often not defined for a certain substance, but for a group or subgroup of antibiotics. For example, if an antibiotic group, subgroup, or substance is specified as contraindicated the tables antibiotic group, subgroup, and substance must be searched through for all substances, which belong to the same group.

```
CREATE TABLE ABGroup(
    name varchar(100),
    betaLactam char(1) CONSTRAINT bl_constraint check(betaLactam in('T', 'F')),
    PRIMARY KEY(name)
);
```

Table D.1: Knowledge database table "ABGROUP" representing antibiotic groups. T (true) respectively F (false) state whether that the group belongs to the group of betalactam antibiotics, or not.

NAME	BETALACTAM
Penicillin	T
Cephalosporine	T
Carbapeneme	T
Monobactame	T
BetaLactamase-Inhibitor	T
MLS-Group	F
Aminoglycoside	F
Tetracyclin	F
GyraseInhibitor	F
Oxazolidinone	F
Nitroimidazole	F
Local	F
Other	F
Glycopeptide	F
Lipopeptide	F
Tuberculostatics	F
Unspecified	F

D.2 Antibiotic subgroup

Each antibiotic subgroup belongs to an antibiotic group. The use of this table is described in section D.1.

```
CREATE TABLE ABSubgroup(
    name varchar(100),
    groupname varchar(100),
    PRIMARY KEY(name),
    FOREIGN KEY(groupname) REFERENCES ABGroup(name)
);
```

Table D.2: Knowledge database table "ABSUBGROUP" representing antibiotic subgroups and their associated antibiotic group

NAME	GROUPNAME
Benzylpenicillin	Penicillin
Phenoxymethylpenicillin	Penicillin
Aminopenicillin	Penicillin
Penicillin + BLI	Penicillin
Isoxazolylpenicillin	Penicillin
Ureidopenicillin	Penicillin
Carboxypenicillin	Penicillin
Amidinopenicillin	Penicillin
1. Generation	Cephalosporine
2. Generation	Cephalosporine
3. Generation	Cephalosporine
4. Generation	Cephalosporine
Cefamycine	Cephalosporine
Oralcephalosporine 1. Generation	Cephalosporine
Oralcephalosporine 2. Generation	Cephalosporine
Oralcephalosporine 3. Generation	Cephalosporine
Carbapenem	Carbapeneme
Monobactam	Monobactame
BLI	BetaLactamase-Inhibitor
Macrolide	MLS-Group
Ketolide	MLS-Group
Streptogramin	MLS-Group
Lincosamid	MLS-Group
Old Aminoglycoside	Aminoglycoside
New Aminoglycoside	Aminoglycoside
Tetracyclin, Glycylcyclin	Tetracyclin
Old Fluoroquinolone	GyraseInhibitor
Fluoroquinolone Group 1	GyraseInhibitor
Fluoroquinolone Group 2	GyraseInhibitor
Fluoroquinolone Group 3	GyraseInhibitor

further data see next page

Table D.2: Continued

NAME	GROUPNAME
Fluoroquinolone Group 4	GyraseInhibitor
Oxazolidinon	Oxazolidinone
Nitroimidazol	Nitroimidazole
Local	Local
Chloramphenicol-Group	Other
Polymyxine	Other
Sulfonamid-Combination	Other
Epoxyd	Other
Fusidic Acid	Other
Rifamycine	Other
Trimethoprim	Other
Glycopeptid	Glycopeptide
Lipopeptid	Lipopeptide
Tuberculostatic	Tuberculostatics
U	Unspecified

D.3 Antibiotic substance

Table D.3 is the main table of the knowledge base. It contains information about antibiotic substances. Each antibiotic substance belongs to an antibiotic subgroup. Knowledge in this table is accessed as described in section D.1 and each time resistance for a pathogen is assessed (see section 3.8).

```
CREATE TABLE Substance (
    name varchar(100),
    isbactericidal char(1) CONSTRAINT cidal_constraint
     check(bactericidal in('Y', 'N','U')),
    isbacteristatic char(1) CONSTRAINT static_constraint
     check(bacteristatic in('Y', 'N','U')),
    subgroupname varchar(100),
    aliasName varchar(100),
    isallowedInAt char(1) CONSTRAINT allowed_constraint
     check(allowedInAt in('Y', 'N','U')),
    PRIMARY KEY(name),
    FOREIGN KEY(subgroupname) REFERENCES ABSubgroup(name)
);
```

Columns "isbactericidal", "isbacteristatic", and "isallowedinat" may contain the first letter of the following three values: yes, no, or unknown. The aliasname contains so far only the value "U" for unknown.

Table D.3: Knowledge database table "SUBSTANCE" representing antibiotic groups

NAME	ISBACTERICIDAL	ISBACTERISTATIC	SUBGROUPNAME	ALIASNAME	ISALLOWEDINAT
Penicillin G	Y	N	Benzylpenicillin	U	N
Aminobenzylpenicillin	Y	N	Benzylpenicillin	U	N
Benzylpenicillin	Y	N	Benzylpenicillin	U	N
Penicillin V	Y	N	Phenoxymethylpenicillin	U	N
Phenoxymethylpenicillin / Kalium	Y	N	Phenoxymethylpenicillin	U	N
Amoxicillin	Y	N	Aminopenicillin	U	N
Ampicillin	Y	N	Aminopenicillin	U	N
Bacampicillin	Y	N	Aminopenicillin	U	N
Piperacillin/Tazobactam	Y	N	Penicillin + BLI	U	N
Amoxicillin/Clavulanate acid	Y	N	Penicillin + BLI	U	N
Aminobenzylpenicillin + BLI	Y	N	Penicillin + BLI	U	N
Ampicillin/Sulbactam	Y	N	Penicillin + BLI	U	N
Flucloxacillin	Y	N	Isoxazolylpenicillin	U	N
Oxacillin	Y	N	Isoxazolylpenicillin	U	N
Mezlocillin	Y	N	Ureidopenicillin	U	N
Azlocillin	Y	N	Ureidopenicillin	U	N
Piperacillin	Y	N	Ureidopenicillin	U	N
Temocillin	Y	N	Carboxypenicillin	U	Y
Ticarcillin	Y	N	Carboxypenicillin	U	Y
Pivmecillinam	Y	N	Amidinopenicillin	U	N
Cefaloridin	Y	N	1. Generation	U	Y
Cefalotin	Y	N	1. Generation	U	Y
Cefazolin	Y	N	1. Generation	U	N
Cefamandol	Y	N	2. Generation	U	N
Cefotiam	Y	N	2. Generation	U	N
Cefuroxim	Y	N	2. Generation	U	N
Cefodizim	Y	N	3. Generation	U	N
Cefoperazon	Y	N	3. Generation	U	N
Cefotaxim	Y	N	3. Generation	U	N
Ceftazidim	Y	N	3. Generation	U	N
Ceftizoxim	Y	N	3. Generation	U	Y
Ceftriaxon	Y	N	3. Generation	U	N

further data see next page

Table D.3: Continued

NAME	ISBACTERICIDAL	ISBACTERISTATIC	SUBGROUPNAME	ALIASNAME	ISALLOWEDINAT
Cefepim	Y	N	4. Generation	U	N
Cefpirom	Y	N	4. Generation	U	N
Cefoxitin	Y	N	Cefamycine	U	N
Latamoxef	Y	N	Cefamycine	U	Y
Cefalexin	Y	N	Oralcephalosporine 1. Generation	U	N
Cefaclor	Y	N	Oralcephalosporine 1. Generation	U	N
Cefadroxil	Y	N	Oralcephalosporine 1. Generation	U	N
Cefuroxim/Axetil	Y	N	Oralcephalosporine 2. Generation	U	N
Cefixim	Y	N	Oralcephalosporine 3. Generation	U	N
Cefpodoxim/Proxetil	Y	N	Oralcephalosporine 3. Generation	U	N
Imipenem/Cilastatin	Y	N	Oralcephalosporine 3. Generation	U	N
Meropenem	Y	N	Oralcephalosporine 3. Generation	U	N
Ertapenem	Y	N	Oralcephalosporine 3. Generation	U	N
Aztreonam	Y	N	Oralcephalosporine 3. Generation	U	N
Sulbactam	Y	N	Oralcephalosporine 3. Generation	U	N
Josamycin	N	Y	Macrolide	U	N
Azithromycin	N	Y	Macrolide	U	N
Clarithromycin	N	Y	Macrolide	U	N
Erythromycin	N	Y	Macrolide	U	N
Roxithromycin	N	Y	Macrolide	U	N
Spiramycin	N	Y	Macrolide	U	N
Telithromycin	N	Y	Ketolide	U	N
Quinupristin/Dalfopristin	N	Y	Streptogramin	U	N
Clindamycin	N	Y	Lincosamid	U	N
Lincomycin	N	Y	Lincosamid	U	Y
Kanamycin	U	U	Old Aminoglycoside	U	Y
Neomycin	U	U	Old Aminoglycoside	U	N
Paromomycin	U	U	Old Aminoglycoside	U	N
Amikacin	U	U	New Aminoglycoside	U	N
Gentamicin	U	U	New Aminoglycoside	U	N
Netilmicin	U	U	New Aminoglycoside	U	N
Tobramycin	U	U	New Aminoglycoside	U	N

further data see next page

Table D.3: Continued

NAME	ISBACTERICIDAL	ISBACTERISTATIC	SUBGROUPNAME	ALIASNAME	ISALLOWEDINAT
Doxycyclin	N	Y	Tetracyclin, Glycylcyclin	U	N
Minocyclin	N	Y	Tetracyclin, Glycylcyclin	U	N
Tetracyclin	N	Y	Tetracyclin, Glycylcyclin	U	N
Tigecyclin	N	Y	Tetracyclin, Glycylcyclin	U	N
Oxolin acid	Y	N	Old Fluoroquinolone	U	Y
Lomefloxacin	Y	N	Fluoroquinolone Group 1	U	N
Norfloxacin	Y	N	Fluoroquinolone Group 1	U	N
Ofloxacin	Y	N	Fluoroquinolone Group 2	U	N
Ciprofloxacin	Y	N	Fluoroquinolone Group 2	U	N
Enoxacin	Y	N	Fluoroquinolone Group 2	U	Y
Levofloxacin	Y	N	Fluoroquinolone Group 3	U	N
Gemifloxacin	Y	N	Fluoroquinolone Group 4	U	Y
Moxifloxacin	Y	N	Fluoroquinolone Group 4	U	N
Linezolid	U	U	Fluoroquinolone Group 4	U	N
Metronidazol	Y	N	Fluoroquinolone Group 4	U	N
Tinidazol	Y	N	Fluoroquinolone Group 4	U	N
Bacitracin	U	U	Fluoroquinolone Group 4	U	N
Tyrothricin	U	U	Fluoroquinolone Group 4	U	N
Chloramphenicol	N	Y	Chloramphenicol-Group	U	N
Colistin	Y	N	Polymyxine	U	N
Cotrimoxazol	Y	N	Sulfonamid-Combination	U	N
Trimethoprim/Sulfamethoxazol	N	Y	Sulfonamid-Combination	U	N
Trimethoprim/Sulfametrol	N	Y	Sulfonamid-Combination	U	N
Fosfomycin	Y	N	Epoxyd	U	N
Fosfomycin-Trometamol	Y	N	Epoxyd	U	N
FusidicAcid	N	Y	Fusidic Acid	U	N
Rifampicin	Y	N	Rifamycine	U	N
Trimethoprim	N	Y	Trimethoprim	U	N
Teicoplanin	Y	N	Glycopeptid	U	N
Vancomycin	Y	N	Glycopeptid	U	N
Daptomycin	Y	N	Lipopeptid	U	N
Ethambutol	N	Y	Tuberculostatic	U	N

further data see next page

Table D.3: Continued

NAME	ISBACTERICIDAL	ISBACTERISTATIC	SUBGROUPNAME	ALIASNAME	ISALLOWEDINAT
Isoniazid (INH)	N	Y	Tuberculostatic	U	N
Protionamid	Y	N	Tuberculostatic	U	N
Pyrazinamid	N	Y	Tuberculostatic	U	N
Rifabutin	N	Y	Tuberculostatic	U	N

D.4 Institution-specific substance names

Table D.4 contains institution-specific substance names and is necessary to translate them to general substance names specified in the substance Table D.3. An institution-specific name of a substance does not necessarily have to be a substance name, it may also be a name for a group or subgroup of antibiotics. For further details see section 3.6.1.

```
CREATE TABLE InstitutionSubstance (
    name varchar(100),
    substanceName varchar(100),
    subgroupName varchar(100),
    groupName varchar(100),
    FOREIGN KEY(substanceName) REFERENCES Substance(name),
    FOREIGN KEY(subgroupName) REFERENCES ABSubgroup(name),
    FOREIGN KEY(groupName) REFERENCES ABGroup(name)
);
```

Table D.4: Knowledge database table "Institutionsubstance"

NAME	SUBSTANCENAME	SUBGROUPNAME	GROUPNAME
5-Flucytosin	None	U	Unspecified
Amikacin	Amikacin	New Aminoglycoside	Unspecified
Aminobz.pen. + BLI	Aminobenzylpenicillin + BLI	Penicillin + BLI	Unspecified
Aminobz.pen. + BLI (n)	Aminobenzylpenicillin + BLI	Penicillin + BLI	Unspecified
Aminobz.pen.(a)	Aminobenzylpenicillin	Benzylpenicillin	Unspecified
Amoxicil./Clavulansre.	Amoxicillin/Clavulanate acid	Penicillin + BLI	Unspecified
Amoxicillin	Amoxicillin	Aminopenicillin	Unspecified
Amphotericin B	None	U	Unspecified
Azlocillin (b)	Azlocillin	Ureidopenicillin	Unspecified
Aztreonam	Aztreonam	Oralcephalosporine 3. Generation	Unspecified
Bacitracin	Bacitracin	Fluoroquinolone Group 4	Unspecified
Carbapeneme (g)	None	Carbapenem	Carbapeneme
Cefaclor	Cefaclor	Oralcephalosporine 1. Generation	Unspecified
Cefalotin	Cefalotin	1. Generation	Unspecified
Cefamandol (c)	Cefamandol	2. Generation	Unspecified
Cefepime	Cefepim	4. Generation	Unspecified
Cefotaxim	Cefotaxim	3. Generation	Unspecified
Cefotaxim (d)	Cefotaxim	3. Generation	Unspecified
Cefoxitin	Cefoxitin	Cefamycine	Unspecified
Cefpirom	Cefpirom	4. Generation	Unspecified
Cefpodoxime	Cefpodoxim/Proxetil	Oralcephalosporine 3. Generation	Unspecified
Ceftazidim	Ceftazidim	3. Generation	Unspecified
Cefuroxim	Cefuroxim	2. Generation	Unspecified
Cefuroxime Oral	Cefuroxim/Axetil	Oralcephalosporine 2. Generation	Unspecified
Cephalospor. I + II	None	2. Generation	Unspecified
Cephalospor. I-III (g)	None	3. Generation	Unspecified
Cephalospor. I-IV (g)	None	4. Generation	Unspecified

further data see next page

Table D.4: Continued

NAME	SUBSTANCENAME	SUBGROUPNAME	GROUPNAME
Cephalosporine I-IV	None	U	Unspecified
Cephazolin (d)	Cefazolin	1. Generation	Unspecified
Chloramphenicol	Chloramphenicol	Chloramphenicol-Group	Unspecified
Ciprofloxacin	Ciprofloxacin	Fluoroquinolone Group 2	Unspecified
Ciproxin	Ciprofloxacin	Fluoroquinolone Group 2	Unspecified
Clarithromycin	Clarithromycin	Macrolide	Unspecified
Clindamycin	Clindamycin	Lincosamid	Unspecified
Clotrimazol	None	U	Unspecified
Colistin	Colistin	Polymyxine	Unspecified
Cotrimoxazol	Cotrimoxazol	Sulfonamid-Combination	Unspecified
Doxycyclin	Doxycyclin	Tetracyclin, Glycylcyclin	Unspecified
Econazol	None	U	Unspecified
Erythromycin	Erythromycin	Macrolide	Unspecified
Erythromycin (j)	Erythromycin	Macrolide	Unspecified
Fluconazol	None	U	Unspecified
Fosfomycin	Fosfomycin	Epoxyd	Unspecified
Fusidinsäure	FusidicAcid	Fusidic Acid	Unspecified
Gentamicin	Gentamicin	New Aminoglycoside	Unspecified
Gentamicin (e)	Gentamicin	New Aminoglycoside	Unspecified
Gentamicin (h)	Gentamicin	New Aminoglycoside	Unspecified
Imipenem	Imipenem/Cilastatin	Oralcephalosporine 3. Generation	Unspecified
Isepamicin	None	U	Unspecified
Isoxazolylpenicillin	Oxacillin	Isoxazolylpenicillin	Unspecified
Itraconazol	None	U	Unspecified
Josamycin	Josamycin	Macrolide	Unspecified
Ketoconazol	None	U	Unspecified
Lincomycin	Lincomycin	Lincosamid	Unspecified
Linezolid	Linezolid	Fluoroquinolone Group 4	Unspecified
Mecillinam	Pivmecillinam	Amidinopenicillin	Unspecified
Meropenem	Meropenem	Oralcephalosporine 3. Generation	Unspecified
Metronidazol	Metronidazol	Fluoroquinolone Group 4	Unspecified
Miconazol	None	U	Unspecified
Minocycline	Minocyclin	Tetracyclin, Glycylcyclin	Unspecified
Moxifloxacin	Moxifloxacin	Fluoroquinolone Group 4	Unspecified
Mupirocin	None	U	Unspecified
Neomycin	Neomycin	Old Aminoglycoside	Unspecified
Netilmicin	Netilmicin	New Aminoglycoside	Unspecified
Nitrofurantoin	None	U	Unspecified
Norfloxacin	Norfloxacin	Fluoroquinolone Group 1	Unspecified
Nystatin	None	U	Unspecified
Ofloxacin (k)	Ofloxacin	Fluoroquinolone Group 2	Unspecified
Pefloxacin	None	U	Unspecified
Penicillin-G	Penicillin G	Benzylpenicillin	Unspecified
Piperacillin	Piperacillin	Ureidopenicillin	Unspecified
Piperacillin/Tazobactam	Piperacillin/Tazobactam	Penicillin + BLI	Unspecified
Pristinamycin	None	U	Unspecified
Quinu-/Dalfopristin	Quinupristin/Dalfopristin	Streptogramin	Unspecified

further data see next page

Table D.4: Continued

NAME	SUBSTANCENAME	SUBGROUPNAME	GROUPNAME
Rifampicin	Rifampicin	Rifamycine	Unspecified
Streptomycin	None	U	Unspecified
Sulfon. + Trim.	Trimethoprim/Sulfamethoxazol	Sulfonamid-Combination	Unspecified
Sulfonamide	None	Sulfonamid-Combination	Unspecified
Teicoplanin	Teicoplanin	Glycopeptid	Unspecified
Tetrazykline	Tetracyclin	Tetracyclin, Glycylcyclin	Unspecified
Tobramycin	Tobramycin	New Aminoglycoside	Unspecified
Trimethoprim	Trimethoprim	Trimethoprim	Unspecified
Trovafloxacin	None	U	Unspecified
Vancomycin	Vancomycin	Glycopeptid	Unspecified

D.5 Resistance types

Table D.5 specifies resistance types and the causative antibiotic substance, see section 3.8.

```
CREATE TABLE ResistanceTypes(
    name varchar(100),
    substanceName varchar(100),
    PRIMARY KEY(name),
    FOREIGN KEY(substanceName) REFERENCES Substance(name)
);
```

Table D.5: Knowledge database table "Resistancetypes"

NAME	SUBSTANCENAME
ESBL	Cefepim
MRSA	Oxacillin
VRE	Vancomycin

D.6 Resistance check

In Table D.6 it is specified for which substances which pathogen must especially be monitored. This knowledge is used in section 3.8.

```
CREATE TABLE ResistanceCheck (
    pathogenName varchar(200),
    substanceName varchar(100),
    FOREIGN KEY(pathogenName) REFERENCES Pathogen(name),
    FOREIGN KEY(substanceName) REFERENCES Substance(name)
);
```

Table D.6: Knowledge database table "Resistancecheck"

PATHOGENNAME	SUBSTANCENAME
Escherichia coli	Levofloxacin
Escherichia coli	Ciprofloxacin

D.7 Pathogen group

The pathogen group is analogous to the antibiotic group. This knowledge is not yet used in the application. The column gram behavior may contain one of three values: negative (n), positive (p), or other (o), the latter is used only for fungi, which do not have a gram behavior.

```
CREATE TABLE PathogenGroup (
    name_en varchar(100),
    name_de varchar(100),
    gramBehavior varchar(25),
    PRIMARY KEY(name_en)
);
```

Table D.7: Knowledge database table "Pathogengroup"

NAME EN	NAME DE	GRAMBEHAVIOR
Acetobacter	U	n
Achromobacter	U	n
Acidaminococcus	U	n
Acinetobacter	U	n
Actinobacillus	U	n
Actinomyces	U	p
Actinoplanes	U	p
Aerococcus	U	p
Aeromonas	U	n
Agrobacterium	U	n
Alcaligenes	U	n
Aquaspirillum	U	p

further data see next page

Table D.7: Continued

NAME EN	NAME DE	GRAMBEHAVIOR
Arthrobacter	U	p
Azomonas	U	n
Azospirillum	U	p
Bacillus	U	p
Bacteroides	U	n
Bdellovibrio	U	p
Beijerinckia	U	n
Bifidobacterium	U	p
Borrelia	U	p
Brevibacterium	U	p
Brucella	U	n
Burkholderia	U	n
Butyvibrio	U	p
Campylobacter	U	n
Candida	U	o
Caryophanon	U	p
Cellulomas	U	p
Citrobacter	U	n
Clostridium	U	p
Cocci	Kokken	p
Cocci (anaerob)	Kokken	p
Corynebacterium	U	p
Crenothrix	U	n
Cristispira	U	p
Cytophaga	U	n
Dermabacter	U	p
Dermatophilus	U	p
Derxia	U	n
Desulfotomaculum	U	p
Desulfovibrio	U	p
Desulfurococcus	U	n
Enterobacter	U	n
Enterococcus	Enterokokken	p
Erwinia	U	n
Erysipelothrix	U	p
Escherichia	U	n
Eubacterium	U	p
Fibrobacter	U	n
Flavimonas	U	n
Frankia	U	p
Fungi	U	o
Fusobacterium	U	n
Glucnobacter	U	n
Granulicatella	U	p
Haemophilus	U	n
Hafnia	U	n

further data see next page

Table D.7: Continued

NAME EN	NAME DE	GRAMBEHAVIOR
Halobacterium	U	n
Halococcus	U	n
Haloferax	U	n
Helicobacter	U	p
Klebsiella	U	n
Kocuria	U	p
Lactobacillus	U	p
Lampropedia	U	n
Legionella	U	n
Leptospira	U	p
Leptothrix	U	n
Leptotrichia	U	n
Leuconostoc	Leukonostoc	p
Listeria	U	p
Megasphaera	U	n
Methanobacterium	U	n
Methanococcus	U	n
Methanosarcina	U	n
Methanothermus	U	n
Methanotrix	U	n
Microbispora	U	p
Micrococcus	U	p
Microcyclus	U	p
Micromonospora	U	p
Moraxella	U	n
Morganella	U	n
Mycobacterium	U	p
Neisseria	U	n
Nitrobacter	U	n
Nitrococcus	U	n
Nitrosococcus	U	n
Nitrosolobus	U	n
Nitrosomonas	U	n
Nitrosospira	U	n
Nitrospina	U	n
Nocardia	U	p
Oceanospirillum	U	p
Oscillospira	U	p
Pantoea	U	n
Paracoccus	Parakokken	n
Pasteurella	U	n
Pediococcus	Pediokokken	p

D.8 Pathogen

Pathogen names are used to retrieve the correct pathogen treatment (see sections 3.7.2, 4.4.4), or to check resistance (see section 3.8).

```
CREATE TABLE Pathogen (
    name varchar(200),
    groupName varchar(100),
    PRIMARY KEY(name),
    FOREIGN KEY(groupName) REFERENCES PathogenGroup(name_en)
);
```

Table D.8: Knowledge database table "Pathogen"

NAME	GROUPNAME
Achromobacter (Alcaligenes) xylosoxidans	Achromobacter
Acinetobacter baumannii	Acinetobacter
Acinetobacter calcoaceticus	Acinetobacter
Acinetobacter junii	Acinetobacter
Acinetobacter lwoffii	Acinetobacter
Acinetobacter species	Acinetobacter
Actinobacillus actinomycetemcomitans	Actinobacillus
Aerococcus species	Aerococcus
Aeromonas hydrophila	Aeromonas
Alcaligenes xylosoxidans	Alcaligenes
Anhaemolysierende Streptokokken	Streptococcus
Bacillus cereus	Bacillus
Bacillus species	Bacillus
Bacteroides caccae	Bacteroides
Bacteroides distasonis (anaerob)	Bacteroides
Bacteroides fragilis (anaerob)	Bacteroides
Bacteroides species (anaerob)	Bacteroides
Bacteroides thetaiotaomicron (anaerob)	Bacteroides
Bacteroides uniformis (anaerob)	Bacteroides
Bifidobacterium species (anaerob)	Bifidobacterium
Brucella melitensis	Brucella
Burkholderia cepacia	Burkholderia
Burkholderia species	Burkholderia
Campylobacter jejuni	Campylobacter
Candida albicans	Candida
Candida glabrata	Candida
Candida krusei (Fluconazol Resistant)	Candida
Candida lusitaniae	Candida
Candida melibiosica	Candida
Candida parapsilosis	Candida
Candida species	Candida
Candida tropicalis	Candida

further data see next page

Table D.8: Continued

NAME	**GROUPNAME**
Capnocytophaga species	Cytophaga
Citrobacter braakii	Citrobacter
Citrobacter freundii	Citrobacter
Citrobacter koseri	Citrobacter
Citrobacter youngae	Citrobacter
Clostridium paraputrificum (anaerob)	Clostridium
Clostridium perfringens (anaerob)	Clostridium
Clostridium putrificum (anaerob)	Clostridium
Clostridium ramosum (anaerob)	Clostridium
Clostridium species (anaerob)	Clostridium
Clostridium symbiosum (anaerob)	Clostridium
Clostridium tertium (anaerob)	Clostridium
Corynebacterium propinquum	Corynebacterium
Corynebacterium pseudodiphteriae	Corynebacterium
Corynebacterium species	Corynebacterium
Corynebacterium striatum	Corynebacterium
Dermabacter hominis	Dermabacter
Enterobacter aerogenes	Enterobacter
Enterobacter amnigenus 1	Enterobacter
Enterobacter cloacae	Enterobacter
Enterobacter cloacae ESßL	Enterobacter
Enterobacter sakazakii	Enterobacter
Enterobacter species	Enterobacter
Enterococcus avium	Enterococcus
Enterococcus durans	Enterococcus
Enterococcus faecalis	Enterococcus
Enterococcus faecium	Enterococcus
Enterococcus faecium VRE	Enterococcus
Enterococcus gallinarum	Enterococcus
Enterococcus species	Enterococcus
Escherichia coli	Escherichia
Escherichia coli ESßL	Escherichia
Flavimonas oryzihabitans	Flavimonas
Fusobacterium necrophorum (anaerob)	Fusobacterium
Fusobacterium nucleatum (anaerob)	Fusobacterium
Fusobacterium species (anaerob)	Fusobacterium
Fusobacterium varium (anaerob)	Fusobacterium
Gramnegative Stäbchen	Rods
Gramnegative Stäbchen (anaerob)	Rods (anaerob)
Grampositive Kokken	Cocci
Grampositive Kokken (anaerob)	Cocci (anaerob)
Granulicatella adiacens	Granulicatella
Granulicatella elegans	Granulicatella
Haemophilus influenzae	Haemophilus
Haemophilus parainfluenzae	Haemophilus
Haemophilus species	Haemophilus
Hafnia alvei	Hafnia

further data see next page

Table D.8: Continued

NAME	GROUPNAME
Klebsiella ornithinolytica	Klebsiella
Klebsiella oxytoca	Klebsiella
Klebsiella pneumoniae	Klebsiella
Klebsiella pneumoniae ESßL	Klebsiella
Klebsiella species ESßL	Klebsiella
Klebsiella terrigena	Klebsiella
Kocuria rosea	Kocuria
Kocuria species	Kocuria
Lactobacillus casei	Lactobacillus
Lactobacillus fermenti	Lactobacillus
Lactobacillus species	Lactobacillus
Leuconostoc species	Leuconostoc
Listeria monocytogenes	Listeria
Micrococcus luteus	Micrococcus
Micrococcus species	Micrococcus
Moraxella atlantae	Moraxella
Moraxella osloensis	Moraxella
Morganella morganii	Morganella
Neisseria meningitidis	Neisseria
Neisseria meningitidis Group B	Neisseria
Neisseria species	Neisseria

D.9 Institution-specific pathogen names

Institution-specific pathogen names are used analogously to names of institution-specific substances.

```
CREATE TABLE InstitutionPathogen(
    name varchar(200),
    pathogenName varchar(200),
    Foreign key(pathogenName) references Pathogen(name)
);
```

Table D.9: Knowledge database table "Institutionpathogen"

NAME	PATHOGENNAME
22 Achromobacter (Alcaligenes) xylosoxidans	Achromobacter (Alcaligenes) xylosoxidans
1050 Acinetobacter baumannii	Acinetobacter baumannii
1080 Acinetobacter junii	Acinetobacter junii
1079 Acinetobacter lwoffii	Acinetobacter lwoffii
26 Actinobacillus actinomycetemcomitans	Actinobacillus actinomycetemcomitans
517 Aerococcus species	Aerococcus species
22 Alcaligenes xylosoxidans	Alcaligenes xylosoxidans
36 Bacillus cereus	Bacillus cereus
1076 Bacteroides caccae	Bacteroides caccae
1101 Burkholderia cepacia	Burkholderia cepacia
1149 Burkholderia species	Burkholderia species
111 Campylobacter jejuni	Campylobacter jejuni
21 Candida albicans	Candida albicans
105 Candida krusei (Fluconazol Resistant)	Candida krusei (Fluconazol Resistant)
112 Candida lusitaniae	Candida lusitaniae
119 Candida melibiosica	Candida melibiosica
136 Candida parapsilosis	Candida parapsilosis
381 Candida species (nicht albicans)	Candida species
176 Candida tropicalis	Candida tropicalis
1134 Citrobacter braakii	Citrobacter braakii
123 Citrobacter freundii	Citrobacter freundii
1059 Citrobacter koseri	Citrobacter koseri
1140 Citrobacter youngae	Citrobacter youngae
1132 Corynebacterium propinquum	Corynebacterium propinquum
1119 Dermabacter hominis	Dermabacter hominis
137 Enterobacter amnigenus 1	Enterobacter amnigenus 1
138 Enterobacter cloacae	Enterobacter cloacae
1213 Enterobacter cloacae ESßL	Enterobacter cloacae ESßL
141 Enterobacter sakazakii	Enterobacter sakazakii
1210 Enterococcus faecium VRE	Enterococcus faecium VRE
1098 Enterococcus gallinarum	Enterococcus gallinarum
133 Escherichia coli	Escherichia coli
1178 Escherichia coli ESßL	Escherichia coli ESßL
1054 Flavimonas oryzihabitans	Flavimonas oryzihabitans
157 Fusobacterium necrophorum (anaerob)	Fusobacterium necrophorum (anaerob)
160 Fusobacterium varium (anaerob)	Fusobacterium varium (anaerob)
1180 Granulicatella adiacens	Granulicatella adiacens
1181 Granulicatella elegans	Granulicatella elegans
171 Haemophilus influenzae	Haemophilus influenzae
176 Haemophilus parainfluenzae	Haemophilus parainfluenzae
164 Hafnia alvei	Hafnia alvei
1093 Klebsiella ornithinolytica	Klebsiella ornithinolytica
180 Klebsiella oxytoca	Klebsiella oxytoca
183 Klebsiella pneumoniae	Klebsiella pneumoniae
1215 Klebsiella pneumoniae ESßL	Klebsiella pneumoniae ESßL
1183 Klebsiella species ESßL	Klebsiella species ESßL
185 Klebsiella terrigena	Klebsiella terrigena

further data see next page

Table D.9: Continued

NAME	PATHOGENNAME
1138 Kocuria rosea	Kocuria rosea
1137 Kocuria species	Kocuria species
1018 Leuconostoc species	Leuconostoc species
1014 Listeria monocytogenes	Listeria monocytogenes
519 Micrococcus species	Micrococcus species
207 Morganella morganii	Morganella morganii
1034 Neisseria meningitidis Gruppe B	Neisseria meningitidis Group B
1128 Pantoea species	Pantoea species
383 Pasteurella multocida	Pasteurella multocida
285 Peptostreptococcus anaerobius (anaerob)	Peptostreptococcus anaerobius (anaerob)
288 Peptostreptococcus magnus (anaerob)	Peptostreptococcus magnus (anaerob)
289 Peptostreptococcus micros (anaerob)	Peptostreptococcus micros (anaerob)
272 Propionibacterium acnes (anaerob)	Propionibacterium acnes (anaerob)
278 Proteus mirabilis	Proteus mirabilis
281 Proteus vulgaris	Proteus vulgaris
295 Pseudomonas aeruginosa	Pseudomonas aeruginosa
330 Pseudomonas fluorescens	Pseudomonas fluorescens
360 Pseudomonas putida	Pseudomonas putida
1071 Ralstonia pickettii (Burkholderia pickettii)	Ralstonia pickettii
408 Salmonella Gr.D	Salmonella Gr.D
411 Salmonella Serotyp Enteritidis	Salmonella Serotyp Enteritidis
413 Salmonella typhi	Salmonella Serotyp Typhi
407 Salmonella virchow	Salmonella Serotyp Virchow
420 Serratia liquefaciens	Serratia liquefaciens
422 Serratia marcescens	Serratia marcescens
430 Shigella sonnei	Shigella sonnei
1084 Sphingomonas paucimobilis	Sphingomonas paucimobilis
1086 Sphingomonas paucimobilis	Sphingomonas paucimobilis
335 Sproßpilze	Sproßpilze
522 Staphylococcus aureus	Staphylococcus aureus
521 Staphylococcus auricularis	Staphylococcus auricularis
523 Staphylococcus capitis	Staphylococcus capitis
1129 Staphylococcus chromogenes	Staphylococcus chromogenes
1037 Staphylococcus lugdunensis	Staphylococcus lugdunensis
1026 Staphylococcus saccharolyticus (anaerob)	Staphylococcus saccharolyticus (anaerob)
1230 Stapylococcus aureus MRSA	Stapylococcus aureus MRSA
1047 Stenotrophomonas maltophilia	Stenotrophomonas maltophilia
1231 Stenotrophomonas maltophilia	Stenotrophomonas maltophilia
433 Stomatococcus mucilaginosus	Stomatococcus mucilaginosus
1088 Streptococcus acidominimus	Streptococcus acidominimus
435 Streptococcus agalactiae	Streptococcus agalactiae
436 Streptococcus anginosus	Streptococcus anginosus
1021 Streptococcus bovis	Streptococcus bovis
1045 Streptococcus constellatus	Streptococcus constellatus
1161 Streptococcus gordonii	Streptococcus gordonii
1025 Streptococcus intermedius	Streptococcus intermedius
445 Streptococcus mitis	Streptococcus mitis

further data see next page

Table D.9: Continued

NAME	PATHOGENNAME
447 Streptococcus mutans	Streptococcus mutans
448 Streptococcus oralis	Streptococcus oralis
451 Streptococcus pneumoniae	Streptococcus pneumoniae
454 Streptococcus salivarius	Streptococcus salivarius
455 Streptococcus sanguis	Streptococcus sanguis
384 Trichosporon asahii	Trichosporon asahii

D.10 Application form

Table D.10 specifies various forms of application of antibiotics.

```
CREATE TABLE ApplicationForm (
    name varchar(100),
    isParenteral char(1) CONSTRAINT par_constraint check(isParenteral in('Y','N','U')),
    Primary key(name)
);
```

It is important to know whether the application form is parenteral or not. These application forms are allowed only in more severe cases. Some substances are administered only parenterally. This knowledge is not yet used in the application. Whether an application form is parenteral is specified by the first letter of the following three values: yes, no, or unknown.

Table D.10: Knowledge database table "Applicationform"

NAME	ISPARENTERAL
Oral	N
I.v.	Y
I.m.	Y
Local	N
Topical	N
Infusion	Y
Intrathecal	Y
Rectal	N
Vaginal	Y
Shortinfusion	Y
Intraperitoneal	Y
Eyedrops	Y

D.11 Allergy restriction

Table D.11 specifies which allergy type restricts the use of a substance.

```
CREATE TABLE AllergyRestriction(
    allergyName varchar(200),
    substanceName varchar(100),
    advice varchar(200),
    isAllowed char(1) constraint isallowed_cons check(isAllowed in ('Y','N','U','I')),
    Foreign key(allergyName) references Allergy(name),
    foreign key(substanceName) references Substance(name)
);
```

Whether the use of a substance in case of an allergy is allowed or not is specified by the first letter of the following four values: yes, no, unknown, or inform. The value "inform" allows the use of a substance despite allergy, however, in case of an allergy there is additional information available.

Table D.11: Knowledge database table "Allergyrestriction"

ALLERGYNAME	SUBSTANCENAME	ADVICE	ISALLOWED
Ceftazidim	Aztreonam	CR("frequent")	I
Ceftazidim	Ceftazidim	None	N
Cephalosporin	Cefalotin	None	N
Cephalosporin	Cefpodoxim/Proxetil	None	N
Cephalosporin	Cefamandol	None	N
Cephalosporin	Cefepim	None	N
Cephalosporin	Cefotaxim	None	N
Cephalosporin	Cefadroxil	None	N
Cephalosporin	Cefalexin	None	N
Cephalosporin	Cefaloridin	None	N
Cephalosporin	Cefixim	None	N
Cephalosporin	Cefodizim	None	N
Cephalosporin	Cefotiam	None	N
Cephalosporin	Cefoperazon	None	N
Cephalosporin	Cefpirom	None	N
Cephalosporin	Cefazolin	None	N
Cephalosporin	Cefoxitin	None	N
Cephalosporin	Latamoxef	None	N
Cephalosporin	Cefuroxim/Axetil	None	N
Cephalosporin	Aztreonam	CR("rare")	I
Cephalosporin	Imipenem/Cilastatin	CR("low")	I
Cephalosporin	Pivmecillinam	None	N
Cephalosporin	Ceftizoxim	None	N
Cephalosporin	Cefaclor	None	N
Cephalosporin	Ceftazidim	None	N
Cephalosporin	Ceftriaxon	None	N

further data see next page

Table D.11: Continued

ALLERGYNAME	SUBSTANCENAME	ADVICE	ISALLOWED
Cephalosporin	Cefuroxim	None	N
Macrolide	Erythromycin	None	N
Macrolide	Josamycin	None	N
Macrolide	Clarithromycin	None	N
Macrolide	Spiramycin	None	N
Macrolide	Roxithromycin	None	N
Macrolide	Azithromycin	None	N
Penicillin	Ticarcillin	None	N
Penicillin	Piperacillin	None	N
Penicillin	Piperacillin/Tazobactam	None	N
Penicillin	Temocillin	None	N
Penicillin	Penicillin G	None	N
Penicillin	Penicillin V	None	N
Penicillin	Phenoxymethylpenicillin / Kalium	None	N
Penicillin	Mezlocillin	None	N
Penicillin	Aminobenzylpenicillin	None	N
Penicillin	Amoxicillin	None	N
Penicillin	Amoxicillin/Clavulanate acid	None	N
Penicillin	Imipenem/Cilastatin	CR("50%")	I
Penicillin	Aminobenzylpenicillin + BLI	None	N
Penicillin	Ampicillin	None	N
Penicillin	Ampicillin/Sulbactam	None	N
Penicillin	Pivmecillinam	None	N
Penicillin	Benzylpenicillin	None	N
Penicillin	Cefalotin	CR("1.7-5.6%")	I
Penicillin	Cefpirom	None	N
Penicillin	Cefaclor	CR("1.7-5.6%")	I
Penicillin	Cefpodoxim/Proxetil	CR("1.7-5.6%")	I
Penicillin	Cefodizim	None	N
Penicillin	Cefotaxim	CR("1.7-5.6%")	I
Penicillin	Cefamandol	CR("1.7-5.6%")	I
Penicillin	Cefazolin	CR("1.7-5.6%")	I
Penicillin	Cefuroxim	CR("1.7-5.6%")	I
Penicillin	Azlocillin	None	N
Penicillin	Flucloxacillin	None	N
Penicillin	Cefuroxim/Axetil	CR("1.7-5.6%")	I
Penicillin	Aztreonam	CR("rare")	I
Penicillin	Ceftazidim	CR("1.7-5.6%")	I
Penicillin	Oxacillin	None	N
Penicillin	Bacampicillin	None	N

D.12 Contraindication

Contraindications are sex-specific (e.g., pregnancy). Therefore in this table there is a column for the value sex, which may contain the first letters of the following three values: female, male, or unknown.

```
CREATE TABLE Contraindication (
    name varchar(200),
    sex char(1) CONSTRAINT sex_constraint check(sex in('F','M','U')),
    PRIMARY KEY (name)
);
```

Table D.12: Knowledge database table "Contraindication"

NAME	SEX
Pregnancy	F
Penetration placenta	F
Penetration breast milk	F
Liver dysfunction	U
Renal dysfunction	U
Liver function inhibited	U
Renal impairment unilateral	U
Renal impairment ambilateral	U
Neurologic disorder	U
Diabetes	U

D.13 Contraindication restriction

The contraindication restriction contains analogously to the allergy restriction information about which substance is allowed and why for a certain contraindication. The column "isAllowed" may contain first letters of the following three values: yes, no, or unknown. Not all information contained in this table is evaluated by the application yet.

```
CREATE TABLE ContraindicationRestriction (
   ciName varchar(200),
   substanceName varchar(100),
   advice varchar(200),
   restriction varchar(100),
   isAllowed char(1) constraint allowed-cons check(isAllowed in ('Y','N','U')),
   Foreign key(ciName) references Contraindication(name),
   foreign key(substanceName) references Substance(name)
);
```

Table D.13: Knowledge database table "Contraindicationrestriction"

CINAME	SUBSTANCENAME	ADVICE	RESTRICTION	ISALLOWED
Liver dysfunction	Chloramphenicol	Severe liver impairment	absolut	N
Liver dysfunction	Doxycyclin	Hepatoxicity	absolut	N
Liver dysfunction	Erythromycin	Liver impairment	absolut	N
Liver dysfunction	Minocyclin	Hepatoxicity	absolut	N
Liver dysfunction	Tigecyclin	Hepatoxicity	absolut	N
Penetration breast milk	Amikacin	Y	None	U
Penetration breast milk	Amoxicillin	Y	None	U
Penetration breast milk	Amoxicillin/Clavulanate acid	Y	None	U
Penetration breast milk	Aztreonam	Y	None	U
Penetration breast milk	Cefaclor	Y	None	U
Penetration breast milk	Cefamandol	Y	None	U
Penetration breast milk	Cefazolin	Y	None	U
Penetration breast milk	Cefepim	Y	None	U
Penetration breast milk	Cefotaxim	Y	None	U
Penetration breast milk	Cefoxitin	Y	None	U
Penetration breast milk	Cefpodoxim/Proxetil	Y	None	U
Penetration breast milk	Ceftazidim	Y	None	U
Penetration breast milk	Cefuroxim	Y	None	U
Penetration breast milk	Cefuroxim/Axetil	Y	None	U
Penetration breast milk	Chloramphenicol	Y	None	U
Penetration breast milk	Clarithromycin	Y	None	U
Penetration breast milk	Clindamycin	Y	None	U
Penetration breast milk	Cotrimoxazol	Y	None	U
Penetration breast milk	Doxycyclin	Y	None	U
Penetration breast milk	Ertapenem	Y	None	U
Penetration breast milk	Fosfomycin	Y	None	U
Penetration breast milk	FusidicAcid	Y	None	U
Penetration breast milk	Gentamicin	Y	None	U
Penetration breast milk	Imipenem/Cilastatin	U	None	U
Penetration breast milk	Josamycin	Y	None	U
Penetration breast milk	Meropenem	Y	None	U
Penetration breast milk	Metronidazol	Y	None	U
Penetration breast milk	Minocyclin	Y	None	U
Penetration breast milk	Moxifloxacin	Y	None	U
Penetration breast milk	Oxacillin	Y	None	U
Penetration breast milk	Penicillin G	Y	None	U
Penetration breast milk	Piperacillin	Y	None	U
Penetration breast milk	Pivmecillinam	N	None	U
Penetration breast milk	Quinupristin/Dalfopristin	U	None	U
Penetration breast milk	Rifampicin	Y	None	U
Penetration breast milk	Teicoplanin	U	None	U
Penetration breast milk	Tobramycin	Y	None	U
Penetration breast milk	Trimethoprim	Y	None	U
Penetration breast milk	Vancomycin	Y	None	U

further data see next page

Table D.13: Continued

CINAME	SUBSTANCENAME	ADVICE	RESTRICTION	ISALLOWED
Penetration placenta	Amikacin	+	None	U
Penetration placenta	Amoxicillin	++	None	U
Penetration placenta	Aztreonam	+	None	U
Penetration placenta	Cefazolin	+++	None	U
Penetration placenta	Cefepim	++	None	U
Penetration placenta	Cefotaxim	+	None	U
Penetration placenta	Cefoxitin	+++	None	U
Penetration placenta	Ceftazidim	++	None	U
Penetration placenta	Cefuroxim	+	None	U
Penetration placenta	Cefuroxim/Axetil	+	None	U
Penetration placenta	Chloramphenicol	+++	None	U
Penetration placenta	Ciprofloxacin	+++	None	U
Penetration placenta	Clindamycin	++	None	U
Penetration placenta	Cotrimoxazol	+++	None	U
Penetration placenta	Doxycyclin	++	None	U
Penetration placenta	Fosfomycin	++	None	U
Penetration placenta	FusidicAcid	+	None	U
Penetration placenta	Gentamicin	++	None	U
Penetration placenta	Imipenem/Cilastatin	+++	None	U
Penetration placenta	Levofloxacin	+++	None	U
Penetration placenta	Metronidazol	+++	None	U
Penetration placenta	Netilmicin	+	None	U
Penetration placenta	Oxacillin	+	None	U
Penetration placenta	Penicillin G	+++	None	U
Penetration placenta	Piperacillin	+	None	U
Penetration placenta	Rifampicin	++	None	U
Penetration placenta	Tobramycin	+++	None	U
Penetration placenta	Trimethoprim	+++	None	U
Penetration placenta	Vancomycin	+++	None	U
Pregnancy	Amikacin	D (FDA)	None	U
Pregnancy	Aminobenzylpenicillin	YES (FDA)	None	Y
Pregnancy	Aminobenzylpenicillin + BLI	YES (FDA)	None	Y
Pregnancy	Amoxicillin	B (FDA)	None	U
Pregnancy	Amoxicillin/Clavulanate acid	B (FDA)	None	U
Pregnancy	Ampicillin/Sulbactam	C (FDA)	None	U
Pregnancy	Aztreonam	B (FDA)	None	U
Pregnancy	Cefaclor	B (FDA)	None	U
Pregnancy	Cefamandol	B (FDA)	None	U
Pregnancy	Cefepim	B (FDA)	None	U
Pregnancy	Cefodizim	NO	absolut	N
Pregnancy	Cefotaxim	B (FDA)	None	U
Pregnancy	Cefoxitin	B (FDA)	None	U
Pregnancy	Cefpirom	B (FDA)	None	U
Pregnancy	Cefpodoxim/Proxetil	B (FDA)	None	U
Pregnancy	Ceftazidim	B (FDA)	None	U
Pregnancy	Cefuroxim	B (FDA)	None	U
Pregnancy	Cefuroxim/Axetil	B (FDA)	None	U

further data see next page

Table D.13: Continued

CINAME	SUBSTANCENAME	ADVICE	RESTRICTION	ISALLOWED
Pregnancy	Chloramphenicol	C (FDA)	None	U
Pregnancy	Ciprofloxacin	C (FDA)	None	U
Pregnancy	Clarithromycin	C (FDA)	None	U
Pregnancy	Clindamycin	B (FDA)	None	U
Pregnancy	Cotrimoxazol	C (FDA)	None	U
Pregnancy	Doxycyclin	D (FDA)	None	U
Pregnancy	Ertapenem	B (FDA)	None	U
Pregnancy	Fosfomycin	B (FDA)	None	U
Pregnancy	FusidicAcid	C (FDA)	None	U

D.14 Diagnosis

Table D.14 contains all possible diagnoses handled by the application. Until now only UTI-specific diagnoses are listed. The column "ischronic" may contain the first letter of the following three values: yes, no, or unknown.

```
CREATE TABLE Diagnosis(
   diagName varchar(100),
   ischronic char(1) constraint chron check(chronic in ('Y','N','U')),
   Primary key(diagName)
);
```

Table D.14: Knowledge database table "Diagnosis"

DIAGNAME	ISCHRONIC
Acute Cystitis	N
Acute Pyelonephritis	N
Acute Pyelonephritis and complication	N
Acute Pyelonephritis and pregnant	N
Chronic Cystitis	Y
Chronic Pyelonephritis	Y

D.15 Diagnosis treatment

Table D.15 links the tables "antibiotic" (group, subgroup, or substance) and "diagnosis" and is extended by two columns containing additional information. The column ranking specifies the ranking of the substance for the specified diagnosis mentioned in literature—it gives the order, in which this substance is mentioned as treatment for the diagnosis. The column reserve specifies reserve substances, substances to use only if nothing else works, for a given diagnosis. The first letter of the following three values may be contained in column reserve: yes, no, or unknown. The information contained in the ranking and reserve column is not shown on the output form yet.

```
CREATE TABLE DiagnosisTreatment(
   diagName varchar(100),
   substanceName varchar(100),
   subgroupName varchar(100),
   groupName varchar(100),
   ranking varchar(3),
   reserve char(1) constraint reservation check(reserve in ('Y','N','U')),
   FOREIGN KEY(substanceName) REFERENCES Substance(name),
   FOREIGN KEY(diagName) REFERENCES Diagnosis(diagName),
   FOREIGN KEY(subgroupName) REFERENCES ABSubgroup(name),
   FOREIGN KEY(groupName) REFERENCES ABGroup(name)
);
```

Table D.15: Knowledge database table "Diagnosistreatment"

DIAGNAME	SUBSTANCENAME	SUBGROUPNAME	GROUPNAME	RANKING	RESERVE
Chronic Pyelonephritis	None	U	GyraseInhibitor	1	N
Chronic Pyelonephritis	Cotrimoxazol	U	Unspecified	2	N
Acute Pyelonephritis	Cotrimoxazol	U	Unspecified	1	N
Acute Pyelonephritis	Cefixim	U	Unspecified	2	N
Acute Pyelonephritis	Cefpodoxim/Proxetil	U	Unspecified	3	N
Acute Pyelonephritis	Ciprofloxacin	U	Unspecified	4	N
Chronic Cystitis	None	U	Unspecified	0	U
Acute Cystitis	Ciprofloxacin	U	Unspecified	1	N
Acute Cystitis	Levofloxacin	U	Unspecified	2	N
Acute Cystitis	Cotrimoxazol	U	Unspecified	3	N
Acute Cystitis	None	Oralcephalosporine 1. Generation	Unspecified	1	Y
Acute Cystitis	None	Oralcephalosporine 2. Generation	Unspecified	1	Y
Acute Cystitis	None	Oralcephalosporine 3. Generation	Unspecified	1	Y
Acute Cystitis	Amoxicillin	U	Unspecified	2	Y
Acute Cystitis	Fosfomycin	U	Unspecified	3	Y
Acute Cystitis	Ceftriaxon	U	Unspecified	4	Y
Acute Cystitis	None	U	Aminoglycoside	5	Y
Acute Pyelonephritis and pregnant	Amoxicillin	U	Unspecified	1	N
Acute Pyelonephritis and pregnant	None	Oralcephalosporine 1. Generation	Unspecified	2	N
Acute Pyelonephritis and pregnant	None	Oralcephalosporine 2. Generation	Unspecified	2	N
Acute Pyelonephritis and pregnant	None	Oralcephalosporine 3. Generation	Unspecified	2	N
Acute Pyelonephritis and pregnant	Cefuroxim	U	Unspecified	3	N
Acute Pyelonephritis and pregnant	Ceftriaxon	U	Unspecified	4	N
Acute Pyelonephritis and pregnant	Mezlocillin	U	Unspecified	5	N
Acute Pyelonephritis and complication	None	U	Cephalosporine	1	N
Acute Pyelonephritis and complication	Piperacillin/Tazobactam	U	Unspecified	2	N
Acute Pyelonephritis and complication	Ciprofloxacin	U	Unspecified	3	N
Acute Pyelonephritis and complication	Levofloxacin	U	Unspecified	4	N
Acute Pyelonephritis and complication	None	U	Carbapeneme	1	Y

D.16 Pathogen treatment

The knowledge obtained in table D.16 is analogous to the diagnosis treatment table. For a pathogen group or a pathogen a treatment is defined. Treatment may be either an antibiotic group, subgroup, or substance. The column effectiveness defines how effective the given antibiotic (group, subgroup, or substance) treats the pathogen (group, or pathogen). It may contain the first letter of the following four values: yes, no, moderate, or unknown.

```
CREATE TABLE PathogenTreatment (
   pathogenName varchar(200),
   pgroupName varchar(100),
   substanceName varchar(100),
   groupName varchar(100),
   subgroupName varchar(100),
   effectiveness char(1) constraint effective ´
    check(effectiveness in (’Y’,’M’,’N’,’U’)),
   FOREIGN KEY(pathogenName) REFERENCES Pathogen(name),
   foreign key(pgroupName) references PathogenGroup(name_en),
   FOREIGN KEY(substanceName) REFERENCES Substance(name),
   FOREIGN KEY(subgroupName) REFERENCES ABSubgroup(name),
   FOREIGN KEY(groupName) REFERENCES ABGroup(name)
);
```

Table D.16: Knowledge database table "Pathogentreatment"

PATHOGENNAME	PGROUPNAME	SUBSTANCENAME	GROUPNAME	SUBGROUPNAME	EFFECTIVENESS
Escherichia coli	Unspecified	Metronidazol	Unspecified	U	N
Escherichia coli	Unspecified	Amoxicillin/Clavulanate acid	Unspecified	U	Y
Escherichia coli	Unspecified	Moxifloxacin	Unspecified	U	Y
Escherichia coli	Unspecified	Aztreonam	Unspecified	U	Y
Escherichia coli	Unspecified	Cotrimoxazol	Unspecified	U	Y
Escherichia coli	Unspecified	Mezlocillin	Unspecified	U	M
Escherichia coli	Unspecified	Amikacin	Unspecified	U	Y
Escherichia coli	Unspecified	Cefpodoxim/Proxetil	Unspecified	U	Y
Escherichia coli	Unspecified	Cefaclor	Unspecified	U	M
Escherichia coli	Unspecified	Cefpirom	Unspecified	U	Y
Escherichia coli	Unspecified	Netilmicin	Unspecified	U	Y
Escherichia coli	Unspecified	Ciprofloxacin	Unspecified	U	Y
Escherichia coli	Unspecified	Cefotaxim	Unspecified	U	Y
Escherichia coli	Unspecified	Daptomycin	Unspecified	U	N
Escherichia coli	Unspecified	Cefuroxim	Unspecified	U	Y
Escherichia coli	Unspecified	Clindamycin	Unspecified	U	N
Escherichia coli	Unspecified	Cefadroxil	Unspecified	U	M
Escherichia coli	Unspecified	Flucloxacillin	Unspecified	U	N
Escherichia coli	Unspecified	Ceftazidim	Unspecified	U	Y
Escherichia coli	Unspecified	Fosfomycin	Unspecified	U	Y
Escherichia coli	Unspecified	FusidicAcid	Unspecified	U	N
Escherichia coli	Unspecified	Chloramphenicol	Unspecified	U	M
Escherichia coli	Unspecified	Ertapenem	Unspecified	U	Y
Escherichia coli	Unspecified	Josamycin	Unspecified	U	N
Escherichia coli	Unspecified	Cefazolin	Unspecified	U	Y
Escherichia coli	Unspecified	Telithromycin	Unspecified	U	N
Escherichia coli	Unspecified	Clarithromycin	Unspecified	U	N
Escherichia coli	Unspecified	Cefamandol	Unspecified	U	Y
Escherichia coli	Unspecified	Cefepim	Unspecified	U	Y
Escherichia coli	Unspecified	Cefoxitin	Unspecified	U	Y
Escherichia coli	Unspecified	Minocyclin	Unspecified	U	Y
Escherichia coli	Unspecified	Trimethoprim	Unspecified	U	Y

further data see next page

Table D.16: Continued

PATHOGENNAME	PGROUPNAME	SUBSTANCENAME	GROUPNAME	SUBGROUPNAME	EFFECTIVENESS
Escherichia coli	Unspecified	Meropenem	Unspecified	U	Y
Escherichia coli	Unspecified	Penicillin V	Unspecified	U	N
Escherichia coli	Unspecified	Cefalexin	Unspecified	U	M
Escherichia coli	Unspecified	Bacampicillin	Unspecified	U	M
Escherichia coli	Unspecified	Penicillin G	Unspecified	U	N
Escherichia coli	Unspecified	Piperacillin	Unspecified	U	M
Escherichia coli	Unspecified	Gentamicin	Unspecified	U	Y
Escherichia coli	Unspecified	Rifampicin	Unspecified	U	N
Escherichia coli	Unspecified	Ceftriaxon	Unspecified	U	Y
Escherichia coli	Unspecified	Roxithromycin	Unspecified	U	N
Escherichia coli	Unspecified	Pivmecillinam	Unspecified	U	Y
Escherichia coli	Unspecified	Cefotiam	Unspecified	U	Y
Escherichia coli	Unspecified	Ampicillin	Unspecified	U	M
Escherichia coli	Unspecified	Oxacillin	Unspecified	U	N
Escherichia coli	Unspecified	Doxycyclin	Unspecified	U	Y
Escherichia coli	Unspecified	Amoxicillin	Unspecified	U	M
Escherichia coli	Unspecified	Quinupristin/Dalfopristin	Unspecified	U	N
Escherichia coli	Unspecified	Teicoplanin	Unspecified	U	N
Escherichia coli	Unspecified	Levofloxacin	Unspecified	U	Y
Escherichia coli	Unspecified	Piperacillin/Tazobactam	Unspecified	U	Y
Escherichia coli	Unspecified	Tigecyclin	Unspecified	U	Y
Escherichia coli	Unspecified	Cefodizim	Unspecified	U	Y
Escherichia coli	Unspecified	Tobramycin	Unspecified	U	Y
Escherichia coli	Unspecified	Cefixim	Unspecified	U	Y
Escherichia coli	Unspecified	Ampicillin/Sulbactam	Unspecified	U	Y
Escherichia coli	Unspecified	Vancomycin	Unspecified	U	N
Escherichia coli	Unspecified	Imipenem/Cilastatin	Unspecified	U	Y
Escherichia coli	Unspecified	Azithromycin	Unspecified	U	N
Escherichia coli	Unspecified	Norfloxacin	Unspecified	U	Y
Escherichia coli	Unspecified	Linezolid	Unspecified	U	N
Enterobacter cloacae	Unspecified	Metronidazol	Unspecified	U	N
Enterobacter cloacae	Unspecified	Amoxicillin/Clavulanate acid	Unspecified	U	N

further data see next page

Table D.16: Continued

PATHOGENNAME	PGROUPNAME	SUBSTANCENAME	GROUPNAME	SUBGROUPNAME	EFFECTIVENESS
Enterobacter cloacae	Unspecified	Moxifloxacin	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Aztreonam	Unspecified	U	M
Enterobacter cloacae	Unspecified	Cotrimoxazol	Unspecified	U	M
Enterobacter cloacae	Unspecified	Mezlocillin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Amikacin	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Cefpodoxim/Proxetil	Unspecified	U	M
Enterobacter cloacae	Unspecified	Cefaclor	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefpirom	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Netilmicin	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Ciprofloxacin	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Cefotaxim	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Daptomycin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefuroxim	Unspecified	U	N
Enterobacter cloacae	Unspecified	Clindamycin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefadroxil	Unspecified	U	N
Enterobacter cloacae	Unspecified	Flucloxacillin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Ceftazidim	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Fosfomycin	Unspecified	U	M
Enterobacter cloacae	Unspecified	FusidicAcid	Unspecified	U	N
Enterobacter cloacae	Unspecified	Chloramphenicol	Unspecified	U	M
Enterobacter cloacae	Unspecified	Ertapenem	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Josamycin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefazolin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Telithromycin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Clarithromycin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefamandol	Unspecified	U	N
Enterobacter cloacae	Unspecified	Cefepim	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Cefoxitin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Minocyclin	Unspecified	U	M
Enterobacter cloacae	Unspecified	Trimethoprim	Unspecified	U	M
Enterobacter cloacae	Unspecified	Meropenem	Unspecified	U	Y
Enterobacter cloacae	Unspecified	Penicillin V	Unspecified	U	N

further data see next page

Table D.16: Continued

PATHOGENNAME	**PGROUPNAME**	**SUBSTANCENAME**	**GROUPNAME**	**SUBGROUPNAME**	**EFFECTIVENESS**
Enterobacter cloacae	Unspecified	Cefalexin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Bacampicillin	Unspecified	U	N
Enterobacter cloacae	Unspecified	Penicillin G	Unspecified	U	N
Enterobacter cloacae	Unspecified	Piperacillin	Unspecified	U	N

Appendix E: Rules

In this appendix all rules are listed, which are used in the application. In the first part of this appendix the two main rules, first pathogen and quotient rule, are listed. The second part contains rules, which are represented by the decision trees shown in Appendix A.

E.1 First pathogen and quotient rule

A description for the use of these rules can be found in section 3.5.

```
<?xml version="1.0" encoding="utf-8" ?>
<template>

<!--expected pathogen spectrum variables-->
<var name="firstPathogen" type="real"/>
<var name="secondPathogen" type="real"/>
<var name="quotient" type="real"/>
<var name="firstBorder" type="real"/>
<var name="firstWidth" type="real"/>
<var name="firstFunc" type="real"/>
<var name="quotBorder" type="real"/>
<var name="quotWidth" type="real"/>
<var name="quotFunc" type="real"/>
<var name="fuzzyGreater" type="real"/>
<var name="fuzzyQuotient" type="real"/>´


<!--expected pathogen spectrum rules-->
<fuzzy name="fGreater">
<function var="firstFunc"/>
<border var="firstBorder"/>
<width var="firstWidth"/>
</fuzzy>

<rule>
<in op="fGreater" var="firstPathogen"/>
<out var="fuzzyGreater"/>
</rule>
```

```
<fuzzy name="fQuotient" invert="true">
<function var="quotFunc"/>
<border var="quotBorder"/>
<width var="quotWidth"/>
</fuzzy>

<rule>
<in op="fQuotient" var="quotient"/>
<out var="fuzzyQuotient"/>
</rule>
```

E.2 Application rules

The use of these rules is described in section 3.6.

```
<!--duration variables-->
<var name="minDays" type="real"/>
<var name="maxDays" type="real"/>
<var name="singleDoseTherapy" type="boolean"/>
<var name="longTherapy" type="boolean"/>

<!--basic variables-->
<var name="female" type="boolean"/>
<var name="male" type="boolean"/>
<var name="age" type="real"/>
<var name="urine" type="boolean"/>
<var name="young" type="boolean"/>
<var name="old" type="boolean"/>
<var name="emergencyStation" type="boolean"/>

<!--contraindication variables-->
<var name="pregnant" type="boolean"/>
<var name="catheter" type="boolean"/>
<var name="urologicalOP" type="boolean"/>
<var name="diabetes" type="boolean"/>
<var name="renal" type="boolean"/>
<var name="liver" type="boolean"/>

<!--diagnosis variables-->
<var name="symptHematuria" type="boolean"/>
<var name="symptPain" type="boolean"/>
<var name="symptFlankPain" type="boolean"/>
<var name="symptFever" type="boolean"/>
<var name="pyelonephritis" type="boolean"/>
<var name="cystitis" type="boolean"/>
<var name="recurrentUTI" type="boolean"/>
<var name="chronic" type="boolean"/>
```

```
<var name="acute" type="boolean"/>

<!--complication variables-->
<var name="complication" type="boolean"/>
<var name="obstruction" type="boolean"/>
<var name="dosis" type="boolean"/>

<!--recommendation variables-->
<var name="resistance" type="boolean"/>

<!--dosis variables-->
<var name="oral" type="boolean"/>
<var name="parenteral" type="boolean"/>

<!--basic rules-->
<var name="oldBorder" type="real"/>
<var name="oldFunction" type="real"/>
<var name="oldWidth" type="real"/>
<fuzzy name="alt">
<function var="oldFunction"/>
<border var="oldBorder"/>
<width var="oldWidth"/>
</fuzzy>

<rule>
<in var="pregnant" />
<out var="female"/>
</rule>

<rule>
<in var="male"/>
<out op="not" var="pregnant"/>
</rule>

<rule>
<in op="alt" var="age" />
<out var="old" />
</rule>

<rule>
<in op="not" var="old" />
<out var="young" />
</rule>

<!--dosis rules-->
<rule>
<in var="emergencyStation"/>
<in var="young"/>
<out var="oral"/>
</rule>
```

```
<rule>
<in var="oral"/>
<out op="not" var="parenteral"/>
</rule>

<rule>
<in op="not" var="oral"/>
<out var="parenteral"/>
</rule>

<!--diagnosis rules-->
<rule>
<in var="symptFlankPain"/>
<in var="symptFever"/>
<out var="pyelonephritis"/>
</rule>

<rule>
<in var="symptHematuria"/>
<in var="symptPain"/>
<in op="not" var="symptFlankPain"/>
<in op="not" var="symptFever"/>
<out var="cystitis"/>
</rule>

<rule>
<in var="recurrentUTI"/>
<out var="chronic"/>
</rule>

<rule>
<in op="not" var="recurrentUTI"/>
<out var="acute"/>
</rule>

<rule>
<in op="not" var="chronic"/>
<out var="acute"/>
</rule>

<!--complication rules-->
<rule>
<in var="chronic"/>
<out var="complication"/>
</rule>

<rule>
<in var="pyelonephritis"/>
<out var="complication"/>
</rule>
```

```
<rule>
<in var="pregnant"/>
<out var="obstruction"/>
</rule>

<rule>
<in var="catheter"/>
<out var="obstruction"/>
</rule>

<rule>
<in var="male"/>
<out var="complication"/>
</rule>

<rule>
<in var="obstruction"/>
<out var="complication"/>
</rule>

<rule>
<in var="urologicalOP"/>
<out var="complication"/>
</rule>

<rule>
<in var="diabetes"/>
<out var="complication"/>
<out var="dosis"/>
</rule>

<rule>
<in var="renal"/>
<out var="complication"/>
<out var="dosis"/>
</rule>

<rule>
<in var="liver"/>
<out var="complication"/>
<out var="dosis"/>
</rule>

<!--duration rules-->
<rule>
<in op="not" var="complication"/>
<in var="female"/>
<in var="old"/>
<out var="minDays" value="3.0"/>
<out var="maxDays" value="5.0"/>
</rule>
```

```
<rule>
<in op="not" var="complication"/>
<in var="female"/>
<in var="young"/>
<out var="minDays" value="1.0"/>
<out var="maxDays" value="1.0"/>
<out var="singleDoseTherapy"/>
</rule>



<rule>
<in var="complication"/>
<in var="female"/>
<out var="minDays" value="3.0"/>
<out var="maxDays" value="5.0"/>
</rule>

<rule>
<in var="complication"/>
<in var="male"/>
<out var="minDays" value="20.0"/>
<out var="maxDays" value="60.0"/>
</rule>

</template>
```