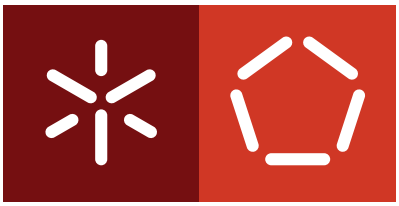
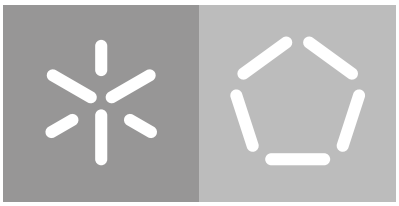


**Universidade do Minho**
Escola de Engenharia
Departamento de Informática

Pedro José Costa de Oliveira

# Conflict Resolution in Clinical Treatments

December 2019



**Universidade do Minho**
Escola de Engenharia
Departamento de Informática

Pedro José Costa de Oliveira

# Conflict Resolution in Clinical Treatments

Masters Dissertation
Integrated Master's Degree in Informatics Engineering

Dissertation supervised by
**Paulo Jorge Freitas de Oliveira Novais**

December 2019

# ACKNOWLEDGEMENTS

It was a long road and many obstacles to reach this moment. During these university years, there were good and bad times, which served as long learning and which I will take with me throughout my life. I keep every moment in my heart.

In this house of knowledge, I learned values that made me become the person I am today, prepared to face the adversities that life will face in front of me, both professionally and personally.

This whole process was only possible thanks to the support of my parents, who were always able to complete this step. I will never have words or gestures that come to thank them for all the sacrifices so that I could get here. I also want to thank my grandmother and my brother for their always transmitted presence and strength. It is a word to all my family members who are no longer physically by my side, but it was also for them that I had the strength to achieve this goal.

A very special thank you to Professor Paulo Novais and António Silva for all the availability and patience throughout this dissertation. Without your guidance, everything would have been more complicated, both in the writing of this document and the elaboration of the proposal to the problem of this dissertation. Thanks also to all the teachers that over the years I had the opportunity to live with and with whom I learned a lot.

A very special thanks to my girlfriend Patrícia Cunha for all the unconditional support throughout this dissertation. She was the one who encouraged me on the bad days this year, giving me strength and optimism that everything would go well. Without her, all these moments would have been harder to overcome.

To all my childhood friends who have always accompanied me throughout my life, especially Pedro Fonseca, a huge thank you for all the trust you have placed in me and all the support provided.

To the friends that the University of Minho gave me, especially to Daniel Cruz and Miguel Silva, thank you for all the availability and support throughout these years and during the development of this dissertation.

Finally, I would like to thank all the people who have spent the years in my life, from kindergarten teachers to high school teachers. All of them, in one way or another, contributed to achieving this goal.

## STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

## ABSTRACT

Currently, in the health area, there is a need for systems that provide support for the decision of health professionals through specific recommendations for each patient based on Clinical Practice Guidelines (CPGs) for automatic interpretation. CPGs are documents that have enormous importance in the daily life of health professionals, playing a key role in reducing variations in medical practice, improving the quality of health care, and reducing health care costs. These documents reflect knowledge about how best to diagnose and treat diseases in the form of a list of clinical recommendations.

However, there may be conflicts and interactions in the application of these clinical recommendations, that which in their maximum exponent may impair the patient's clinical condition. These conflicts are transported to decision support systems, creating the need to develop computational methods to solve these same conflicts. In the case of multimorbid patients, this resolution of conflicts can be very problematic because these patients suffer from several pathologies at the same time, and that the use of a drug for one particular pathology may have a detrimental effect on the application of another drug in another pathology.

Therefore, the objective of this dissertation topic is the determination of conflicts and interactions between drugs and the determination of these same alternatives.



***Keywords*** - Artificial Intelligence in Medicine, Clinical Decision Support System, Clinical Pratice Guidelines, Combining Clinical Pratice Guidelines, Computer-Interpreter Guidelines, Multi Criteria Decision Analysis

## RESUMO

Atualmente na área da saúde, existe uma necessidade de existirem sistemas que forneçam apoio à decisão dos profissionais de saúde através de recomendações específicas para cada paciente com base em protocolos clínicos para interpretação automática. Os protocolos clínicos são documentos que têm enorme importância no dia-a-dia dos profissionais de saúde, desempenhando um papel fundamental na redução das variações na prática médica, na melhoria da qualidade dos cuidados de saúde e na redução dos custos de saúde. Estes documentos reflectem o conhecimento sobre a melhor forma de diagnosticar e tratar doenças na forma de uma lista de recomendações clínicas.

Contudo, podem existir conflitos e interações na aplicação destas recomendações clínicas, que no seu expoente máximo poderão levar a um agravamento do estado clínico do paciente, nomeadamente no caso da aplicação de diferentes fármacos. Estes conflitos são transportados para os sistemas de apoio à decisão, criando a necessidade de desenvolver métodos computacionais de resolução destes mesmos conflitos. No caso dos pacientes multimórbidos esta resolução de conflitos pode ser bastante problemática devido ao facto destes pacientes sofrerem de várias patologias ao mesmo tempo, e que a utilização de um fármaco para uma determinada patologia possa vir a ter um efeito nocivo na aplicação de outro fármaco noutra patologia.

Sendo assim, o objetivo deste tema de dissertação é a determinação dos conflitos e interações entre fármacos e a determinação dessas mesmas alternativas.



***Palavras-Chave*** - Análise de Decisão com Múltiplos Critérios, Combinação de Protocolos Clínicos, Computer-Interpreter Guidelines, Inteligência Artificial em Medicina, Protocolos Clínicos, Sistemas de Apoio à Decisão Clínica

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ACRONYMS

**A**

**AHP** Analytical Hierarchy Process.

**AI** Artificial Intelligence.

**ANN** Artificial Neural Networks.

**API** Application Programming Interface.

**ASTM** American Society for Testing and Materials.

**C**

**CDSS** Clinical Decision Support System.

**CIG** Computer-Interpretable Guidelines.

**CLP** Constraint Logic Programming.

**CPG** Clinical Pratice Guidelines.

**CSS** Cascading Style Sheets.

**CT** Computed Tomography.

**D**

**DCE** Discrete Choice Experiments.

**DKO** Domain Knowledge Ontology.

**DSS** Decision Support Systems.

**G**

**GLARE** Guideline Acquisition, Representatio and Execution.

**GLIF** Guideline Interchange Format.

**H**

**HL7** Health Level 7.

**HTA** Health Technology Assessment.

HTML HyperText Markup Language.

I

IMIE Integrated Master in Informatics Engineering.

K

KMO Knowledge Mapping Ontology.

KTO Knowledge Transformation Ontology.

L

LKO Local Knowledge Ontology.

M

MCDA Multi Criteria Decision Analysis.

MLM Medical Logic Module.

MRI Magnetic Resonance Imaging.

MVC Model-View-Controller.

N

NCCN National Comprehensive Cancer Network.

O

OTC1 Organic Cation Transporter 1.

OWL Ontology Web Language.

R

R2L Red Representation Language.

REST Representational State Transfer.

S

SAGE Standards-Based Sharable Active Guideline Environment.

SDM Shared Decision Making.

SWRL Semantic Web Rule Language.

T

TMN Task-Network Model.

TMR4I Transition-based Medical Recommendations model.

U

UI User Interface.

UK United Kingdom.

UM University of Minho.

UML Unified Modeling Language.

UN United Nations.

X

XML Extensible Markup Language.

# 1

# INTRODUCTION

The present work of the dissertation, developed within the course of Integrated Master in Informatics Engineering (IMIE) at the University of Minho (UM), has a theme: "Conflict Resolution in Clinical Treatments".

This dissertation study covers different Computer Science fields, in particularly Artificial Intelligence (AI) which have a more application-oriented approach, such as CPGs, Computer-Interpretable Guidelines (CIGs), and Clinical Decision Support Systems (CDSSs).

Section 1.1 provides a theoretical background of the developed work, as well as, the motivation that underlies it. Section 1.2 references the main objectives inherent in the development of this dissertation. Lastly, section 1.3 describes the organization of the document and the topics covered in each chapter.

## 1.1 BACKGROUND AND MOTIVATION

The subject of this dissertation involves different areas of great relevance to our society, therefore it becomes relevant to frame them in the present project scope.

One of the most influential areas of this dissertation is *eHealth*, which is a recent term in the practice of health care, dating back at least to the year 1999. *eHealth* is a term that has a very close relationship with computer science, which aims to help improve people's quality of life through improved clinical conditions [2].

Another area of great interest is AI, which is a branch of computer science that proposes to devise methods that attempt to simulate the human capacity to reason, perceive, make decisions and solve problems. Furthermore, AI has an important role in the eHealth technologies sector, as it helps to improve their overall performance through Decision Support Systems (DSS). [3] [4].

CDSSs are specialized systems that assist health professionals in making decisions in the fulfilment of their clinical tasks. For example, diagnosis identification [5]. This type of systems can help to improve health care. Nevertheless, there must be support that represents the medical knowledge and that crosses, automatically, with the condition of the patient. One of these supports are algorithms based on clinical protocols, in a format type which allows their interpretation to be performed automatically.

The following subsections 1.1.1 and 1.1.2, specify some aspects and challenges of the CDSSs, as well as the role that the CPGs have in their development.

### 1.1.1 *Clinical Decision Support System*

Decision Support Systems (DSS) in the last decade have undergone a great evolution, particularly in the transition from theoretical concepts to the computational world. A DSS main goal is to assist users in decision making. The applications of DSS returned over time cover several areas, from safety and transportation to Medicine, as well as others [5]. The combination of a knowledge base with certain rules of inference makes DSS capable of improving user decision-making. Among all the areas of the DSS, the most relevant for this dissertation in the area of Medicine.

The research about the potentiality of the application of artificial intelligence techniques in the branch of Medicine, already reports to the middle of the last century. In these last decades, the exploration of these techniques has been growing in the medical area. Increasingly, clinical problems are more complex, and for that, we have to acquire, analyze and also apply a great deal of knowledge, such as parameters of the patient's condition and clinical conditions. The application of AI in medicine has sought to develop systems capable of assisting health professionals in decision making and diagnoses, for example, help clinician in the formulation of a diagnosis, the making of therapeutic decisions and the prediction of an outcome [6].

CDSSs are computer systems whose goal is to assist health professionals in their decision making, where their largest knowledge base is composed of patient's data [7]. These types of systems represent a measure of prevention against clinical error [8], in a way to improve patient safety [9], such as:

- CDSSs can detect the development of serious conditions more quickly than unassisted observation. By continuously monitoring a patient, CDSSs can detect initial signs of deterioration – such as the slow rise of white blood cell counts revealed in a lab test paired with the beginning of fever and hypotension – and alert to the possibility of sepsis and the immediate need for intervention.

- In case the patient develops dizziness, clinical guidelines may direct physicians to request a Computed Tomography (CT) scan to rule out a stroke. If the patient is having a stroke, the best route would be to order a Magnetic Resonance Imaging (MRI). However, ordering MRI to rule out stroke in all cases, would increase cases of unnecessary medical examination. CDSS may align features by suggesting magnetic resonance imaging only for patients with general clinical indications of increased stroke risk.

Figure 1, describes the two major types of CDSS, Knowledge-based CDSS and Non-Knowledge-based CDSS [5].

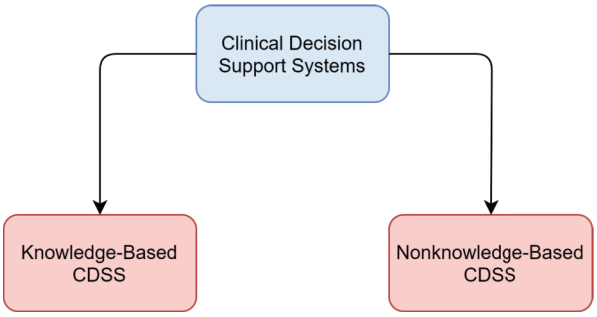



Figure 1.: Types of Clinical Decision Support Systems

In the case of Knowledge-Based CDSS, these type of systems came up to create computational programs that could simulate the cognitive abilities of a human being [10]. A knowledge-based CDSS contains rules, and it mainly comes in the form of *If-Then* statements, with data usually associated with them. This type of CDSS generally consists of three main parts:

- **Knowledge base** - contains the rules;
- **Inference Mechanism** - combines rules with patient data;
- **Communication Mechanism** - show the result to users and to provide information to the system.

The structure of a knowledge-based CDSS is always different because it depends on several factors, such as the source of data and the use of the same.

Unlike the previously described approach, the Nonklowledge-based CDSS, applies a machine learning principle, allowing a program to learn about past experiences and discover patterns in clinical data. An example of using this type of approach is the computational models applied by computer systems, called Artificial Neural Networks (ANN). [11].

Over the years there has been a great deal of research around CDSSs, and the challenges that were imposed were even greater. Some of these challenges are important to consider [12]:

- **Improve the human-computer interface** - it is always necessary to envolve the paradigms of human-computer interface, to present recommendations to support the decision that supports not interrupt the workflow of health professionals, that is, remind these professionals of things that they can neglect and support their corrections;
- **Prioritize and filter recommendations to the user** - a robust, reliable and evidence-based system is always needed. This aspect represents the main challenge, therefore it is relevant to take into account the influences and competing values that impact clinical decision making. An additional challenge is the reduction of the

number of recommendations generated by this type of systems, which a health professional deals with a reduction of ”alert fatigue”;

- **Combine recommendations for patients with multimorbidity** - clinical treatments often ignore the fact that many of the elderly patients suffer from multiple diseases and take various medications. The challenge here is to create mechanisms to identify and eliminate recommendations that are contraindicated, discordant, or mutually exclusive, hence this type of system must present various types of clinical guideline recommendations.

Currently, several CDSSs were developed, that include technologies such as machine learning, ontologies and decision trees. In the past, several research approaches focused on developing systems capable of providing support for the clinical decision, such as probabilistic and data-based classification, being used in diagnostics, evidence-based medicine, technology evaluation, etc [13]. Some of these surveys, which serve as an example of CDSS, are presented below.

*Zynx Health* is a CDSS developed by the *Hearst Corporation*, which helps hospitals improve patient outcomes and clinical monitoring. The evidence-based tools in this system, provide information to health professionals and workflow suggestions, encouraging collaboration between all parties to improve clinical outcomes.

The Cerner system, owned by *Cerner Corporation*, uses a set of evidence-based standards and criteria to provide healthcare professionals with reliable guidance to ensure that patients receive the most appropriate treatment for their needs. This system also supports clinical decisions for a diverse range of health services, such as in the field of radiology. Also, health professionals are provided with information on the clinical workflow to allow more precise prescriptions, to improve patient care [14].

PERFEX is a system that supports clinical decision-making, rather than supporting health professionals in perfusion assessment problems. This is a rule-based system which contains more than 250 rules, each of which serves for automatic interpretation of SPECT cardiac data. Moreover, it infers the extent and severity of coronary artery disease, where his purposes are to aid in the diagnosis of this same disease [15].

PUFF diagnoses and fills a lung disease. This system requires linear sharing with health professionals. The knowledge base present in this type of system was incorporated into commercial products, such as the case of ”Pulmonary Consultation” [16].

ILIAD system applies Bayesian reasoning as a method for calculating the posterior probabilities of various diagnoses, based on the findings provided in a particular case. This system was developed by Apple Mac, and when it was created it’s with the main objective to provide a diagnosis in the Internal Medicine, where nowadays it already covers several diagnoses [17].

In this dissertation, the main focus relies on the CDSSs that provide decision support based on clinical protocol versions for automatic interpretation, as well as systems that are capable of resolving the conflicts and interactions in clinical treatments.

### 1.1.2 *Clinical Pratice Guidelines*

In healthcare facilities, such as Hospitals and Clinics, that have a great diversity of procedures, the use of CPGs is of extreme importance, because health professionals, which are subject to stressful situations, responsible for medical errors, variations in clinical practice, and practice of defensive medicine.

The formalisation of a CPG in versions for automatic interpretation, the CIGs, it makes possible the development of DSSs based on CIGs, which offer a better possibility to affect the clinical behaviour compared to the narrative documents of the corresponding textual versions.

A CPG can act as a guide to assist the healthcare professional. An example of its application can be verified when a healthcare professional needs to review the administration of a given drug to a patient in a given case. Also, CPGs allows the health professional access to the treatment plan, as well as, provides tasks for the monitoring of the patient's health condition [18].

There are some advantages with the use of CPGs, such as the eradication of omissions, since the human can easily omit something important, and with this type of document, such omissions cease to exist. Other advantages are the reduction of confusions among health professionals, due to stress and pressure factors, the possibility of communication failure is greater. These advantages, among others, make the use of these types of protocols very beneficial to the health professionals [19].

Over the years, we have observed an increase in the world population, which according to the United Nations (UN), will continue to grow in the coming years. In Table 1, it is possible to see the level of population growth, compared with the changes between the year 2010 and 2100 [1].

Table 1.: Total Population in 2010 and 2100, The World and Major Areas (Extracted from "Demographic components of future population growth" [1])

| World and Major Areas | **Total Population (millions)** | | **Population Change 2010-2100** | |
|---|---|---|---|---|
| | 2010 | 2100 | Absolute (millions) | Relative 2010 (per cent) |
| World | 6.916 | 10.854 | 3.938 | 57 |
| Africa | 1.031 | 4.185 | 3.153 | 306 |
| Asia | 4.165 | 4.712 | 546 | 13 |
| Europe | 740 | 639 | -101 | -14 |
| Latin America and the Caribbean | 596 | 736 | 140 | 23 |
| Northern America | 347 | 513 | 167 | 48 |
| Oceania | 37 | 70 | 33 | 90 |

When analyzing the table above, it can be verified that the world population will continue to grow over the next few years, in particular with a growth forecast of 4 million people, with a higher incidence in the regions of Africa and Asia. Therefore, with a population increase, there are a few factors to take into account.

Based on the factors mentioned before, the overall age of the global population tends to increase, due to the increase in the average life expectancy. Figure 2 shows a forecast for the increase of ageing at a global level.

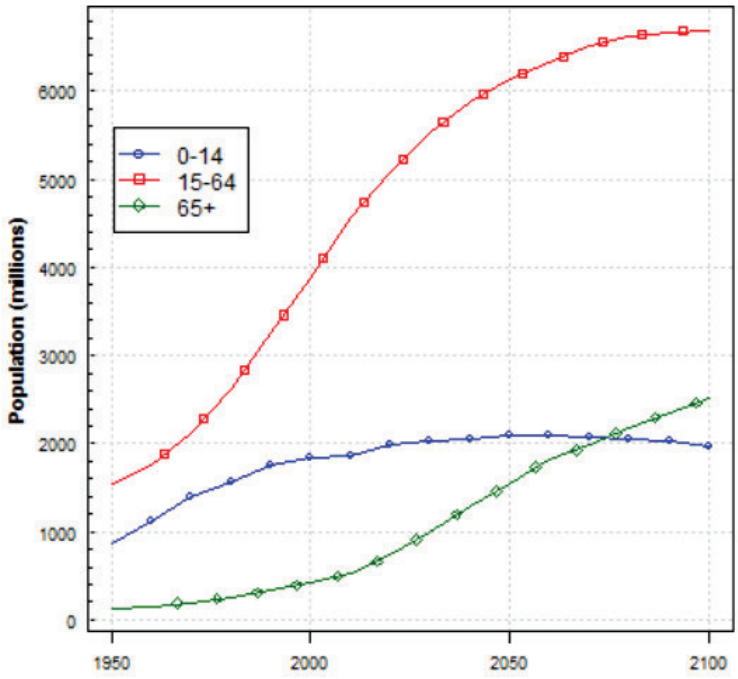



Figure 2.: Total Population by broad age group (extract from World Population Prospect 2017 - United Nation[1].)

In Figure 2 it is possible to verify that over the coming years the world population with 65 or more years of age will continue to increase. By the year 2050, this population is expected to reach 1500 million inhabitants, while in 2100 it is expected to exceed the value of 2000 million inhabitants. It is also possible to observe that the young population will suffer a small decrease until the year 2100.

Based on the numbers described before, the cases of patients with multimorbidity may also increase. Multimorbidity is defined by the presence of two or more chronic diseases in the patients [20]. When health professionals provide treatment recommendations based on CPGs specific to each disease, it can lead to problems ranging from adverse drug events, increased treatment complexity, and an increase of the cost of treatment [21] [22]. Multimorbidity has several consequences such as functional decline and a decrease in people's quality of life [23] [24].

Since most CPGs are disease-specific, when combining these different disease-specific treatment plans, this can lead to interactions and conflicts that can impair the patient

1 https://population.un.org/wpp/Graphs/DemographicProfiles/

clinical condition. The application of multiple CPGs individually can result in complex multiple drug regimens (polypharmacy) with the potential for harmful combinations of drugs.

For instance, a patient with hypertension and diabetes [25] in whom the use of medicines to treat these diseases may conflict. Therefore, the effect of one drug alters the effect of the other. This may result in an adverse drug event that may impair the patient's condition.

To use a real-life example, Verapamil is one of the most commonly used medications to control hypertension [25]. However, in diabetic patients treated with metformin, which is the most common oral medications used to control diabetes, should be avoided. Due to the inhibitory action of verapamil on hepatic metformin uptake by Organic Cation Transporter 1 (OTC1) and related transporters [25].

## 1.2 OBJECTIVES

This work of thesis planning presents as the theme: Conflict Resolution for Clinical Treatments. According to CDSSs that use CIGs as a basis for their knowledge base, it is crucial to study and identify the conflicts and interactions, which may occur when concurrently apply CPGs to patients. Thus, the main objectives of this dissertation are:

- Identification of the main aspects of the resolution of conflicts in clinical treatments, in the context of CDSSs;
- Identification of the main limitations in the current models;
- Designing an execution engine that manipulates the workflow of CPGs tasks and automatically identifies possible conflicts or interactions that may occur when multiple clinical protocols are applied at the same time;
- Execution engine should be able to provide alternative measures (mainly in the form of alternative drug recommendations) that resolve the conflicts identified in three steps: first looking at guideline itself, second using the RxNorm Similar API and finally using an MCDA model;
- Develop interfaces that allow a healthcare professional to visualise clinical recommendations.

## 1.3 DOCUMENT STRUTURE

The present dissertation is structured in two chapters. Chapter 1 provides a brief description of the background of the work, main concepts, and presentation of the motivation for this dissertation. The objectives inherent in the development of the dissertation are stated. A brief description of the dissertation structured is also given.

In chapter 2, the *State of Art* is presented. In this chapter, several CIG models were analysed, the challenges were addressed and limitations inherent to each model were discussed. Another topic covered in this chapter is the Combination of Clinical Practice Guidelines and Multimorbidity, describing some of the existing approaches, with a brief analysis of each. Finally, a study on Multi-Criteria Decision Analysis (MCDA) topic is carried out, identifying its structure, as well as analysing some approaches.

Chapter 3 focuses on the problem of conflict resolution in clinical treatments, addressing a set of elements that describe the problem domain, the functionality that the proposed solution should provide, as well as the actors who interact with the solution. This chapter also describes the use cases of the proposed solution. This chapter also presents the mockups developed.

Regarding chapter 4, the conflict resolution model is proposed using an MCDA approach. This model must be integrated into the CompGuide system. Firstly, the main classes and properties of the CompGuide model are presented. Next, we provided the system architecture, with the proposed solution for conflict resolution, with an explanation of each how it will be developed.

In chapter 5, we present case study implementation where we explain how the system provides alternative measures using MCDA approach. First, we describe the technologies used to develop the case study, followed by the explanation of conflict resolution through an MCDA model. Finally, we present the developed web application.

Finally, chapter 6 summarizes the work done and the main conclusions to be drawn. Future work prospects are also mentioned.

# 2

# STATE OF THE ART

This chapter presents an analysis of the state of the art, to identify the researches available in the literature as well as the different applications that currently exist.

The methodology used to elaborate the research on the topics covered in this chapter consisted mainly of the analysis of conference articles and journals available in the databases of Google Scholar, Science Direct, Pubmed, and others. Among the terms researched, it is possible to highlight: CDSSs, CIGs modelling languages, Combining clinical protocols, Multimorbidity and Multi Criteria Decision Analysis (MCDA).

The purpose of this analysis is not only to identify some of the relevant points in the different existing solutions in the literature but also to reflect on the most important aspects in the domain of this dissertation.

Based on the mentioned scope, section 2.1 provides an analysis of the most representative CIG approaches, with the specification of each studied model and its main components. Also in this section, a brief discussion of the limitations and the challenges of the CIGs models, are presented.

Finally, section 2.3 provides a brief description of the main concepts of MCDA and an analysis of some relevant research works in this area.

## 2.1 COMPUTER INTERPRETABLE GUIDELINES

CIGs are representations of CPGs in a structured and machine-readable digital format. The representation of CPGs as CIGs make it possible to develop decision support systems, which offer a better possibility of affecting clinical behaviour concerning narrative documents of the corresponding text versions [26].

Although the application of these guidelines has great potential, there are limitations related to the development and application of these guidelines. One of these limitations relies on the interpretation of a guideline: the lack of precision of concepts gives rise to ambiguity and limitations in knowledge, in which computers can't handle. CPGs are large documents that have complex and intricate instructions, complex execution structures and they imply the manipulation of too many variables, which leads to non-deterministic and complex algorithms [27].

The implementation of DSS promises a better admission and application of these daily practice guidelines because these systems enable the monitoring of actions and

observations of health care assistants and allow recommendations based on guidelines when treating the patient [28].

CIGs allow a set of benefits, for example, the identification of requirements that have to be verified before having a decision, enabling the aid to health care assistants in critical points of the clinical procedure. The CIGs can automate processes of verification and validation of CPGs [29]. On the other hand, they allow the reuse of knowledge, this is, if the user model separates the CPGs in modules, each with some fragment of knowledge, it's easier to add this fragments in other guidelines and even refers a certain guideline in the context of a more in-depth guideline [30].

To develop these DSSs based on guidelines, four main areas of importance were considered when developing the CIGs to use, which are: modelling and representing the guidelines, acquisition of guidelines, verification, testing, and finally the execution of guidelines [31].

An important aspect of CPGs is the temporal patterns since it is important to ensure the correct application of the task enactment time and exact start and end time of the recommendations. A good interpretation of these temporal patterns is vital to integrate CPG recommendations into health care assistants practice. With this in mind, it's possible to identify two temporal patterns groups [32]. The first group includes temporal patterns that determine how tasks should be executed. These temporal patterns are as follow [33] :

- **Duration** - How long should a task be executed;
- **Repetitions** - How many iterations should a task be executed;
- **Periodicities** - How frequent should a task be executed and the interval time between the executions;
- **Waiting Time** - How long should wait until the end of the previous task and the start of the new task;
- **Repeat Conditions** - Conditions about the state of the patient that has to be verified before the repetition of a task.

The second group of temporal patterns is related to the state of the patient. They are applied to specify the interval that the patient will manifest, or should have manifested a certain clinical condition. So they should be used to reason about the past or future of a patient.

Currently, there are some existing approaches to specifying CIGs, each one with their motivations and characteristics, although none of them is accepted as a standard in decision supports systems [27]. For example, some of these approaches focus on standardization and interoperability, while others focus on policy development or decision support.

In the following subsections, some of the existing CIG models will be described, namely Arden Syntax in section 2.1.1, Guideline Interchange Format (GLIF) in section 2.1.2, Asbru in section 2.1.3, PROforma in section 2.1.4, EON in section 2.1.5 and Guideline Acquisition, Representation and Execution (GLARE) in section 2.1.6. Finally, a critical analysis of CIG approaches is carried out in section 2.1.7, as well as possible solutions to resolve these limitations.

### 2.1.1 *Arden Syntax*

Arden Syntax is one of the best-established frameworks to provide a decision support system. It was founded in 1989 but was only recognized as a standard in 1992 by the American Society of Testing and Materials (ASTM). At this point, Arden Syntax is distributed by the Health Leven Seven (HL7) Group [34].

The first version of Arden Syntax was developed to the inability to exchange clinical knowledge between different institutions. The framework is established as an open standard for representing the sharing of clinical knowledge and defines the guidelines in the following module: MLMs (Medical Logical Modules) [35]. Besides, the Arden Syntax focus on sharing independent guidelines and "simpler" in modules. It was not developed for complex guidelines, for example, the ones that approach treatment procedures. It has been used in different institutions and some companies to develop and implement guidelines in multiple clinical scenarios.

As said before, the guidelines are modelled as MLMs. Each MLM is an ASCII file and represents a single decision with grouped compartments in three categories: Maintenance, Library, and Knowledge [36].

The Maintenance and Library have all the documentation needed to each MLM [36]. The maintenance compartments include simple information about the file author of the MLM (file name, the author, the version, the institutions, the latest modification, the state of the validation).

The library compartment is used for documentation and has the purpose of the MLM, with a detailed description and different passwords.

Clinical knowledge is stored in the knowledge compartment. This category consists of five obligatory compartments (type, data, evoke, logic and action) and two optional compartments (priority and urgency) [36].

These MLMs can be developed through an IDE, the Arden Syntax IDE. This software provides a simple development environment, which includes syntax and testing of MLMs. In addition to this development environment, there are others, such as the Fuzzy Arden Syntax IDE.

### 2.1.2 *GLIF*

Another CIG approach is GLIF. This was the result of an effort of the Intermed Collaboratory with the collaboration of several researchers from several universities, such as Stanford, Harvard, and Columbia, with the main objective of developing a CIG representation that was shareable, and its first appearance dates back to 1998 [37].

The GLIF was developed to model the guidelines by a flow chart that consists of structured steps to scheduling and represent clinical actions and decisions.

The pretended GLIF purpose is to facilitate the sharing of guidelines between different institutions by modelling the guidelines in a way that is understandable by specialists, as well as, automatic analyzers in different decision support systems [38].

The current version of GLIF is the GLIF3. It represents an update of the earlier versions because the GLIF and GLIF2 didn't have constructors that allowed the mapping of patient clinical records elements for the guideline elements [39]. Besides that, the number of constructors in GLIF2 was limited so it didn't allow alternative decision, iterations, and state of patients.

The GLIF3 model consists of classes, their attributes and their relations between classes, required to model the clinical guidelines [40]. The description of the model can be seen in Figure 3.

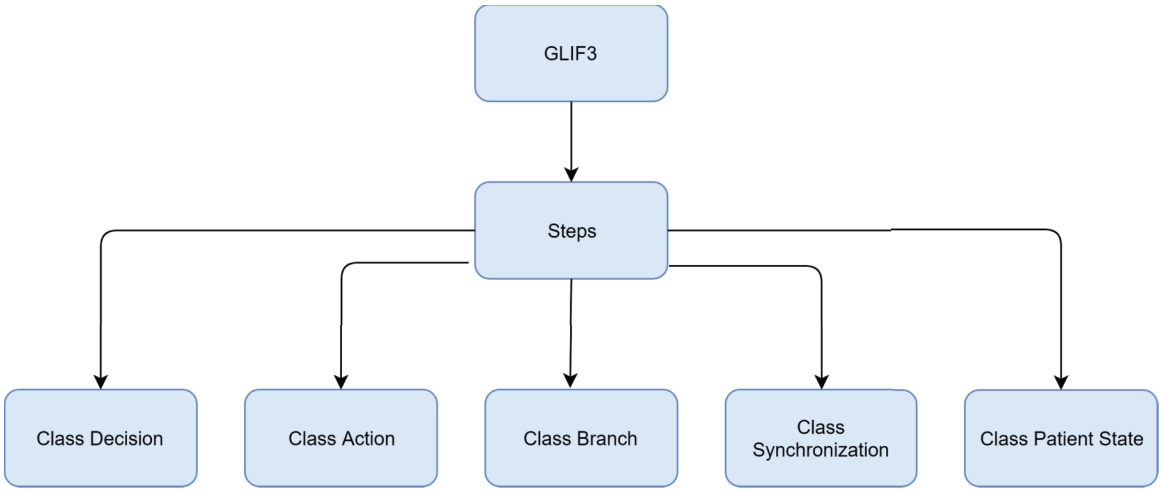



Figure 3.: Schematic of the step classes of the GLIF3 Model

GLIF3 shows the guidelines as flowcharts with temporarily ordered nodes named guideline phases and they're represented as an abstract class called Guideline Step. This class includes the following subclasses:

- **Class Decision Step** - represents the decision points in the guideline;
- **Class Action Step** - it's used to model recommended actions or tasks;

- **Classes Branch Step and Synchronization Step** - used together, they allow modelling multiple concurrent paths over the guideline;
- **Class Patient State Step** - describes the clinical state that characterizes the patient. The state of the patient phase can designate a summary of the clinical state from a patient and can work as an entry point in the flowchart.

The GLIF specification includes one expression and a query language operating over an object-oriented data model. This query language provides a way to access the patient data and to map the variables used in decision criteria and other expressions [30]. The data model used to represent the information is based on HL7.

### 2.1.3 *Asbru*

Asbru [41] represents a guideline modelling language, developed jointly by Ben-Gurion University and Vienna University (more specifically by the Technological University). Regarding temporal aspects, Asbru is quite advanced in its modelling.

Asbru is an approach to CIGs that focus on application and review of temporal oriented clinical guidelines. The main objective of this approach is to represent the clinical guidelines as time-oriented skeletal plans, which are plan schematics with different detail levels [42].

To manage this plans, there are numerous key aspects in Asbru, like the representation of high-level objectives, the representation of temporal patterns, time annotations, and the development of user interfaces to visualize developed plans [43].

Moreover, this guideline consists of a number of elements, of which the Plan element is the most important one [44]. Besides administrative attributes, each plan contains the following attributes: *preferences*, *intentions*, *conditions*, *effects* and *plan body*. Figure 4 describes this attributes.

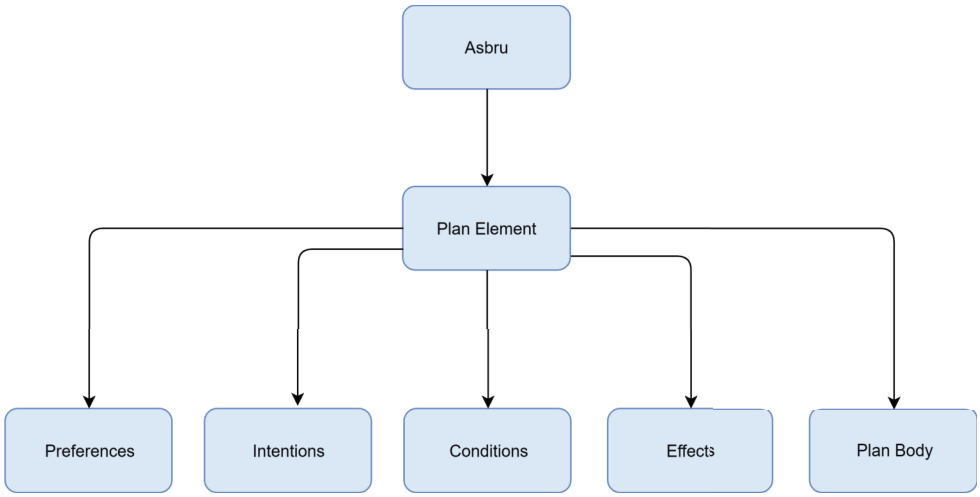



Figure 4.: Schematic of the main classes of the Asbru Model

- **Preferences** - are applicability restrictions of a plan to achieve a certain goal;
- **Intentions** - are used to model the aims of the plan, independent of the plan body;
- **Conditions** - are temporal patterns and are used to change the state of a plan. Asbru defines a number of condition categories such as 'filter-preconditions' and 'setup-preconditions' that need to hold if a plan is considered applicable, 'suspend-conditions' that determine when an action plan must be (temporarily) suspended, 'abort-conditions' that determine when an active or suspended plan has to be aborted and 'completed-conditions' that determine when a plan is (successfully or not) completed;
- **Effects** - can be applied to select the most appropriate plan by describing the expected behaviour of the plan's execution;
- **Plan body** - is a set of sub-plans or actions (which are plans that do not contain any sub plans anymore) that have to be performed whenever the plan is considered appropriate (based on the plan's preconditions, intentions or effects).

Many tools, based on this approach, have been proposed. One such tool is PROTEGE, which generates a fairly well-structured editor from a defined language. Being useful for applications in its essence Extensible Markup Language (XLM), its rigidity with regard to the structure of the dialogues generated increases its difficulty of use. Another tool is the *Asbru View*, which is a preview interface and users to edit the Asbru plans [45]. Due to its graphical representation, this tool has become more accessible to health professionals, and even so, it ignores all existing materials.

### 2.1.4 *PROforma*

Developed at the Advanced Computing Research Laboratory on Cancer in the United Kingdom (UK), PROforma has several distinctive features [46].

PROforma is a CIG approach supported by acquisition and execution tools to support guideline dissemination in the decision support systems that assist patient care through active decision support and workflow management [47].

The name PROforma is a concatenation of the terms proxy ('authorized to act for another') and formalize.

Similar to GLIF, PROforma also represents guidelines as a directed graph in which the nodes are instances of a fixed set of classes.

Each guideline in PROforma is modelled as a plan that consists of a sequence of tasks. The PROforma task ontology defines four classes, each with their attributes: Plans, Decisions, Actions, and Enquiries [48]. These four tasks are derived from the generic Keystone task, which contains several attributes that are common to all four derived

tasks. These include administrative ones that hold a name, caption, or description but also attributes that describe the capabilities of a task such as goals and conditions.

Each Plan models a (sub)guideline. The Plans defines the following tasks:

1. an ordered sequence of tasks.
2. logical and temporal constraints on their enactment.
3. circumstances in which a plan must be aborted or terminated.

The schematic representation of the PROforma Model is represented in Figure 5.

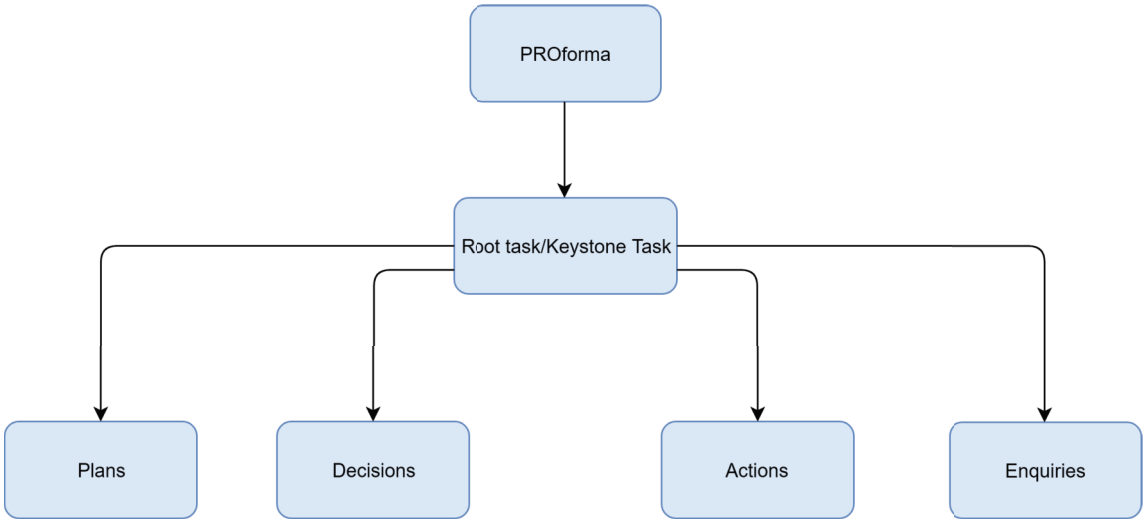



Figure 5.: Schematic of the main classes of the PROforma Model

In Figure 5, the four tasks derived from the keystone task are represented:

- **Plans** - contains additional attributes that store the plan's task network, scheduling, and temporal constraints and abort or termination conditions;
- **Decisions** - is represented as a set of possible outcome candidates plus various types of schemas (logical expressions) that support or oppose each candidate. Every candidate is associated with a set of schemas. Schemas consist of rules, qualitative variables, quantitative weightings, and certainty factors and support or oppose candidates, establishing a preference order among the candidates;
- **Actions** - is a task that a PROforma execution engine can request for enactment by an external agent;
- **Enquiries** - are used to acquire various kinds of information, such as clinical or administrative information. This information can be obtained from a clinical user or can be directly extracted from an external software agent or hardware device.

The PROforma stores the clinical protocols using a Red Representation Language (*R2L*), which is a graphic display language Arezzo,which is a strong goal-based system.

### 2.1.5 *EON*

The EON system is one of the first comprehensive CIG systems [49]. This system was developed by Stanford University and it is a forerunner of systems such as GLIF and Standards-Based Sharable Active Guideline Environment (SAGE).

EON is a CIG approach that aims to develop decision support systems, which reason about guideline-directed care [50]. The EON approach consists of several components that facilitate the acquisition and execution of clinical guidelines.

Similar to GLIF, the EON model, called Dharma, is object-oriented and consists of classes that describe guideline tasks as a sequence of structured temporal steps. The Dharma model is non-monolithic, meaning that it can be extended with additional classes that capture new guideline behaviour [51]. Besides the Dharma guideline model, the EON architecture also contains several run-time components, used to construct execution-time systems.

The primitive classes in EON are represented in Figure 6 and represent scenarios, decisions, actions and goals.

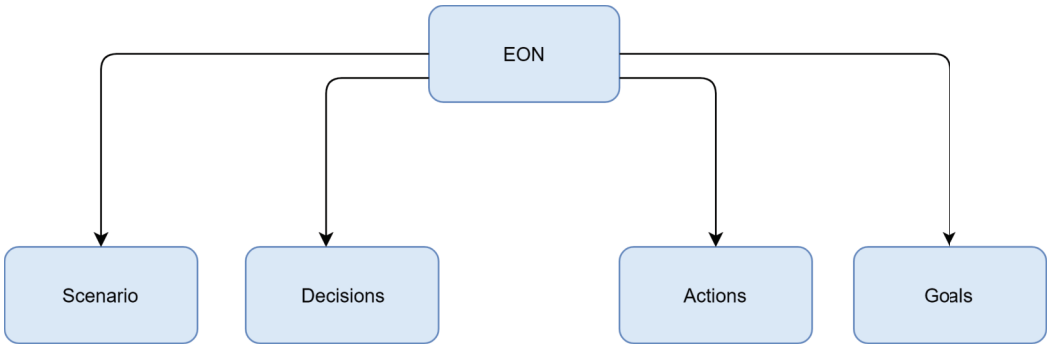



Figure 6.: Schematic of the main classes of the EON Model

The Dharma guideline model, in contrast with GLIF and PROforma that model guidelines in terms of a fixed number of classes, the researchers of EON propose a non-monolithic model, which consists of a standard set of classes that can be extended with task-specific submodels, and are the following:

- **Scenario** - it's a partial characterization of a patient state. In a scenario, the eligibility conditions specify the necessary conditions for the patient to be in this scenario. On the Dharma ontology, a scenario always precedes the decision and action phases;
- **Decisions** - two basic types of Decisions are defined (using two subclasses): decisions that model 'if-then-else' choices and decisions that require making a heuristic choice from a set of pre-enumerated alternatives. The latter is aided by preferences as determined by rule-in and rule-out conditions that support or oppose alternatives;

- **Actions** - are instantaneous acts that lead to changes in the state of the world such as collecting patient data, displaying a message to the user or starting a drug regimen. Actions are used heavily throughout guidelines modelled in EON;
- **Goals** - have states that can change from time to time. These changes are usually the result of actions specified in a guideline, as actions can start a new activity, stop an ongoing activity or change the attribute values of ongoing activity.

Every class in the Dharma ontology can be associated with a goal. The notion of goals is comparable with the notion of intentions in Asbru, although less sophisticated.

### 2.1.6 *GLARE*

GLARE is a software system that includes a model for representing clinical protocols, with a system that can execute them [52]. This system was developed in cooperation between the Department of Computer Science of the University of Piedmont Orientale, Alessandria, and Azienda Ospedaliera San Giovanni Battista, Turin (the third largest Italian hospital), both in Italy. This type of model does not use any standard representation.

Regarding the formalism of representation, GLARE has a limited mechanism but on the other hand, their basic primitives are atomic and compound actions. Atomic actions are used to model elementary steps in a given guideline. In the case of composite actions, these represent more complex procedures that can be defined in terms of their components. Four of the types of atomic actions were introduced from GLARE, work actions, query action, consultation actions, and decisive actions [53].

- **Work Action** - represent operational steps, which are/should be performed at a particular point in the guideline;
- **Consultation Actions** - information requests from the outside world;
- **Decisive Actions** - means to select among several alternative paths.

The architecture of the GLARE system consists of two main modules: an acquisition tool and an execution tool [53]. These two modules are represented in Figure 7.

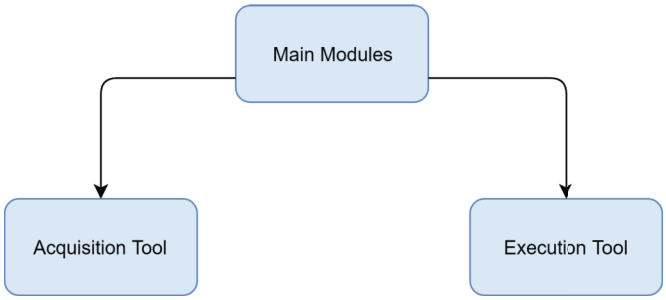



Figure 7.: Main Modules of GLARE Architecture

- **Acquisition Tool** - destined to be adopted to introduce a new guideline in the system, providing a graphical interface to acquire components of the guidelines.
- **Execution Tool** - a tool that is explored by the health professional to apply a specific guideline to a specific patient. This provides practitioners with different forms of consistency checking, ranging from name and interval checking to time consistency checking.

### 2.1.7 *Discussion and Analysis of CIG Modelling Approaches*

´ The representation of CPGs in CIGs, automatically, presents a large number of exceptions and alternatives regarding the solution design. Also, many of the existing CPGs are not designed to be digitally executed, because it involves complex instructions and the manipulation of many variables, which makes it difficult to be interpreted by a computer.

Often, the vocabulary used is evasive, for example, criteria at decision points are not very explicit or do not clearly state what to do. This imprecision produces different meanings and gaps in knowledge, leading to a difficult interpretation by computers. Therefore, the simpler a protocol is, the easier it will be to adapt to the CIG format.

It is necessary to increase the effectiveness and interactivity of CIG systems to enable new services, both information, and communication, to support health professionals in their duties. One example is to include alerts in a way that can help health care professionals take more control throughout the clinical process.

Regarding systems to CIG representation presented in the section, each one has different types of models and represent different guidelines. It is important to check some aspects of each approach. None represent CIGs as a Task-Network Model (TMN), to model the workflow structure of the tasks in the guidelines (e.g. flowchart) except Arden Syntax, which as a collection of totally independent model rules. Thus, Arden Syntax approach is one of the best models to represent simple guidelines. On the other hand, in case of more complex CPGs that present an intricate task network, Arden Syntax presents some limitations.

In general, all approaches support basic tasks of CPGs, such as decisions, actions, and entry criteria, even using different terminology. In the case of actions, in Arden Syntax

are represented by logical slots, in GLIF by decision steps, in PROforma by decision tasks, in Asbru as conditions, in EON as decision and GLARE as work action.

Most approaches provide the support that can modulate complex guidelines into sub-guidelines, such as in GLIF and EON, or subplans in PROforma and EON. However, Arden Syntax is an exception, in the sense that it can not only support this nesting of rules but also call additional rules to the action slot. Nonetheless, there is no general control flow for controlling for those same calls. In the case of the GLIF, this also supports the representation of common guideline structures through MACROS, facilitating reuse of the most commonly used guidelines.

Despite these aspects, none of the approaches can deal with interactions and conflicts that may exist when applying multiple concurrent CPGs to multimorbid patients. Moreover, the application of CPGs independently for different clinical conditions can lead to adverse events that may impair the patient's clinical condition. In other words, there is a general lack of flexibility to support cases where multiple protocols need to be combined, which is the most challenging part when dealing with multimorbid patients. Also, these models are unable to detect the conflicts for combinations of protocols automatically. In the next section, some approaches to deal with this problem will be described.

## 2.2 Combining Clinical Pratice Guidelines - Multimorbidity

Another relevant theme in this dissertation is the interactions and conflicts that may occur when merging CPGs for multimorbid patients. As mentioned in section 2.1.7, multimorbidity is the major limitation in existing CPGs. First of all, the three technologies covered to integrate with CDSSs to deal with the multimorbidity problem, are presented. After a brief analysis of these technologies, the different formalisms for combining CPGs with multimorbidity are described.

According to Abid et al. [54] knowledge about the diseases that affect multimorbid patients can focus on two fundamental points:

- **Modeling Level** - when different disease-specific CPGs are integrated into a single structure that is used to support the clinical decision;
- **Execution Level** - when each of the CPGs is applied, and the suggestions of each are integrated with a single proposal of clinical practice.

Firstly, it is necessary to identify the three great types of technology intrinsically linked to this theme [55]. In the Figure 8 these three approaches are represented.

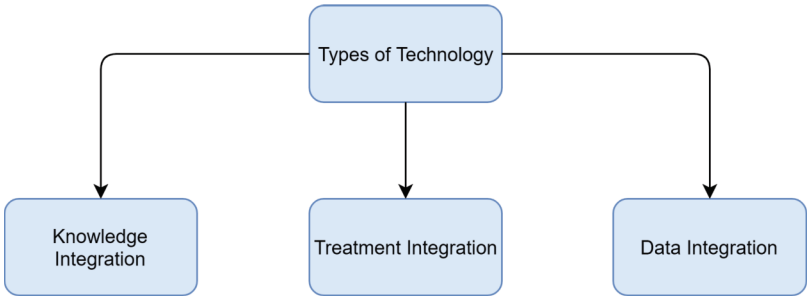



Figure 8.: Types of Technology

It is then necessary to know the meaning of each one of them.

- **Knowledge Integration** - this type of integration aims to integrate all the medical knowledge that is available for the management of multimorbid patients. The most complex technologies are defined in this integration, which is the most difficult to manage and validate;
- **Treatment Integration** - its focus relies on the detection and resolution of conflicts in specific clinical interventions. As far as technologies are concerned, these are more practical in the sense that they are defined to support the needs of health professionals since they have a low degree of autonomy, thus invalidating support for preventive medicine;
- **Data Integration** - finally, the integration of data has as main concern to identify patterns common to clinical practice in the accumulated data on the treatment of patients with multimorbidity. The technologies used are adaptable to the changes that may arise in the treatment patterns as well as the therapeutic characteristics of the medical scenario, and this requires a pre-processing of the clinical data.

In each of the three types of technologies incorporated different formalisms that are based on integrating the different clinical treatments in the case of patients with multimorbidity.

In the following sections, three formalisms of combining CPGs are presented: OntoMorph in section 2.2.1, Constraint Logic Programming (CLP) in the section 2.2.2 and Transition-based Medical Recommendations model (TML4I model) in section 2.2.3. Finally, in section 2.2.4 is made a critical discussion of the approaches described in this section.

### 2.2.1 *OntoMorph*

OntoMorph aims to propose a treatment plan, distributed in several tasks that do not conflict and are as efficient as possible, both in terms of time and resources.

The OntoMorph approach was proposed by Jabarpour [56] and aims to define a set of ontologies to represent [55]:

1. The guidelines - local knowledge ontology (LKO).
2. The general domain - domain of knowledge ontology (DKO).
3. The mappings between LKO and DKO - knowledge mapping ontology (KMO).
4. Decision rules for execution of LKO provided by experts in their domain - knowledge transformation ontology (KPO).

The use of this diversity of ontologies can potentiate some challenges from having rules for all decision steps to maintenance and even to consistency [57]. In the case of the maintenance in the way of managing the consequences that can exist of the alteration of the ontologies or the own mapping, already in the case of the consistency, concerning the verification of contradictions between the local rules or the domain.

All these challenges require experts to pay particular attention to several issues, such as identifying all the possible interactions between LKOs, providing decision rules that are consistent and that correspond as a solution to existing conflicts. These same rules should be as general as possible to apply to all existing combinations of LKOs.

Another of the great challenges of using this approach are the mappings, which must exist between heterogeneous data. These mappings may require more complex techniques, such as natural language processing.

OntoMorph uses Ontology Web Language (OWL) - which is a W3C standard for web ontologies - and Semantic Web Rule Language (SWRL) both to represent and to merge the different guidelines. SWRL aims to define constraints as an entity that relates actions. These same constraints are created manually.

### 2.2.2 *Constraint Logic Programming*

CLP is an approach proposed by Wilk et al. [58] that describes the guidelines as an activity chart. The use of CLP allows identifying the conflicts that may result from the application of two CPGs to the same patients and propose alternatives to these conflicts [59].

This proposition can only be applied to specific situations, because the temporal aspect is ignored, i.e., diseases that are diagnosed during a single encounter between the patient and the health professional. Besides, it is considered that the predicates use the same terminology and that in the end there can only be two states: true and false [59].

Although this approach allows the identification of conflicts and also provide automatic solutions, it is dependent on the availability of the knowledge bases that are associated with each guideline. This point means that both conflicts and their solutions need to be defined in advance, such as medical background knowledge, as guidelines-dependent constraints.

### 2.2.3 *Transition-based Medical Recommendations model*

The TMR4I approach aims to detect interactions between different clinical recommendations, especially in cases of patients with multimorbidity. The different recommendations combined in the different guidelines may interact, for example, presenting inconsistencies, and may lead in extreme cases, to the administration of these recommendations are harmful to the patient [60]. Today, this approach is used to find conflicts that exist between CPGs statements around drug prescribing, and can also be used for treatment recommendations that do not include drugs.

In this model, meta-rules are defined to identify and reconcile three drug categories using SPARQL queries - W3C standard for semantic queries. These meta-rules define how much conflict is identified and how drugs with similar effects (without conflict) are selected for CPG-Knowledge.

Before detailing this model, it is necessary to define some concepts that are extremely important throughout the approach [60].

- **Care Actions** - this concept represents the various types of action that can be performed by health professionals to be able to change a situation;
- **Transitions** - represents the possibility of being able to change a situation concerning a particular patient performing a particular type of care action;
- **Situations** - finally, this concept represents a property as well as all its admissible values.

The main concept of this approach is based on interaction [60]. This same interaction can be based on two types: internal or external.

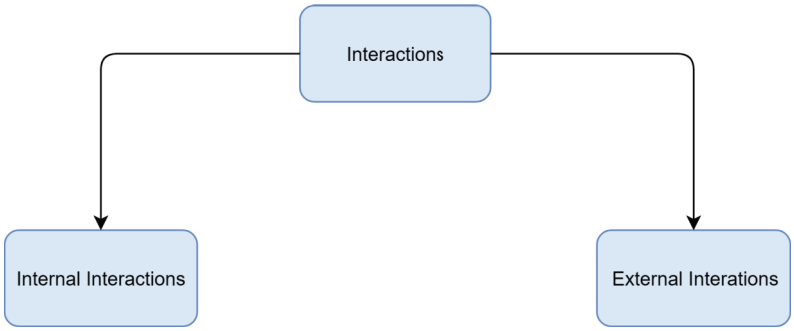



Figure 9.: Types of Interactions

As shown in Figure 9, two types of interactions are identified:

- **Internal Interactions** - interactions between the recommendations themselves;
- **External Interactions** - interactions in which it is necessary to access some external database containing clinical knowledge (e.g Drugbank).

Regarding internal interactions, there are three categories of conflicts: repetition interaction, contradiction interaction and alternative interaction [61]. These three categories are illustrated in Figure 10.

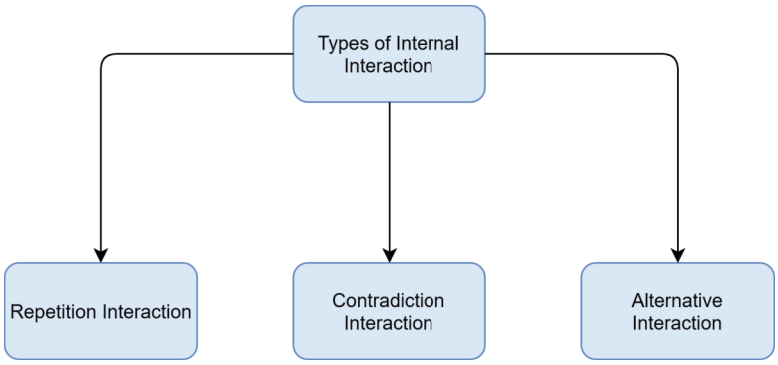



Figure 10.: Types of Internal Interactions

The meaning of these three types of interactions is as follows:

- **Repetition Interaction** - set of repeated recommendations for the same care action;
- **Contradiction Interaction** - interactions that occur when two recommendations can lead to conflict if they are recommended at the same time;
- **Alternative Interaction** - set of alternative recommendations.

An example is shown in the Table 2 bellow[1], which is based on the administration of aspirin, in the three categories of conflicts for internal interactions.

Table 2.: Example of Internal Interations

| **Category** | **Example** |
|---|---|
| **Repetition** | Administer aspirin |
| **Contradiction** | |
| To same care Action | Administer aspirin/ Do not administer aspirin |
| To similar transitions | Lower blood pressure / Avoid lowering blood pressure |
| To inverse transitions | Lower blood pressure / Increase blood pressure |
| **Alternative** | |
| To similar transitions | Administer aspirin, ibuprofen and naproxen to handle inflamation |
| To inverse transitions | No apirin to avoid increasing the riskof gastrointestinal bleeding / PPI to decrease risk of gastrointestinal bleeding |

In the case of external interactions, external sources of clinical knowledge are used to resolve conflicts. Within this type of interaction, there are two conflicts: Incompatible Drugs Interaction and Alternative Drugs Interaction [60]. Figure 11 illustrates these same types of conflicts.

1 Extracted from "Analyzing recommendations interactions in clinical guidelines" [61]

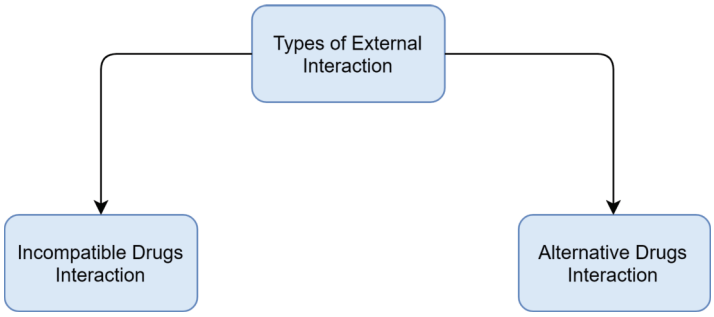



Figure 11.: Types of External Interactions

As mentioned in the case of the internal interaction, it is necessary to understand this type of interactions their types of conflicts.

- **Incompatible Drugs Interaction** - this conflict exists when two recommendations that are adopted are associated with external information on incompatible drugs;
- **Alternative Drugs Interaction** - when a recommendation on a drug has a contradictory or incompatible interaction, and then another type of drug of the same category is suggested that is not incompatible with other recommended drugs.

Using again the example of aspirin administration, Table 3 [2] shows us the example of acting for the two conflicts in the external interactions.

Table 3.: Example of External Interations

| **Category** | **Example** |
|---|---|
| Incompatible Drug | Retrieved from Drugbank |
| Alternative Drugs | Retrieved from Drugbank |

### 2.2.4 *Discussion of the Approaches to handle Multimorbidity*

The TM4I and CLP aim to identify drug conflicts, while OntoMorph focuses on the scheduling of CPGs tasks.

Both OntoMorph and CLP define conflicts, more precisely when two specif CPGs are executed and these conflicts are defined based on constraints. These constraints are stored in a knowledge base, i.e, for each pair of CPGs that are included in a CDSS, these conflicts need to be defined manually by health experts, while restrictions must be defined, also manually by knowledge engineers.

Regarding TM4I, this approach defines meta-rules for general conflicts and is independent of the guidelines. At a later stage, the recommendations of the CPGs are converted into rules, interpretable by the computer, using a syntax of meta-rules. By applying this meta-rules, the conflicts between the different CPGs recommendations and an external knowledge base does not require manual completion.

2 Extracted from "Analyzing recommendations interactions in clinical guidelines" [61]

For all of these approaches, execution mechanisms have been developed so that they can verify the CPGs recommendations in the case of OntoMorph and the CLP in the form of conflict constraints and the case of TM4I in the form of meta-rules.

Among these three approaches, there are differences in the verification of recommendations from CPGs, while in the TM4I approach it is possible to combine an unlimited number of CPGs to identify conflicts, approaches to OntoMorph and CLP are only possible to combine two CPGs.

The use of meta-rules, compared to conflict constraints, has some advantages. Conflicts identified with the use of meta-rules do not require manual identification since they can be derived automatically from the representation of CPGs. Due to the fact they can be reused. This has advantages over conflict restrictions because they require knowledge to be acquired and added manually if there is a change in a guideline or the existence of a new combination of diseases. Despite this the bottleneck in this approach in converting CPGs into rules interpreted by the computer.

All presented approaches, except OntoMorph, focus on the conflicts between drugs, not taking into account other aspects such as dosage or time. Although the TM4I already has some focus on "external conflicts" with access to external sources to assist in conflict resolution, all approaches focus on "internal conflicts", i.e, conflicts between the recommendations themselves, which that in some cases external information is needed to resolve these conflicts.

In general, these approaches hardly use any information from the patient and existing conflicts may also arise from non-pharmacological recommendations or external information (e.g. a patient's diet). In the case of TM4I, which already allows the use of external sources (e.g. Drugbank) for the resolution of conflicts, it is not prepared for the case in which it is not possible to provide alternative recommendations to resolve the conflicts. Moreover, they cannot lead to cases where decision-makers have conflicting solutions or cannot decide on the best treatment alternatives. To provide the best alternative treatment plan, it is necessary to evaluate the risk of applying the adverse recommendations and get patient's preferences on the best treatment alternatives. Therefore, there are other techniques that better address this problem, such as MCDA that will be analysed in the next section.

## 2.3 MULTI CRITERIA DECISION ANALYSIS

MCDA is used when there are conflicting objectives and decision-makers cannot decide on the best treatment alternatives. MCDA is an integrated assessment approach to sustainability. This approach supports decision making to address high complexity problems in which there are multiple solutions with conflicting objectives exist [62]. According to Belton and Stewart [63], the MCDA is defined as "a generic term to describe

a collection of formal approaches that seek to explicitly take into account various criteria to help individuals or groups to explore decisions that matter."

MCDA has been used with some success to support decision support systems that have complex problems and there are several advantages in using this approach, such as the capability of assessing and integrating multiple criteria, comparison and assessment of different decision alternatives, the possibility of structure an assessment of a complex problem, the possibility of deal with incomplete and uncertain information and helps stakeholders summarise complex value trade-offs consistently and transparently helping to do fairer decision-making.

In the decision-making process there are a few steps to follow [64]:

- Identify the objective that is intended in the decision-making process;
- Select the decision criteria;
- Select the alternatives;
- Select of the weighing methods to represent the importance;
- Use the Aggregation Method;
- Decision making based on the results of the Aggregation Method.

The MCDA approach follows a set of fundamental principles, which are described below [64]:

1. **Objectives Identification** - in any MCDA process, it is necessary to understand the problem and the corresponding decision objective. It is also necessary to identify the stakeholders, the alternatives under consideration and the required outcome;
2. **Select Criteria** - this selection must be consistent with the decision. The criteria should be independent of each other, represented on the same scale and should not be related to alternatives;
3. **Select Alternatives** - the selected alternatives must be accessible, comparable and feasible;
4. **Select the Weighting Methods to Represent Importance** - the methods of weight determination should be decided based on the different approaches: value measurement models, outranking models, and reference-level models;
5. **Aggregation Method** - this method can have different ways of being represented: it can be a product, an average or a function. The result of applying this method will separate the best alternative from all the others that have been selected.

As mentioned, there are three types of approaches for determining weights [65]. Figure 12 illustrates these same three types.

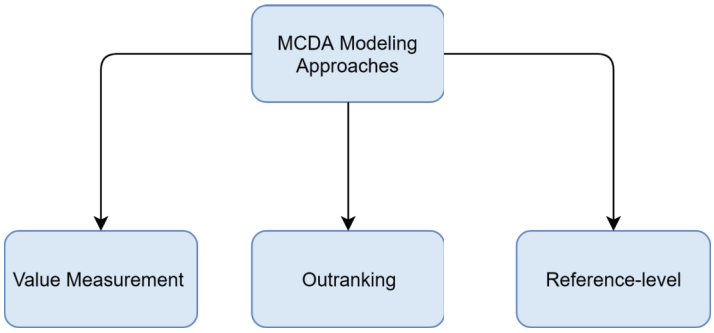



Figure 12.: MCDA Modeling Approches

- **Value Measurement Models** - this type of approach is based on the construction and comparison of numerical scores, to identify the degree to which one decision alternative is more preferred than another. Furthermore, this approach uses additive models, based on the multiplication of a score for each alternative by weight relative to a certain criterion, and these weighted scores are summed to obtain a total score for each alternative;

- **Outranking Models** - in this method, a paired comparison of alternatives is applied in each selected criteria. For that purpose, criteria are combined to obtain a measure that allows supporting the selected alternative, which is the alternative with the highest score, in the rank of alternative solutions. One of the characteristics of this method is that it is possible, under certain conditions, that two alternatives be classified with incomparable. This incompatibility may be associated with the lack of information for these alternatives, at the moment in which the evaluation is performed;

- **Reference-Level Models** - this approach involves a search for the best alternative, depending on which is close to reaching the minimum levels, which are predefined in each criterion. This approach is mostly based on linear programming techniques.

Of all these three approaches, the value measurement model is the most used in the health area, compared to the other two approaches. Therefore, it is important to identify the steps involved in this type of approach. These same steps are identified in Table 4 below[3].

In this dissertation, of these three identified approaches, the value measurement model is followed.

3 Extracted from "Multiple criteria decision analysis for health care decision making—an introduction: report 1 of the ISPOR MCDA Emerging Good Practices Task Force" [65]

Table 4.: Steps of the value measurement model

| | |
|---|---|
| **Define the decision problem** | Identification of objectives, types of decision, alternatives, stakeholders and required results |
| **Select and structure the criteria** | Identification of relevant criteria for evaluating alternatives |
| **Measuring performance** | Collect data on the performance of the alternatives in the criteria and summarize this in a performance matrix |
| **Scoring alternatives** | Eliciting stakeholder preferences for changes within the criteria |
| **Weighting criteria** | Eliciting stakeholder preferences among the criteria |
| **Calculate the aggregate scores** | Use the alternative scores in the criteria and weights for the criteria to get total value by which the alternatives are classified |
| **Dealing with uncertainty** | Carry out an uncertainty analysis to understand the level of robustness of the MCDA results |
| **Preparation of a report and examination of results** | Interpret the outputs of the MCDA, including uncertainty analysis, to support decision making |

### 2.3.1 *Examples of using the MCDA*

This subsection describes some examples where the MCDA approach can be used in health care decisions, in particular, Health Technology Assessment (HTA) [66] and Shared Decision Making (SDM) [67]. These two examples were the ones that best fit the theme of this dissertation. Both studies rely on the presence of participants to obtain their preferences, to use them in clinical decision making.

*Heath Technology Assessment*

The German Institute for Quality and Efficiency in Health Care initiated, in 2010, a study whose objective was to explore the application of the different MCDA methods, so that these could be a means to incorporate the patient's involvement in the HTA process [68]. Despite the involvement of patients in the HTA process and general health care, it is recognized that there are no quantitative approaches to establish their preferences regarding treatments. The project carried out by this institute was based on two techniques of MCDA, the Analytical Hierarchy Process (AHP) [69] and Discrete Choice Experiments (DCE) [70], as a method of analyzing these preferences.

Regarding the first, there was a division into two groups: one with twelve patients and the other with seven health professionals. In each group, their members assessed their

preferences regarding the importance of the different endpoints of the treatment with the use of antidepressants, elaborating a parity comparison of the individual endpoints. The results of these comparisons were analyzed to generate the relative importance of each endpoint.

In the case of the DCE technique, both patients and health professionals had to choose between two alternative forms of treatment, in the specific case for hepatitis C, that differed in performance in various treatment characteristics (e.g. outcomes). These select were later analyzed using logistic regression models, to estimate the importance of the individual characteristics of the treatment.

In conclusion, by applying these two techniques, both studies could provide a way to support the HTA process and some challenges need to be resolved before large-scale implementation, namely challenges methods.

*Shared Decision Making*

In the United States, there are many screening options for people that have a medium risk of suffering from colorectal cancer. Screening guidelines recommend that clinical decisions reflect individual patient preferences [65]. The MCDA technique, called AHP, was used to elucidate the decision priorities of these same people. Some primary care services in the United States have used this approach, such as Rochester in the state of Minnesota.

Based on the American guidelines, the various researchers identified some criteria, for example, [65]:

- Ability to prevent cancer;
- Avoid side effects;
- Minimize false positives;
- Logistic complexity, which is divided into three subcriteria: test frequency, necessary preparation, and test method.

The various participants in this study were asked to indicate the importance given to two criteria, on a scale ranging from 1 to 9, where the value 1 indicated that the criteria were equally important and 9 that one of the criteria would be extremely important compared to the other. These comparisons were made through a computer program, to use it to calculate the priorities assigned by all participants to the criteria presented. In addition to this request, participants were also questioned about the feasibility of using the AHP technique and most indicated that the criteria presented were easy to understand. A majority of participants also replied that they would be willing to use a similar procedure to help make health care decisions.

In the end, the study was able to conclude that patients are capable and are also willing to perform an AHP analysis. This study allows concluding that is possible to promote decision making, using these techniques, patient-centred.

### 2.3.2 *Discussion of the MCDA Approach*

The ability to evaluate and integrate various criteria as well as the comparison and evaluation of different decision alternatives are some of the advantages of the MCDA approach. Another advantage is the possibility of dealing with information, whether incomplete or uncertain, to help different stakeholders to make fairer decisions.

Despite these advantages, there are some limitations in using this approach in a decision-making process. Although there are many MCDA models, none of them can be considered the most appropriate for all situations, which means, there is no standard model that can be followed in general. This means that a few objective guidelines help to choose the most appropriate MCDA model.

On the other side, MCDA methods are only suitable for capturing the preferences of a small group of stakeholders rather than all individuals in a population. Another problem associated with MCDA is the quantification of criteria, as the actors in this process must be as explicit as possible. The assessment of criteria can vary widely, due to the existence of a personal assessment of the importance and severity of health conditions adjacent to a disease. Because of these aspects, it is plausible that models may make mistakes in determining patient preferences.

In short, this type of model is now widely used in case studies and has not yet entered the clinical facilities. In larger situations, it is difficult to evaluate and discuss the application of these methods in the clinical environment. However, the feedback to the developed case studies has been quite positive. At this time it is necessary to take the next step and to apply this type of model in the clinical environment.

# 3

# CONFLICT RESOLUTION PROBLEM ANALYSIS

This chapter will cover several key points to consider that the final project meets the proposed objective. It is necessary to state several elements that allow the understanding of the domain, as well as the system functionalities and the inherent interactions.

Section 3.1 presents the understanding of the problem domain, with the illustration of the Domain Model. Section 3.2 presents the actors and how they interact with the system. Next, section 3.3 describes the functional and non-functional requirements. Section 3.4 presents the Use Cases diagram for the problem in question and their textual description. Finally, section 3.5 presents the mockups of the solution. Mockups serve the guidelines to illustrate the system appearance.

## 3.1 DOMAIN MODEL

A Domain Model represents the classes of the problem to be addressed and should represent the understanding of the information that the system will manage. This model identifies concepts related to system requirements by analysing the problem from a conceptual perspective.

The use of such models is not intended to model software architecture, but rather represents the domain of the problem. Everything that has to do with system architecture belongs to the domain of the solution. Therefore, the domain model is independent of the physical solution.

Figure 13 represents the domain model entities developed for the problem of this dissertation. Only entities for conflict resolution in clinical treatments are represented.

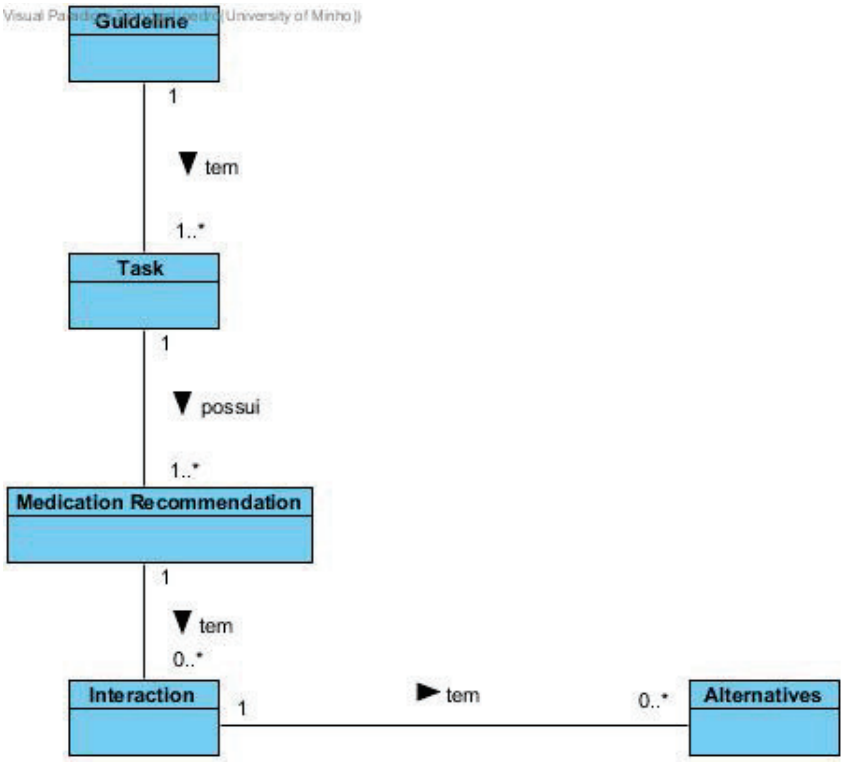



Figure 13.: Domain Model

- **Guideline** - this entity represents the clinical protocols. Each clinical protocol has one or more tasks that must be managed by health professionals;
- **Taks** - represents the tasks to be performed by health professionals. A task has one or more medical recommendations;
- **Medication Recommendation** - drug recommendation application. A medication recommendation may or may not have drug-drug interactions;
- **Interaction** - represents the interactions that may exist in a drug recommendation. An interaction may or may not have drug conflicts;
- **Alternative** - entity that represents alternatives to a particular drug conflict. When one conflict exists one or more alternatives are available (section 4.2.3).

## 3.2 System Actors

In this dissertation, only one type of users was identified: Health Professionals. This type of user is the one who will interact directly with the system, such as setting the ranges of each criterion.

In general, the actors will interact with the system in various ways, representing themselves as important elements of the system's mechanism of operation.

Although not acting directly with the system, patients are a key player in the proposed solution to the problem, because it is in conjunction with them that health professionals

will indicate important values to the system to achieve the desired solution. Patient preferences regarding clinical recommendations must be obtained. This process is performed by obtaining weights between the parties on the different criteria available. The entire process resulted from a discussion between the patient and the physician to be taken into consideration when applying a treatment plan for the patient.

## 3.3 REQUIREMENTS

This section specifies functional (subsection 3.3.1) and non-functional (subsection 3.3.2) requirements. These requirements are specified only in a written and descriptive manner. Requirements were obtained by analyzing existing projects in the State of Art, in section 2, and by their limitations.

### 3.3.1 *Functional Requirements*

Functional requirements are intended to describe the features that the system will provide completely. The proposal for this dissertation should:

- Allow the alternative execution of tasks, i.e., the user must choose from the available tasks to be performed;
- Allow identifying drug interactions between tasks using the RxNorm Interaction API;
- Allows resolve drug conflicts by providing alternative recommendations (recommending alternative drugs).

### 3.3.2 *Non Functional Requirements*

Regarding non-functional requirements, these are related to the use of the application itself, in terms of usability, availability, technologies involved, etc. The defined non-functional requirements are:

- Do not allow the assigned values to be outside the specified ranges;
- The interface should look attractive, interactive and easy to manipulate;
- The system must maintain the same performance even when there is a significant increase in system users;
- The interface must follow a certain pattern, i.e., not different from window to window;
- The proposed solution should be executed on most platforms.

## 3.4 USE CASES MODEL

The use case model allows describing the interactions between the program and the system user. This model helps to create the program interface and its behaviour towards the user. In the following sections, the use cases diagram for the problem is presented, as well as a description of a use case of this diagram.

### 3.4.1 *Use Cases Diagram*

A use case aims to describe a set of actions performed by the actors and the system. A series of interactions are defined between the system and actors that allow a particular goal to be achieved. Figure 14 presents the use cases diagram for the problem of this dissertation.

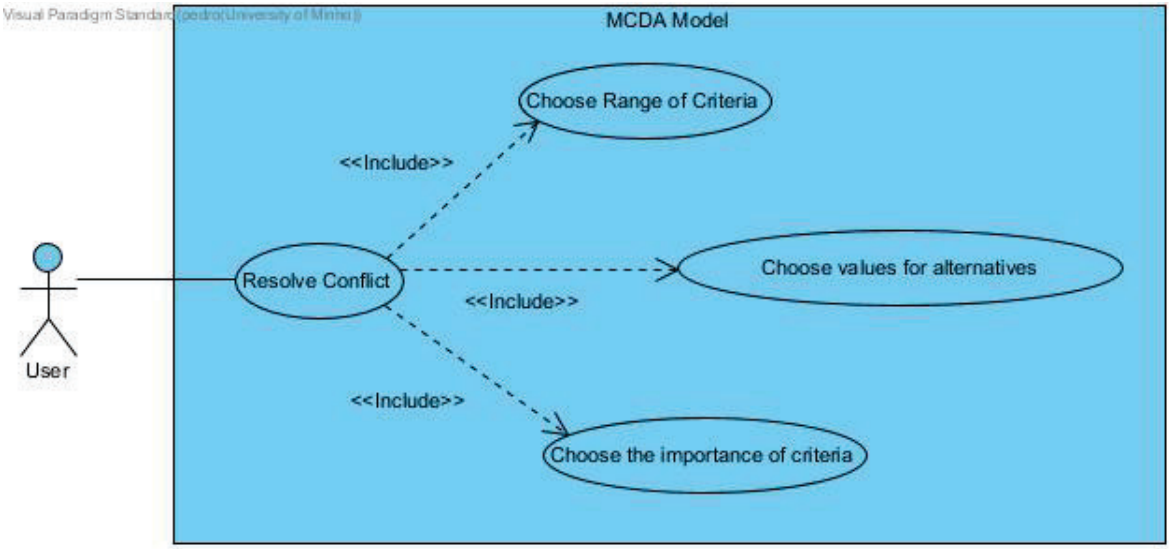



Figure 14.: Use Cases Diagram

### 3.4.2 *Description of Use Case*

This section provides a use case description. The use case presented follows a tabular format, adding additional details beyond what is represented in the use case diagram. Table 5 shows the textual description of the *Use Case Resolve Conflict*. This use case is responsible for representing how the entire conflict resolution process is developed, representing the various steps throughout the process. The remaining descriptions of uses cases can be found in Appendix A.

Table 5.: Description of Use Case Resolve Conflict

| | |
|---|---|
| Super Use Case | |
| Brief Description | Use Case that describes user interaction and system for resolving a drug conflict |
| Preconditions | There is a drug conflict |
| Post-conditions | Choosing an Alternative to Drug Conflict |
| Flow of Events | |

| | Actor Input | System Response |
|---|---|---|
| 1 | Select task details | |
| 2 | | Provides task details |
| 3 | Select ”Recommendation alternatives” | |
| 4 | Select ”Resolve Conflict” | |
| 5 | | Provides available alternatives |
| 6 | Select ”Next” | |
| 7 | | <<include>> Choose Range of Criteria |
| 8 | Select ”Next” | |
| 9 | | <<include>> Choose values for alternatives |
| 10 | Select ”Next” | |
| 11 | | <<include>>Choose the importance of criteria |
| 12 | | Features MCDA model summary table |

## 3.5 UI MOCKUPS

In software development, the creation of mockups is verified before creating the user interfaces. It shows to the end-user, a draft of the interfaces with the adjacent functionality. For the development of mockups, the tool Balsamiq Mockups was used.

The first mockup, shown in Figure 15, indicates the tab where the health professional chooses the ranges for each criterion.

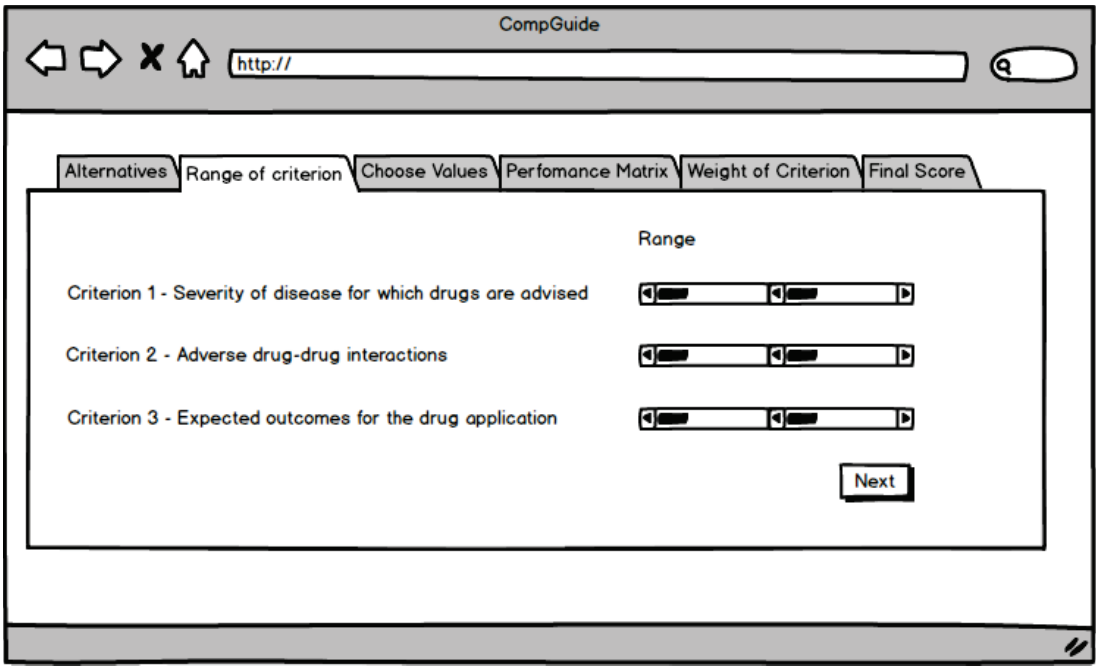



Figure 15.: Mockup for choosing the range of each criterion

Figure 16 shows the tab that corresponds to the choice of importance given to each criterion. After the user input values, it is possible to calculate the weights.

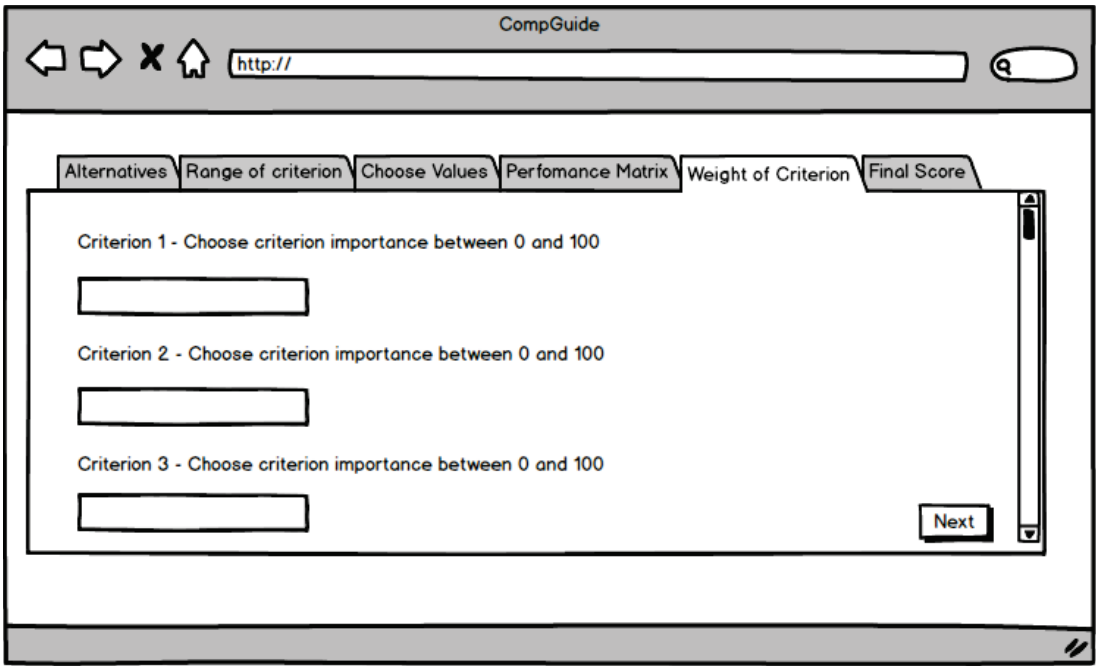



Figure 16.: Mockup for choosing the importance of each criterion

Finally, Figure 17 illustrates the table summarising the MCDA model, where the details and final score of each alternative are displayed.

The remaining mockups developed are in Appendix B.

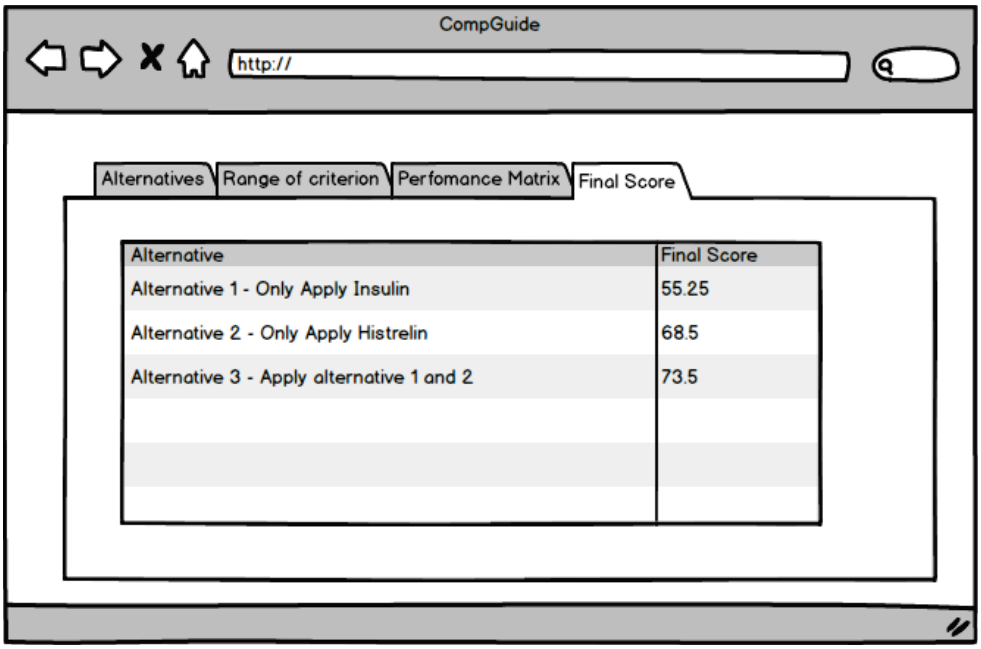



Figure 17.: MCDA Model Summary Table Mockup

# 4

# PROPOSAL OF A CONFLICT RESOLUTION MODEL

In this chapter, we provide details about the system that represents and identifies drug-drug interactions, using the RxNorm API and also provide alternative measures to mitigate these interactions. We use a mitigation function to calculate alternative drugs to the ones recommended that would not cause any conflict. This function uses different mitigation principles to determine solutions such as the similarity between drugs, patient preferences over clinical recommendations and clinician priorities over goals.Section 4.1, describes the MCDA model to conflict resolution.

Section 4.2 presents the CompGuide architecture for the CIG execution, addressing the three levels that encompass the following stages of CIG deployment: representation of CIGs, identification of recommendation interactions, and generation of alternative recommendations.

Although, in the scope of this dissertation, the focus is on generating alternative recommendations using MCDA, a full explanation of the CompGuide model is required, where it will be inserted.

## 4.1 COMPGUIDE ONTOLOGY FOR CLINICAL PRACTICE

CompGuide ontology aims to provide a representation of clinical protocols, with representation in a task network, in OWL. Complex information elements are represented as instances of classes, having these various properties, and simple information has its representation in the data property. Nevertheless, as regards simple information that may be reusable and possible for use in various parts of the clinical protocol, it is represented by instance forms of specific classes.

Clinical practice is represented as an instance of the *ClinicalPraticeGuideline* class, and individuals in this class have a set of data properties as well as objects that allow a descriptive and administrative representation of information found in these protocols. The information contained in the clinical practice is diverse, such as the name of the respective protocol, its general description, the date of its creation and its last update, the protocol version, the clinical speciality, the category, target users and target population. Figure 18 illustrates the definition of the clinical protocol for the treatment of colon can-

cer provided by the National Comprehensive Cancer Network (NCCN) in CompGuide ontology [71].

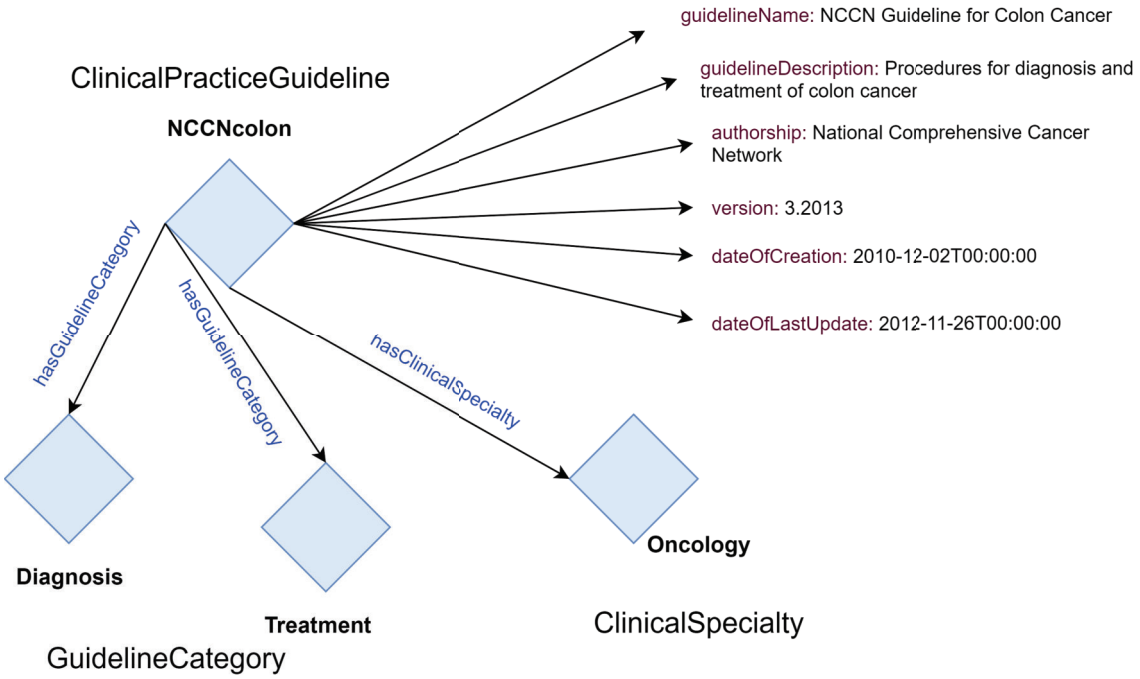



Figure 18.: Clinical Protocol from the NCCN for treatment of Colon Cancer in the Compguide Ontology

Each instance of a clinical protocol is linked to an instance of the *Plan* class, which is a task container, a complex task. An instance of Plan is linked to other instances, which represent basic tasks. This instance can include instances of other *Plans*, which makes it possible to work at different execution levels.

Basic tasks can be represented by three classes: *Action*, *Decision*, and *Question*, as shown in the diagram in Figure 19. Basic tasks aim to create a recommendation plan that contains specific task information.

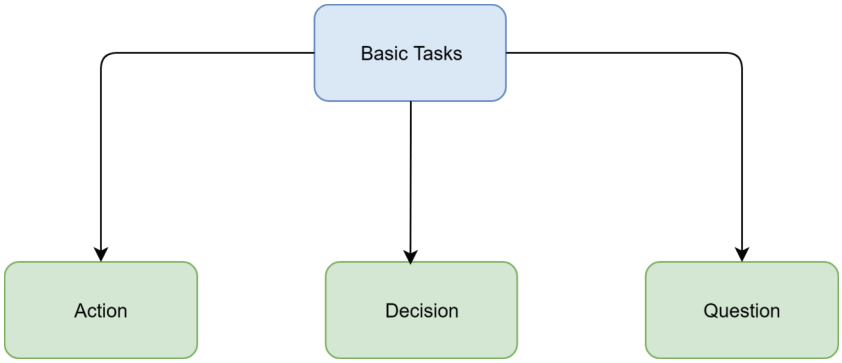



Figure 19.: Basic Task Types

- **Action** - this class represents a procedure that must be performed by a healthcare professional. In CompGuide ontology, there are several subtypes of this class that

can specify their nature in greater detail, such as exams, procedures, medication recommendations, and simpler recommendations.

- **Decision** - regarding this class, it aims to make inferences about the patient's condition, the most concrete example being the clinical diagnosis.
- **Question** - the purpose of this class is to obtain information that may characterize the patient's condition, from signs and symptoms to the patient's health condition. Also, this class allows to record information from health professional observations, as well as save the results of clinical examinations, thus having all the information necessary for the execution of a clinical algorithm.

Despite the existence of these three classes, in this dissertation, we only considered the *Action* class, because it describes clinical tasks that should be performed in the daily clinical practice by a health professional.

The *Action* class has 4 parameters, as illustrated in Figure 20.

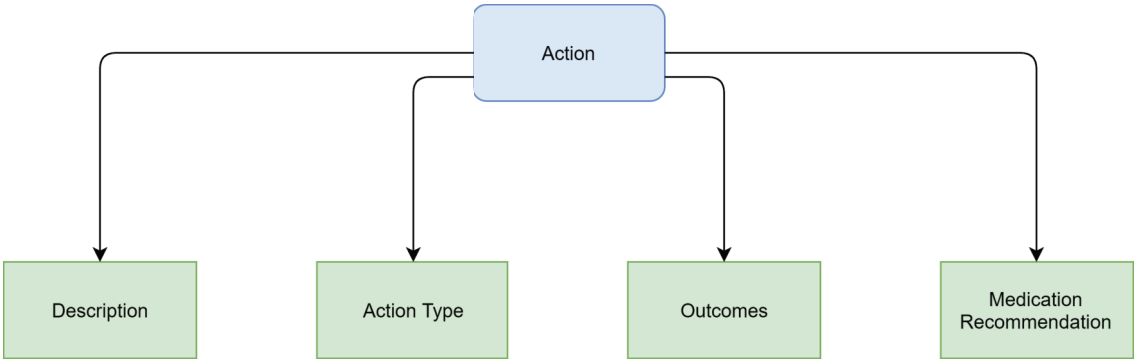



Figure 20.: *Action* Class Parameters

The details of these parameters are as follows:

- **Description** - the description of the action to be performed;
- **Action Type** - identifies the type of action to be performed by a healthcare professional. This parameter includes clinical procedures, clinical examinations, drug recommendations, and non-drug recommendations. It is through this parameter that the interactions between actions can be determined because they consider drug recommendations;
- **Outcomes** - parameter that has a set of conditions that aim to express the expected result of a given task concerning the changes produced in the patient's condition;
- **Medication Recommendation** - through this parameter, it is possible to connect the various types of action to anothers. In this dissertation, only the action type corresponding to the recommended medication is addressed because these advise medicines to treat diseases.

Regarding Outcomes and Medication Recommendation, they also have parameters. In the case of Outcomes, its parameters are shown in Figure 21.

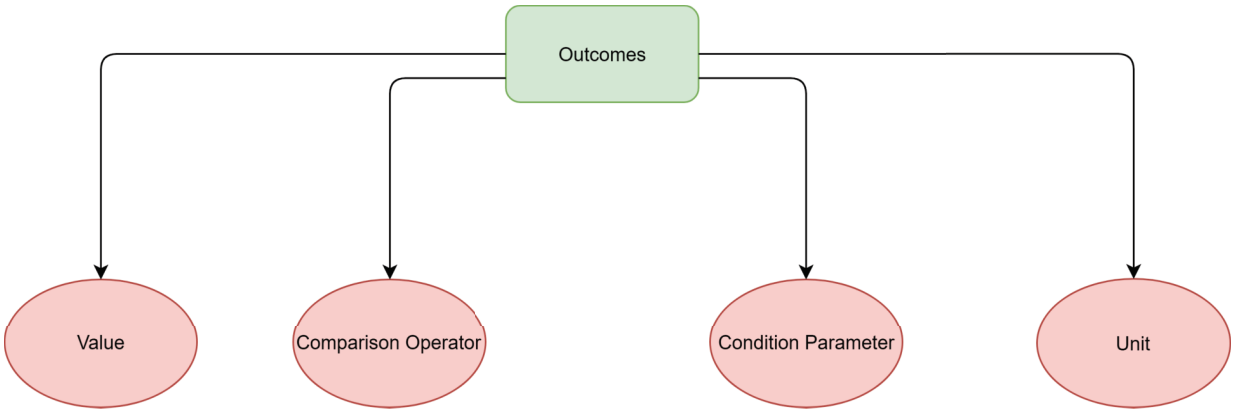



Figure 21.: Outcomes Parameters

The details of these parameters are as follows:

- **Value** - value that aims to quantify the clinical parameter to be compared;
- **Comparison Operator** - includes the various comparison operators: *equal_to*, *greater_than*, *greater_or_equal_than*, *less_than*, *less_or_equal_than*, and *different_from*;
- **Condition Parameter** - what is the clinical parameter to be evaluated (e.g. fever);
- **Unit** - the unit where the expected result of the task should be (e.g. mmol/L).

Regarding the Medication Recommendation parameter, Figure 22 presents its parameters.

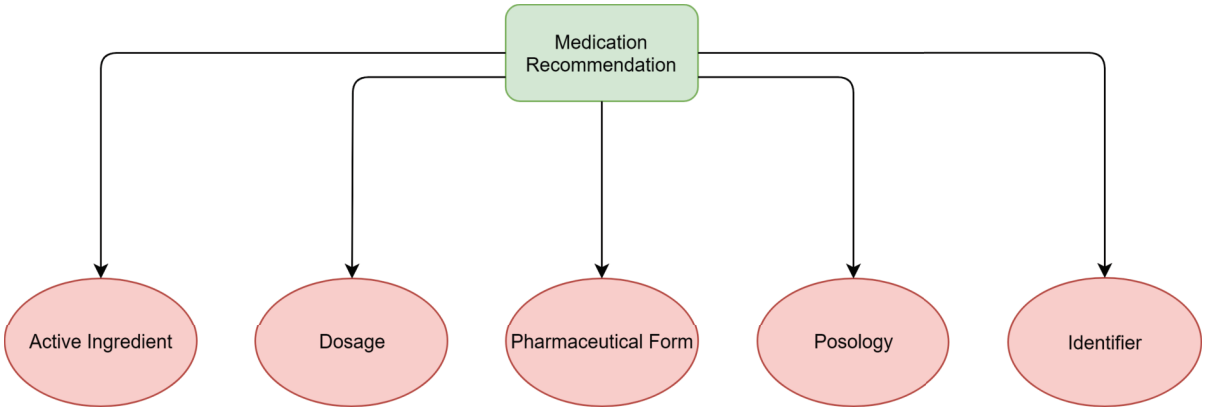



Figure 22.: Medication Recommendation Parameters

The details of Medication Recommendation parameters are as follows:

- **Activation Ingredient** - identifies the component of medication that is responsible for the effects of the medication itself;

- **Dosage** - represents the dosage information of the drug;
- **Pharmaceutical Form** - how the recommended medicine is presented (e.g. injectable, capsule, etc.);
- **Posology** - information on the doses of drugs;
- **Identifier** - the drug identifier.

Based on the properties of the objects, it is possible to define the different control relationships that can exist between tasks, the sequence of task execution, or whether they should be executed simultaneously or in parallel. Regarding these relationships:

- You can identify the first task of a Plan. In Figure 23, we can verify that the instance *Plan* is linked to the instance of the first task, through the *hasFirstTask* property.

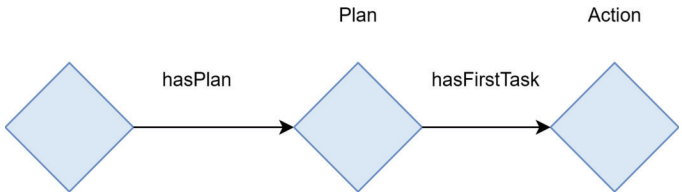



Figure 23.: *HasFirstTask* Property

- If two tasks must occur one after the other, then the first to be executed is bound to the second by the *nextTask* property, thus defining a sequential execution of tasks. This execution is verified in Figure 24.

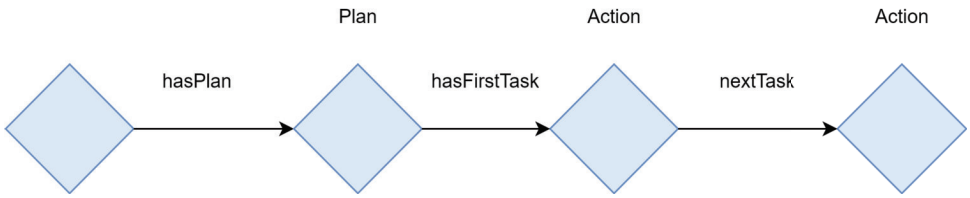



Figure 24.: *NextTask* Property

- If two tasks are to run concurrently, the task preceding them must be linked to them by the parallel task property, as illustrated in Figure 25. This process defines a parallel execution of tasks.

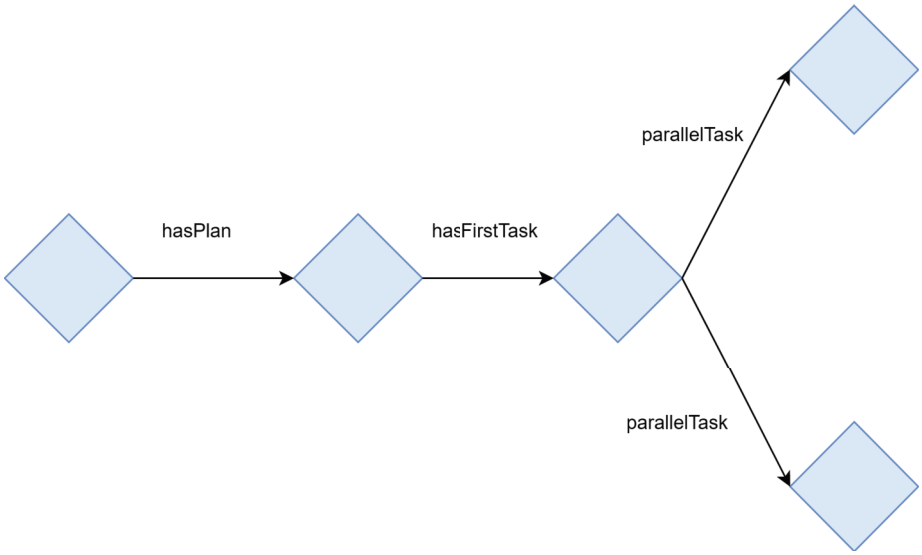



Figure 25.: *ParallelTask* Property

- Based on the patient status you can select a task from a set of alternatives. The task that precedes all alternatives is linked to them by the *alternativeTask* property. However, where the health worker should select the alternative task, the property to use is *preferenceAlternativeTask*. These two options can be seen in Figure 26.

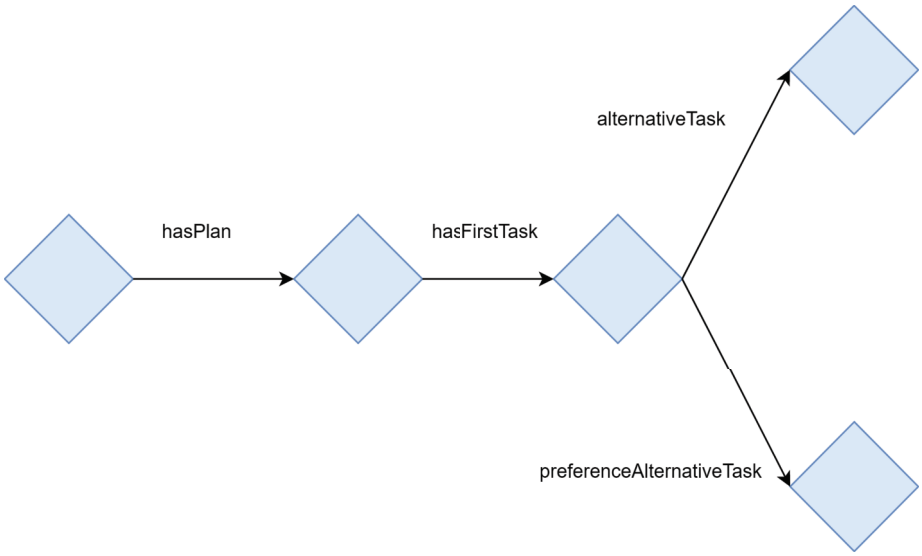



Figure 26.: *AlternativeTask* Property

The CompGuide ontology provides a set of different types of clinical constraints, which are expressed as conditions about the patient's condition. Figure 27 illustrates the three types of constraints provided: *TriggerConditions*, *PreConditions*, and *Outcomes*.

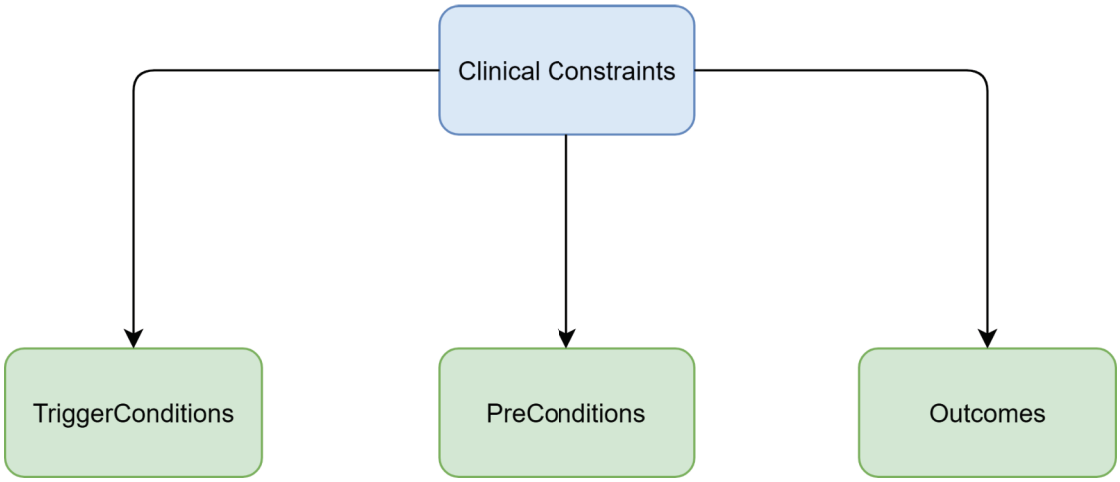



Figure 27.: Clinical Constraints

- **TriggerConditions** - conditions that have an emphasis on the patient's condition. These conditions are expressed in quantitative or qualitative terms, and associated with the execution of alternative tasks, dictating their choice. An alternate task is selected if all your Trigger Conditions are valid.
- **PreConditons** - conditions that have an emphasis on the patient's condition. These conditions are expressed in quantitative or qualitative terms, that define the cases in which a given task can be executed.
- **Outcomes** - conditions that have an emphasis on the patient's condition. These conditions are expressed in quantitative or qualitative terms, which define the objectives of a *Plan* or a *Action*.

## 4.2 COMPGUIDE MODEL FOR CIG DEPLOYMENT

The CompGuide system aims to provide a model that represents and identifies drug-drug interactions, and under this dissertation, the goal is to add an MCDA model that provides an alternative measure to mitigate these interactions.

A mitigation function is used to resolve these conflicts. The purpose of this function is to indicate an alternative recommendation that recommends conflict-free medicines. Different mitigation principles are used to achieve the desired solution, such as patient preferences over clinical recommendations, the similarity between drugs and clinician's priorities over goals.

The architecture of this system is illustrated in Figure 28. As can be seen, there are three well-defined levels in this architecture: representation of CPGs in CIGs, identification of recommendation interactions, and generation of alternative recommendations. This architecture consists of a main component called Core Server. This component was developed as a Representational State Transfer (REST), in this case, RESTful web service application. This type of access to the Core Server enables application performance to

be more consistent in accessing web resources as well as a better scalability. In this way, it is also possible to integrate the functionalities of the execution of CIGs in third party applications.

Core Server has four modules:

- **Authentication Agent**;
- **Database Handler**;
- **Guideline Handler**;
- **Guideline Execution Engine**

The *Authentication Agent* module is responsible for the authentication and authorization of various system users, such as healthcare professionals and administrators.

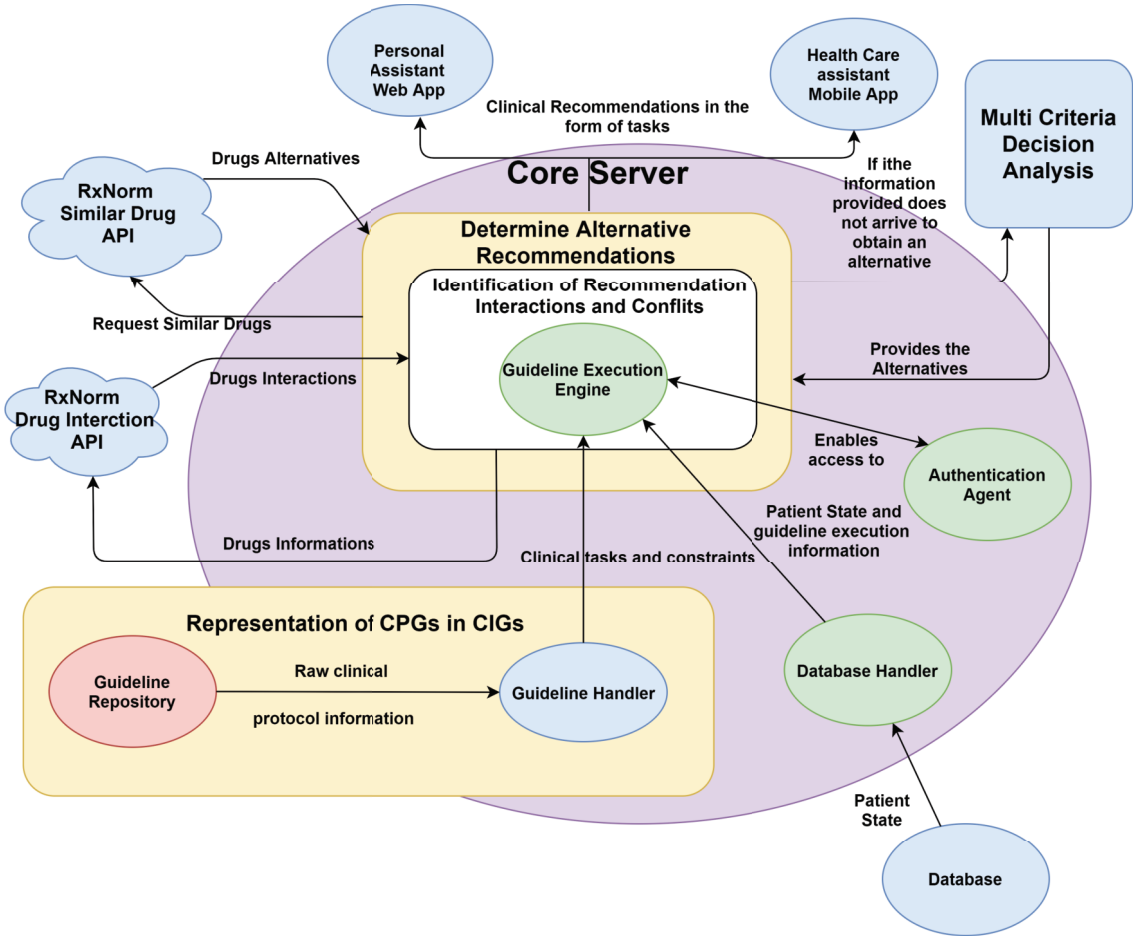



Figure 28.: Architecture of CompGuide System

Concerning the *Database Handler*, it is responsible for providing Guideline Execution Engine with patient status information (from the Database) as well as guideline execution information (e.g. time constraints).

The other Core Server components are described in more detail in the following sections, where each of the three levels in the CompGuide architecture is explained.

Core Server provides the services through a RESTful web service, developed in Java, using the RESTEasy Application Programming Interface (API) on the WildFly application server. The choice came from WildFly because it is an open-source web application server. CompGuide was developed as a web application following the Model-View-Controller (MVC) [72] paradigm.

The use of design patterns, such as MVC, aims to achieve greater scalability by an application, such as greater independence between classes. This independence reduces the impact on changes that are made.

### 4.2.1 *Representation of CPG's in CIGs*

At this level, whose responsibility is the representation of CPGs in CIGs, the ontology mentioned in section 4.1 is used to represent the CPGs in a task network.

The Compguide plugin [73] is used to represent CPGs, indicating how to fill in the data from the guideline entries. The coding output for a guideline is a CIG, that is stored in the *Guideline Repository*. This repository is responsible for maintaining different CIGs defined according to CompGuide ontology. Another component at this level of architecture is the *Guideline Handler*, which has the function of managing the access to the recommendations of a CIG in the *Guidelines Repository*. The *Guideline Handler* provides the clinical tasks and constraints imposed on the respective tasks to the *Guideline Execution Engine* at the following level.

### 4.2.2 *Identification of Recommendation Interactions*

The *Guideline Handler*, as stated, provides all the details of clinical tasks to the *Guideline Execution Engine*, which in turn produces the clinical recommendations. This component produces task execution times, and by calling the RxNorm Interaction API, determines if there are drug-drug interactions.

Using the RxNorm Interaction API makes it possible to determine drug-drug interactions without the necessity of manually define them in the knowledge base. This API has two data sources, ONCHigh and DrugBank, which provide information such as the severity and description of the interaction. *Guideline Execution Engine* processes all tasks that are being performed and for each drug uses the RxNorm Interaction API to obtain the severity and description of the interaction. Table 6 presents the information acquired through this API.

In the CompGuide model, interactions identify the relationships between different recommendations in this case between actions. An interaction results from parallel execution of two recommendations, in which the recommended drugs conflict. Given

Table 6.: Information Acquired from the RxNorm Interaction API.

| Information | Description |
|---|---|
| *RxCui* | The RxNorm Identifier if the drug |
| *Severity* | Identifies the severity of the interaction. If no information on severity is available, then the *N/A* value is assumed. In case there is an adverse interaction, the *high* value is assumed. |
| *Description* | Attribute that describes the severity of the interaction. |
| Source Name | Identifies which source provides the identified interaction information. There are two possible data sources: DrugBank and ONCHigh. |

the information acquired from the RxNorm Interaction API, the Interaction entity is defined by four parameters, as shown in Figure 29.

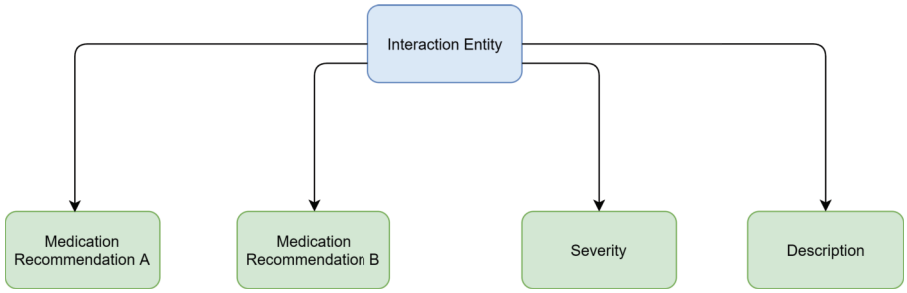



Figure 29.: *Interaction* Entity Parameters

- **Medication Recommendation A** - prescription for drug A corresponding to *Action* A;
- **Medication Recommendation B** - prescription for drug B corresponding to *Action* B;
- **Severity** - the severity of the interaction resulting from the parallel application of medication recommendation A and B;
- **Description** - description of the verified interaction between measurement recommendation A and B.

For a better understanding of the elaborated algorithm, a sequence diagram was developed to improve its comprehension. A sequence diagram is a diagram that aims to describe the sequence of messages exchanged between objects. These type of diagrams describe how objects collaborate on a given behaviour over time. Figure 30 illustrates the algorithm in question. In Appendix C is the other sequence diagram for understanding the algorithm.

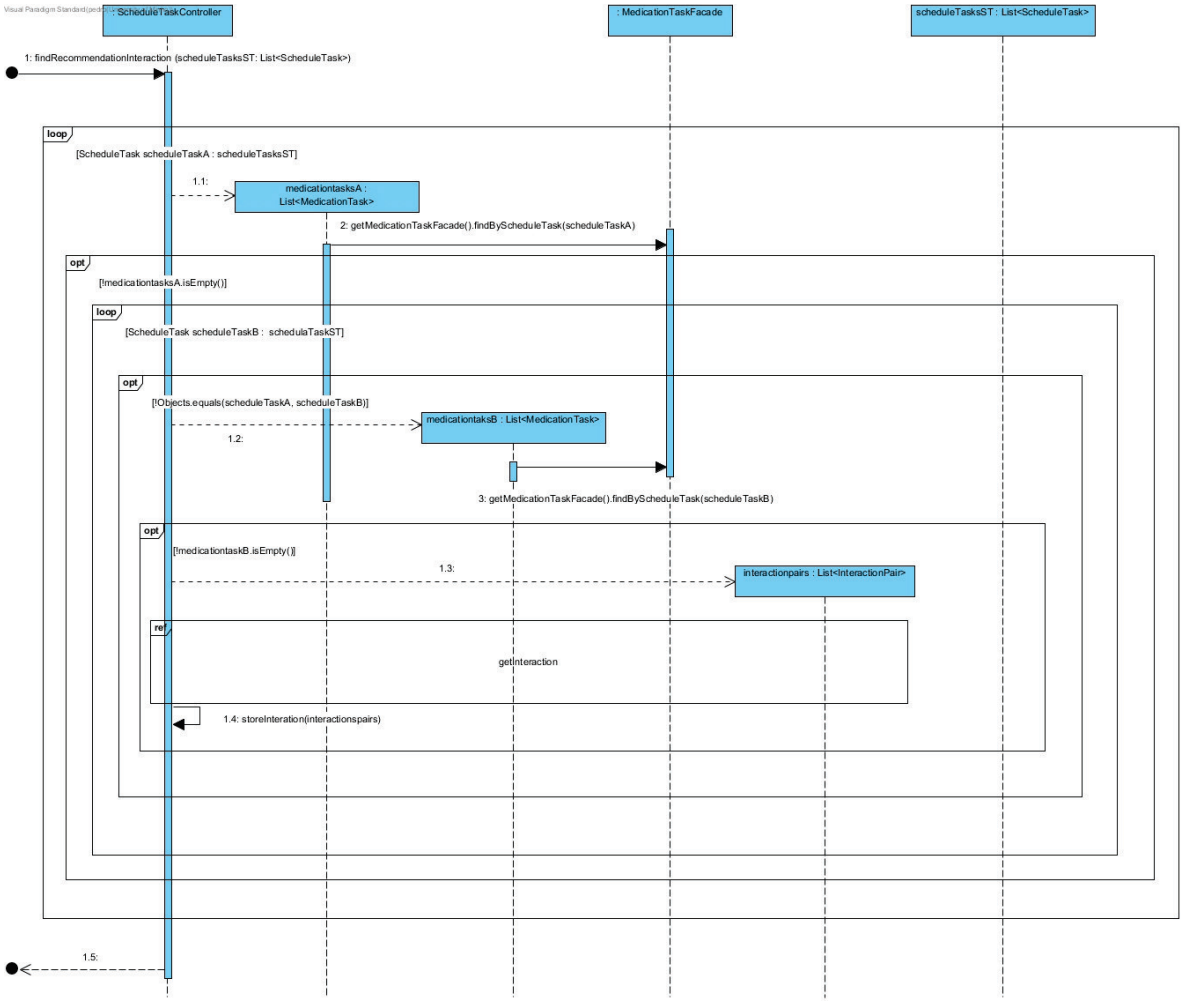



Figure 30.: Interaction Identification Sequence Diagram

### 4.2.3 *Generating Alternative Recomendations*

Finally, the third level of architecture aims to determine alternatives to recommendations. Alternatives are evaluated by the system when the processing of all interactions between clinical recommendations is complete. If there is any adverse drug-drug interaction, the system will automatically attempt to find alternatives that can resolve the conflicts. According to a mitigation function, the system determines the recommended alternatives. This function has a set of steps:

*Step 1: Providing Alternative Recommendations within the Guidelines*

Firstly, the system will check the possibility of obtaining alternative recommendations in the guidelines. For that, Guideline Execution Engine obtains drug recommendations for alternative tasks using the RxNorm Interaction API for each drug paired in the task

and sees if drug interactions exist. If at the end there are no alternative tasks without conflict, the system moves on to step 2, described below.

*Step 2: Providing Alternative Recommendations using RxNorm Similar Drugs API*

If cannot find alternatives in the previous step, at this stage, the system uses the RxNorm Similar API to find conflict-free alternative drugs. For this, a ranking of alternative drugs is produced based on the similarity score provided by the API. Table 7 describes the information that is provided by the API. The similarity score between drugs is a score that determines the similarity between them. The system uses the API to obtain alternative drugs for drug conflicts to calculate the highest similarity score for alternative drugs. For each alternative with the highest score, try to find conflict-free drugs.

Table 7.: Information provided by RxNorm Similar API

| **Information** | **Description** |
|---|---|
| RxCui | Identifier of the similar drug |
| Class Name | Name of the drug |
| Class ID | Class Identifier |
| Equivalence Score | Similarity score between two classes |
| Inclusion Score | Score for finding specific classes that are included in broader classes |
| Drug Source | Which data source (e.g. Drugbank) provides the information on the drug in question. |

The generation of alternatives used by the RxNorm Similar API is illustrated in the sequence diagram of Figure 31. In Appendix C is another diagram that supports the developed algorithm.

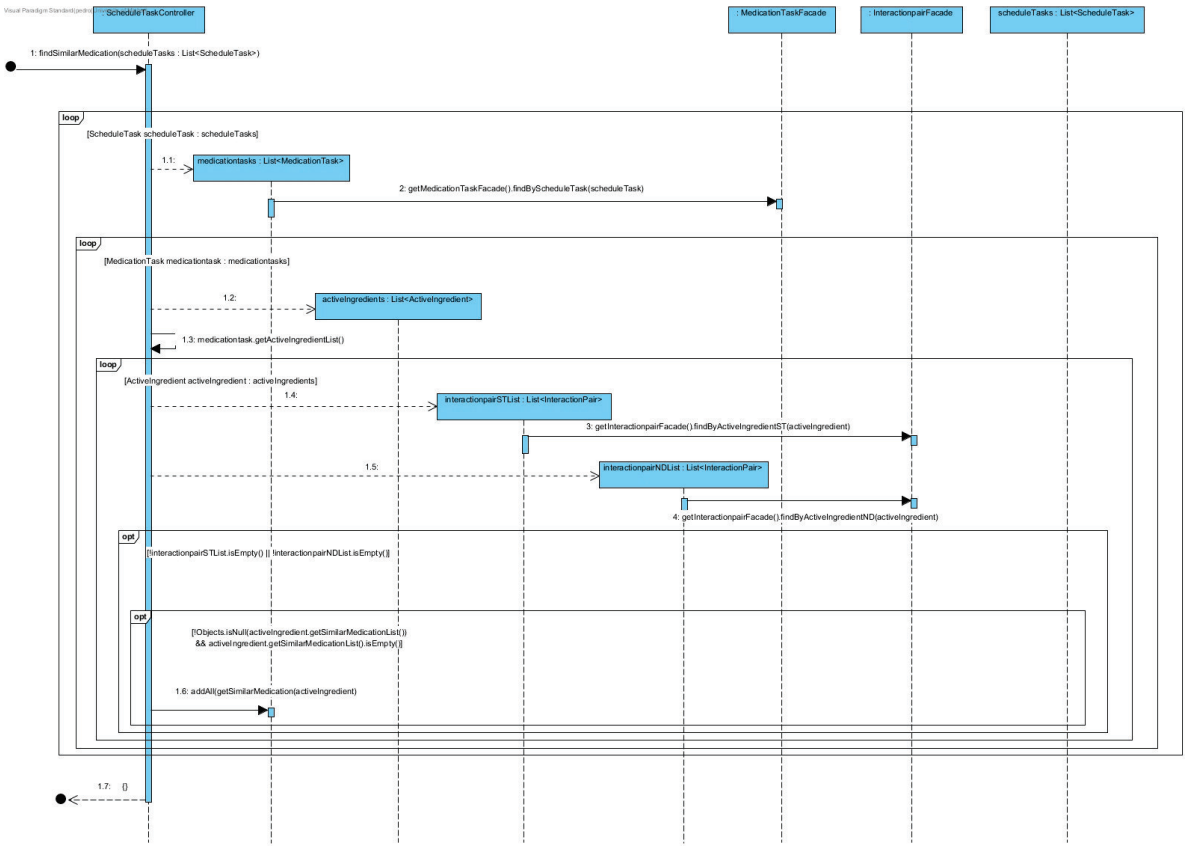



Figure 31.: Similiar Medication Sequence Diagram

In the case of Guideline Execution Engine does not find alternative drugs that do not have conflicts, moves on to step 3.

*Step 3: Multi Criteria Decision Analysis for Clinical Mediation*

Finally, if the other steps fail to provide an alternative, the system evaluates all possible solutions using the MCDA. Because the interactions generate several solutions that are conflicting with each other, it is quite useful to punctuate the solutions. The process of elicit stakeholder preferences on best decision alternatives and criteria should result from a discussion between the patient and the physician and is supported by the system. This approach uses a model, where for each criterion the patient assigns a certain score. The objective is to construct and compare numerical scores to identify the degree to which a particular decision alternative has a greater preference over another. The alternatives to be scored are a combination of drugs so that the system can automatically define the criteria by which decision-makers should orient themselves.

When using an MCDA model, decision-makers have to set preferences within and between criteria through scoring and weighting. In the case of scoring, importance is established using a partial linear function, while in the case of weighting swinging them.

If there is a conflict between two recommendations, such as recommendation A and recommendation B, the solutions that will be evaluated by the MCDA model are:

- Application of recommendation A
- Application of recommendation B
- Application of recommendation A and recommendation B

When the system moves to this stage, the criteria are:

1. **Severity of the Disease for which drugs are advised** - this criterion is obtained through a discussion between the patient and the healthcare professional.
2. **Adverse drug-drug interactions** - criteria obtained through the RxNorm Interaction API.
3. **Expected outcomes for the drug application** - obtained from the Outcome parameter of class Action, as mentioned in section 4.1.

For criteria 1 and 3 mentioned, these are measured in units where higher performance is better, as opposed to criterion 2, where lower performance is better.

For each criterion, it is necessary to assign a score to each alternative. In the end, the performances in each criterion for a given alternative are aggregated to produce an overall value. Therefore, it is possible to compare numerical scores to identify the preferred alternative.

The scores for each criterion are within a given range (e.g. 30-80) determined by the importance of the stakeholders. Then we use linear partial functions, whose purpose is to establish a relationship between the score attributed to each alternative by the stakeholders and the MCDA model score itself, which is defined in a different range between 0 and 100. This function must consider whether the variation along the defined range is linear or not and for each criterion which performance is better, whether the higher or lower performance.

The linear partial functions have the following expression:

$$y = mx + b \tag{1}$$

As can be seen, this expression is the reduced equation of the line, where:

- **m** - slope of the straight
- **x and y** - coordinates of a point belonging to the line
- **b** - linear coefficient

For the calculation of the slope of the line two points belonging to it are necessary, whose formula is the following expression:

$$m = \frac{y_2 - y_1}{x_2 - x1} \tag{2}$$

Once MCDA is a known value that ranges between 0 and 100, and with the ranges set by the stakeholders, we thus two necessary points.

Having the Range (x, y), where x is the defined minimum value and y the maximum value, in the case of criterion 1 and 3 where higher performance is better, the two points for the slope calculation would have the following form:

- A($Range_x$,0);
- B($Range_y$,100).

With the $x$ coordinate being the minimum and maximum value in the range defined by the interveners, and the $y$ coordinate being the minimum and maximum value of the MCDA model score.

Regarding the case where the lower performance is better, the points taken as an example are:

- A($Range_y$,0);
- B($Range_x$, 100).

The difference, in this case, is in the $x$ coordinates of the two points, where in the first point (A) the maximum value defined by the actors is identified, and in the second point (B) the minimum value defined.

Regarding the calculation of the linear coefficient, it is based on the expression:

$$b = y - mx \tag{3}$$

As an example, with the points defined above for the case where the higher performance is better, substituting the values in the expression 3, we get a value of $x$. In the case where the lower performance is better, using the same points exemplified, the coefficient value is $y$.

The first summary of an MCDA process is presented by developing a performance matrix. This table demonstrates the importance given by the patient to alternatives in each criterion. Table 8 illustrates a performance matrix of an MCDA model.

Table 8.: Performance Matrix

| **Criteria** | $Alternative^1$ | ... | $Alternative^m$ |
|---|---|---|---|
| $C_1$ | $Choose\_Value_1^1$ | ... | $Choose\_Value_1^m$ |
| ... | ... | ... | ... |
| $C_n$ | $Choose\_Value_n^1$ | ... | $Choose\_Value_n^m$ |

where:

- **n** - number of criterion;

- **m** - number of alternative.

To support the development of the performance matrix is illustrated in Figure 32, the respective sequence diagram.

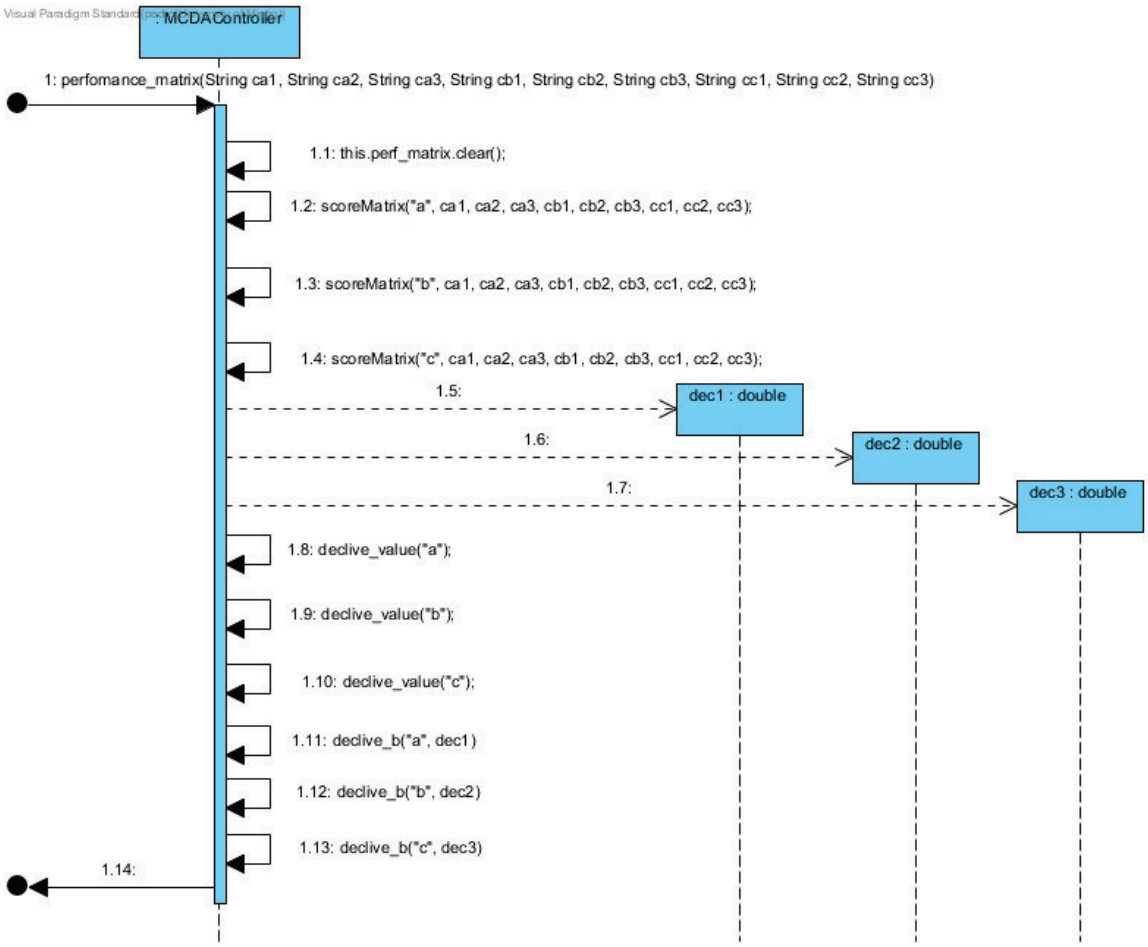



Figure 32.: Performance Matrix Sequence Diagram

The next step in the MCDA model is defining importance for the cause criteria. This importance allows producing "total values", which comes from partial value scores, with the application of weights.

To achieve the "total values" a weighting process is performed, to compare one criterion from the others. This weight is in the range of 0 to 100 to determine the importance of a criterion.

Then it is necessary to normalise the values by dividing it by the sum of the importance given to the criteria. The weight of each criterion is defined by equation 4.

$$WeightCriteria(n) = \frac{P^n}{\sum_{n=1}^{n} p^n}, \tag{4}$$

where P defines how much importance is given to a criterion $n$.

Finally, a score must be calculated for each of the alternatives considered. This value is calculated through an aggregation method and by the use of an additive model. The score is defined by the sum of all criteria multiplied by the attributed weight with the MCDA score previously determined in each alternative.

$$f(n) = \sum_{n=1}^{n} S^n * WeightCriteria^n, \tag{5}$$

where $n$ is the total number of solutions to be scored, $S^n$ is the MCDA score specific to the solution in question, and *WeightCriteria* the weight of the respective criterion to be evaluated.

The aggregation method using the additive model is presented in Table 9, with all the formulas inherent in each cell. This table is available in the Personal Assistant Web App (CompGuide), with the preferred alternative information, using the MCDA model.

Table 9.: MCDA Model Summary

| **Criterion ($\alpha$)** | **Score $A^1$** | **...** | **Score $A^n$** | **Weigths** | **Final Score $A^1$** | **...** | **Final Score $A^n$** |
|---|---|---|---|---|---|---|---|
| $\alpha^1$ | $S_1^{\alpha^1}$ | ... | $S_n^{\alpha^1}$ | $\frac{p^1}{\sum_{n=1}^{n} p^n}$ | $S^1 * WeightCriteria(1)$ | ... | $S^n * WeightCriteria(n)$ |
| ... | ... | ... | ... | ... | ... | ... | ... |
| $\alpha^n$ | $S_1^{\alpha^n}$ | ... | $S_n^{\alpha^n}$ | $\frac{p^n}{\sum_{n=1}^{n} p^n}$ | $S_1^{\alpha^n} * WeightCriteria(1)$ | ... | $S^{\alpha_n^n} * WeightCriteria(n)$ |
| **Total Values** | | | | | $\sum_{n=1}^{n} S^n * WeightCriteria^n$ | ... | $\sum_{n=1}^{n} S^n * WeightCriteria^n$ |

where:

- $\alpha_n$ : $\alpha$ matches a criterion, $n$ the criterion number;
- $S_n^{\alpha^n}$ : S means the score of a given criterion ($\alpha^n$) for a specific alternative;
- $A^n$ - corresponds to the alternative, where $n$ is the alternative number.

The sequence diagram illustrating the development of the MCDA model is shown in Figure 33.

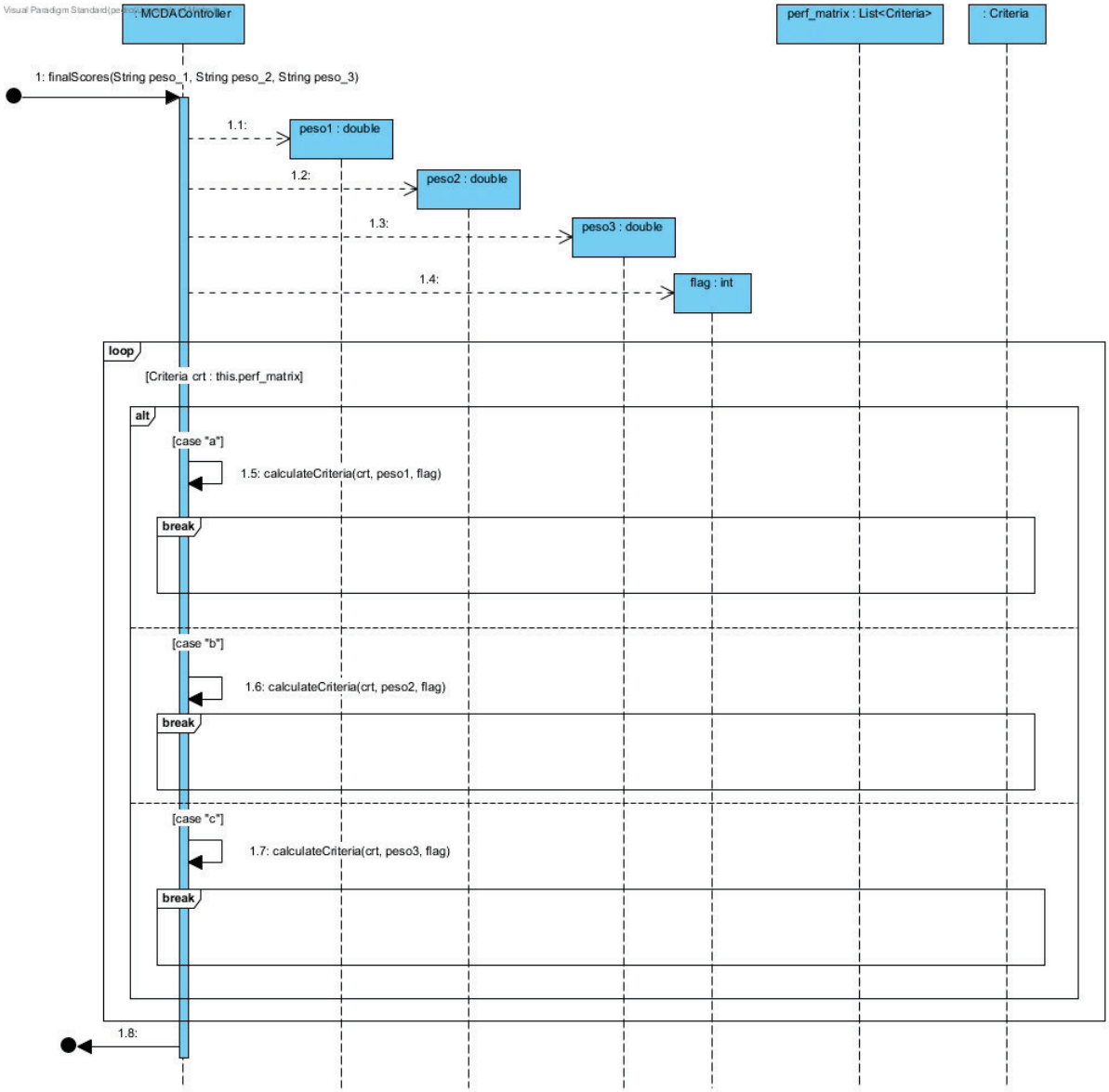



Figure 33.: MCDA Model Final Score Sequence Diagram

# 5

# CASE STUDIES / EXPERIMENTS

This chapter describes the development of the case study used as a solution to the problem of this dissertation. Section 5.1 identifies the technologies used in the development of this dissertation, as well as a brief description of them.

Section 5.2 presents the case study used, describing the recommendations, the alternatives for the MCDA model, as well as the entire procedure of this process, with the presentation of the values of each step performed. Also, at the end of the section is presented the table summarising the MCDA process of this case study, as well as the identification of the preferred alternative. Finally, section 5.3 provides the different interfaces of the MCDA model, taking into account the case study presented.

## 5.1 EXPERIMENTS SETUP

To develop a solution to the problem of this dissertation, it was necessary to use several technologies. At the user level, the solution interface would have to be as intuitive and appellative as possible. Figure 34 illustrates the technologies used in solution development, in the server and interface side.

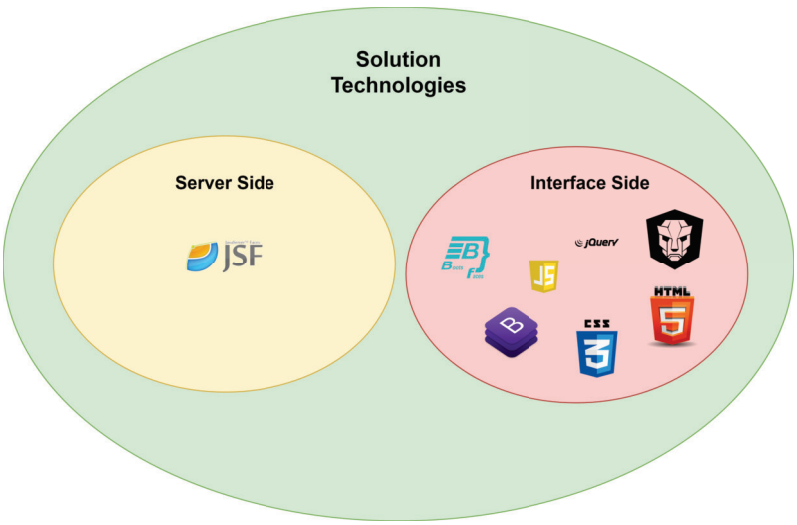



Figure 34.: Technologies used for the solution of the problem

- ***Java Server Faces* (JSF)** - the technology used for web development, specifically the development of the logic of the proposed solution;
- ***HyperText Markup Language 5 (HTML5)*** - solution web page development;
- ***Cascading Style Sheets 3 (CSS3)*** - for defining web page styles developed;
- ***BootStrap*** - a framework that helps in developing responsive web pages and mobile applications, making them look attractive;
- ***BootsFaces*** - the powerful JSF framework which allows the use of BootStrap and jQuery, to facilitate the development of the solution's web page;
- ***PrimeFaces*** - the technology used to enable the use of widgets that allow more intuitive user interaction with the developed solution;
- ***JavaScript*** - the framework that allows the use of dynamic content in the solution, also enriching the interfaces;
- ***jQuery*** - this technology provides interactions, widgets and effects in creating more interactive applications.

## 5.2 RESULTS

This section presents the case studies developed. This case study aims to resolve the conflict adjacent to the interaction between recommendations using an MCDA model. For this, two CIGs were used, the first based on the NCCN Clinical Practice Guideline for Prostate Cancer, and the second based on the IDF Clinical Practice Recommendations for managing Type 2 Diabetes. These two CIGs were represented in CompGuide ontology using the *CompGuide plugin* [73] mentioned in section 4.2.1.

In this example case, two recommendations were considered, one of each guideline mentioned:

- Recommendation 1 belongs to the guideline for managing Type 2 Diabetes;
- Recommendation 2 belongs to the guideline for prostate cancer.

Table 10 gives a brief description of each recommendation.

Table 10.: Description of Recommendations

| Recommendation | Descriptiom |
|---|---|
| 1 | Apply insulin 0.2 units/kg and titrate once weekly at one unit each time during six months to achieve a target fasting blood glucose between 3.9 and 7.2 mmol/L (70 and 130 mg/dL) |
| 2 | Apply leuprolide 180 mg/m2 as part of Androgen Deprivation Therapy |

As mentioned in section 4.1, recommendations are mapped to CompGuide Ontology. These recommendations are *Action* type, therefore having four parameters: *Description*, *Action Type*, *Outcome* and *Recommendation Medication*. For recommendation 1, table 11 describes each of these parameters.

Table 11.: Description of recommendation 1 parameters

| Parameter | Description |
|---|---|
| *Description* | Apply insulin |
| *Action Type* | Medication recommendation |
| *Outcome* | **Value**: 3.9 and 7.2<br>**Comparison operator**: greater than and less than<br>**Condition parameter**: blood glucose<br>**Unit**: mmol/L |
| *Recommendation Medication* | **Active Ingredient**: insulin<br>**Dosage** : 0.2 units/Kg<br>**Pharmaceutical Form**: N/A<br>**Posology**: Insulin 0.2 units/kg give once weekly at 1 unit each time during 6 months<br>**Identifier**: N/A |

Regarding recommendation 2, the description of the respective parameters is given by table 12.

Table 12.: Description of recommendation 2 parameters

| **Parameter** | **Description** |
|---|---|
| *Description* | Apply leuprolide |
| *Action Type* | Medication recommendation |
| *Outcome* | **Value**: 50<br>**Comparison operator**: less or equal than<br>**Condition parameter**: serum testosterone<br>**Unit**: ng/dL |
| *Recommendation Medication* | **Active Ingredient**: leuprolide<br>**Dosage**: 180mg/m2<br>**Pharmaceutical Form**: N/A<br>**Posology**: leuprolide 180mg/m2 given<br>**Identifier**: N/A |

When applying these two recommendations, there is a drug conflict, namely, the drug leuprolide harms the therapeutic efficacy of insulin. This interaction between them was obtained using the *RXNorm Interaction API* described in section 4.2.2.

The alternatives corresponding to this case study are:

- **Alternative 1** - Only apply insulin;
- **Alternative 2** - Only apply leuprolide;
- **Alternative 3** - Apply insulin (alternative 1) and leuprolide (alternative 2) at the same time.

As mentioned, in section 4.2.3 it is necessary to select the criteria for the MCDA model. For this case study the following criteria were selected:

- **Criterion A** - Severity of disease for which drugs are advised;
- **Criterion B** - Adverse drug-drug interactions;
- **Criterion C** - Expected outcomes for the drug application.

After presenting the selection criteria and the alternatives, the health professional must define a determined interval for each evaluation criterion in the MCDA model. Table 13 represents the intervals selected in this case study.

Table 13.: The ranges for each criterion

| **Criteria** | **Range** |
|---|---|
| Criterion A | 80-100 |
| Criterion B | 70-90 |
| Criterion C | 30-50 |

To calculate the score that will be the input of the MCDA model it was necessary to define importance values for each alternative in the three defined criteria. Table 14 shows the values chosen in this example.

Table 14.: Performance matrix for this case study

| Criteria | Alternative 1 | Alternative 2 | Alternative 3 |
|---|---|---|---|
| A | 85 | 90 | 95 |
| B | 70 | 70 | 75 |
| C | 34 | 38 | 43 |

The table shown above is defined as the performance matrix in an MCDA model, where you can see the importance that users attach to each alternative in each criterion. It is necessary to define the partial linear functions to relate the performance matrix scores with the MCDA input scores. The linear partial functions defined for each criterion were as follows:

- **Criterion A** : $y = 5x - 400$
- **Criterion B** : $y = -5x + 450$
- **Criterion C** : $y = 5x - 150$

Using these functions, the scores for the MCDA model input are shown in Table 15.

Table 15.: The score for input to MCDA model

| Criterion | Alternative 1 (score pacient/score MCDA) | Alternative 2 (score pacient/score MCDA) | Alternative 3 (score pacient/score MCDA) |
|---|---|---|---|
| A | 85/25 | 90/50 | 95/75 |
| B | 70/100 | 70/100 | 75/75 |
| C | 34/20 | 38/40 | 43/65 |

In the table, it is possible to verify two values, on the left side of the slash the values of the performance matrix, and on the right side, the values corresponding to the partial linear function defined for each criterion.

The next step in this case study is to get the weights associated with each criterion. The defined importance of each criterion was as follows:

- Importance for criterion A - 80 points;
- Importance for criterion B - 100 points;
- Importance for criterion C - 60 points.

Then, it was necessary to normalise these values to use in the MCDA process. This normalisation process uses equation 4. Table 16 shows the normalised weight values for each criterion.

Table 16.: Weight of each criterion

| Criterion | Weight |
|---|---|
| A | 0.33 |
| B | 0.41 |
| C | 0.25 |

Table 17.: MCDA Model Case Study Summary

| Criteria | Scores Alt.1 | Scores Alt.2 | Scores Alt.3 | Weights | Alt.1 | Alt.2 | Alt.3 |
|---|---|---|---|---|---|---|---|
| A | 25 | 50 | 75 | 0.33 | 8.25 | 16.5 | 24.75 |
| B | 100 | 100 | 75 | 0.41 | 41.0 | 41.0 | 30.74 |
| C | 20 | 40 | 65 | 0.25 | 5.0 | 10.0 | 16.25 |
| Total Value | | | | | 54.25 | 67.5 | 71.74 |

Finally, the total number of values to be used, by the additive model, are produced. To summarise this whole process, Table 17 presents the MCDA process scores, weights and total values, for this example case.

The table above provides a ranking of alternatives, where alternative 3 (with a total of 71.74 points) is the preferred solution in this particular case example, followed by alternative 2 (67.5 points) and alternative 1 (54.25 points).

## 5.3 WEB APPLICATION

This section presents the CompGuide web application, with the implementation of the proposed solution.

Figure 35 represents the timeline of the various tasks. Tasks A02 and A061 represent the application of insulin and leuprolide, respectively.

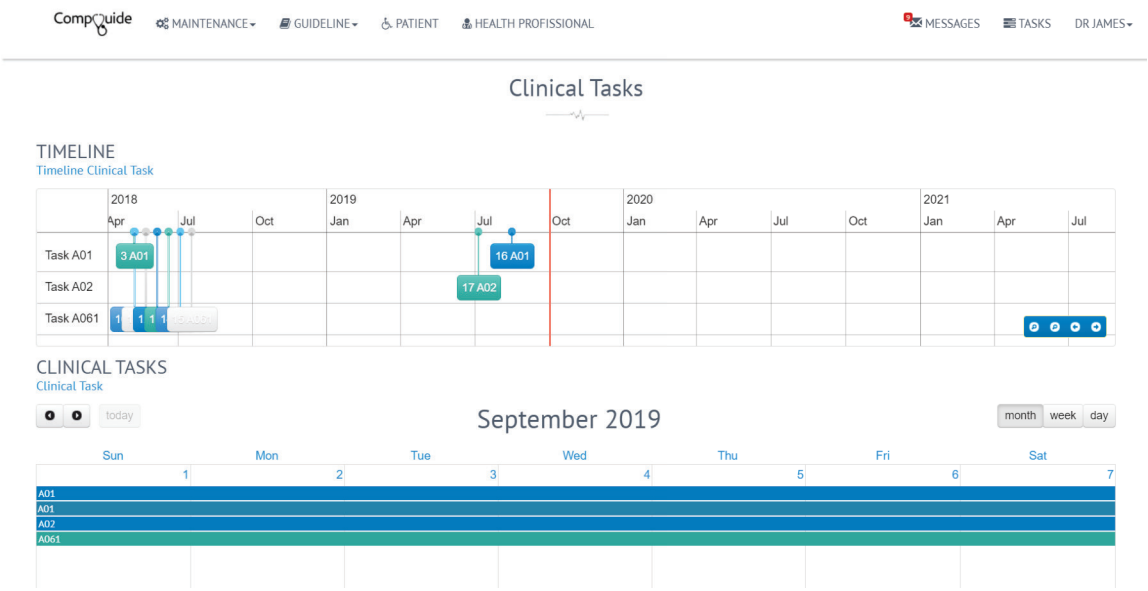



Figure 35.: Task Timeline and Scheduling

After defining the ranges for each criterion, the patient and the healthcare professional must define the importance of each alternative for each criterion. Figure 36 represents the choice of this importance.

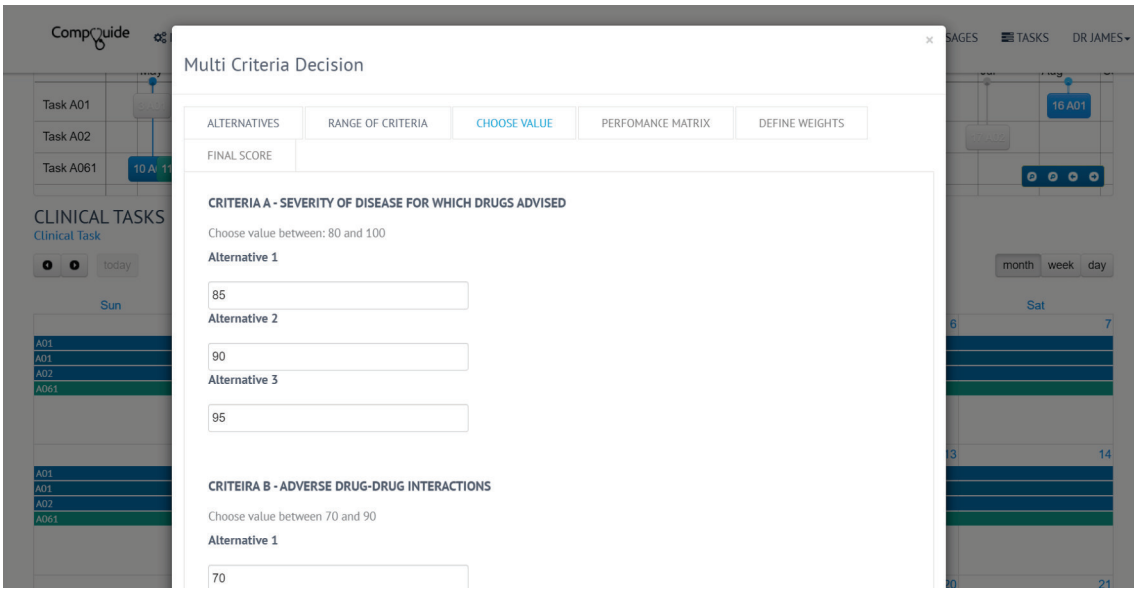



Figure 36.: Importance of alternatives in each criterion

Then, it is necessary to choose the importance of each criterion between the values 0 and 100, as shown in Figure 37.

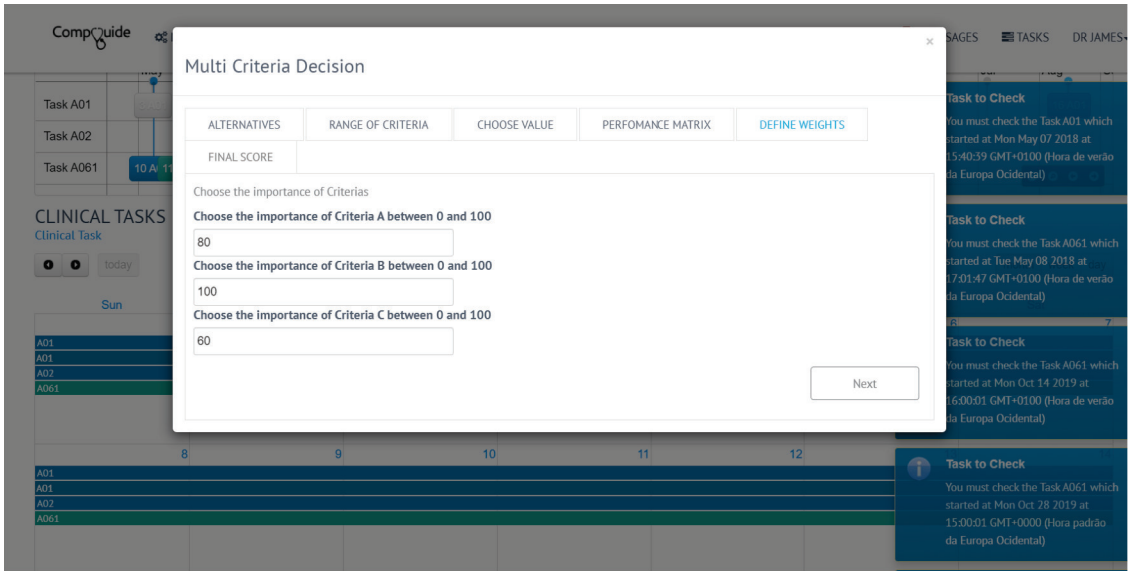



Figure 37.: Choice of importance of each criterion

Finally, Figure 38 depicts a table with the results of the MCDA process with the total and partial scores of each alternative, and which alternative is "chosen" by this process. This figure represents output for the case study referred in section 5.2.

The remaining figures of the web application can be found in Appendix D.

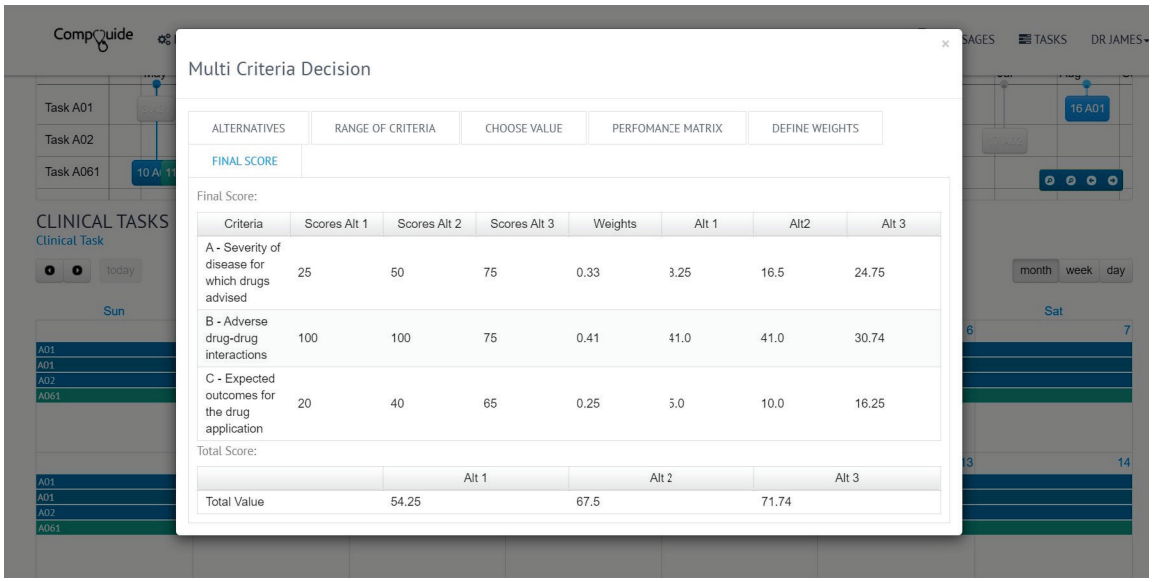



Figure 38.: MCDA Model Summary

# 6

# CONCLUSION

This chapter describes the conclusions of this work, highlighting the objectives achieved, in section 6.1. In section 6.2, major limitations found throughout the dissertation are identified, as well as the definition of future work.

## 6.1 CONCLUSIONS

The study of CPGs presents itself as a very challenging and interesting area. Some solutions provide tools for creating and executing CPGs. However, they are focused on the academic field by either lacking in functionalities such as scheduling and temporal management of clinical protocols, the combination of clinical protocols and drug conflict resolution. Many of these solutions are unintuitive tools to use, whether for creating or editing CIGs. The low intuitiveness of the solutions makes them unsuitable for use in healthcare facilities.

An important aspect in the development of CDSSs is the support that these systems can give to health professionals, which are subject to stressful situations, responsible for medical errors, variations in clinical practice, and practice of defensive medicine. This shows that it is necessary to approach health professionals with good clinical practice and evidence-based medicine, by giving some assistance in the decision-making with the help of computer science.

Many of the projects studied in the literature can automatically identify and resolve the conflicts between guidelines. The alternatives provided for conflict resolution are made through the use of external sources such as DrugBank. The problem evidenced in these solutions is when the external sources used do not have information about alternatives to use, stagnating the process in this phase. When this happens, these solutions fail to provide a process that assesses alternatives outside sources as well as priorities and may lead to a lack of predictability as to the importance of a particular factor or criterion in decision making.

The CompGuide system, with the integration of the MCDA model for conflict resolution, allows solving the above problem, adding the patients perspective/opinion/choice regarding a set of alternatives and criteria. From identifying drug interactions and conflicts automatically, or providing alternative measures to resolve the existing conflicts,

all solutions are evaluated to provide a response that matches the objectives of all parties involved in the decision-making process. Using an MCDA model, it is possible to assess the risk of applying clinical recommendations as well as the possibility of gaining patients' preferences over the various treatment alternatives.

Regarding the objectives of this dissertation, it is considered that they were satisfactorily achieved. Also, the analysis of the problem in question is carried out. Existing approaches to the problem were studied, highlighting in each case the existing limitations so that the solution presented would be an added value in a conflict resolution process.

Using the RxNorm Interaction API it was possible to automatically identify conflicts or interactions that might exist when multiple CPGs are running. The development of a solution to the identified problem was also achieved by drafting a proposal using an MCDA model when there was no information from outside sources to resolve the conflict. The use of this template is only as a third step of the conflict resolution process, after using the guideline itself and using the RxNorm Similar API.

## 6.2 Limitations and prospect for future work

During the work developed to this dissertation, the greatest difficulties were the development of basic knowledge in the clinical field due to the complexity of some concepts. Being this dissertation developed in the field of informatics, the clinical domain was outside of the scope of knowledge. Moreover, it was necessary to conduct research not only about the problem domain but also to understand the impact of CPs on the current clinical practice.

One limitation that can be pointed out is related to the society in which we are included, in this case, the Portuguese one. Due to the social culture of Portuguese society, elderly people commonly aren't included in the decision-making process for choosing the best alternative treatments. As mentioned in section 4.2.3, it is important to include patients preferences since treatment plans can have harmful effects on the patient's health, altering the habits of life and the quality of life. In contrast, the younger generation is opening to know if there is any kind of alternative to a specific treatment, such as knowing the side effects that a particular drug may have.

Because our system requires input from both patients and care professionals, a more integrated approach is needed. In other words, it requires availability from patients and doctors to discuss health issues and potential treatments.

In my opinion, this process helps both parties to clarify their doubts, to know the alternatives and the inherent risks. Moreover, in our proposed approach, the patients play an active role in a process where he will be the main target.

From a future work perspective, the use of machine learning algorithms to predict potential conflicts between drug recommendations, based on some attributes such as

therapeutic, genomic properties, etc. This type of prediction could anticipate existing interactions between the recommendations and thus be able to avoid any negative effect on the patient's health.

Another step to be taken in future development will be the assessment of the functionality developed for conflict resolution in concrete terms, i.e., by conducting a study in which various health professionals interact with the system in a clinical setting. With this study, it would be possible to verify if the system meets the needs of health professionals, so that it can be inserted into clinical practice.

# BIBLIOGRAPHY


[1] K. Andreev, V. Kantorova, and J. Bongaarts, "Demographic components of future population growth," *Technical Paper*, vol. 3, 2013.

[2] V. Della Mea, "What is e-health (2): The death of telemedicine?," *Journal of Medical Internet Research*, vol. 3, no. 2, pp. 6–7, 2001.

[3] E. Rich, K. Knight, and M. Ratto, *Inteligência artificial*. Makron Books, 2 ed., 1988.

[4] G. F. Luger, *Inteligência Artificial - Estruturas e estratégias para a solução de problemas complexos*. Bookman, 4 ed., 2004.

[5] E. S. Berner and T. J. Lande, *Clinical Decision Support Systems: Theory and Practice*. Springer Science & Business Media, 2007.

[6] A. Ramesh, C. Kambhampati, J. R. Monson, and P. Drew, "Artificial intelligence in medicine.," *Annals of The Royal College of Surgeons of England*, vol. 86, no. 5, p. 334, 2004.

[7] E. Coiera, *Guide to health informatics*. CRC press, 2015.

[8] C. P. Landrigan, J. M. Rothschild, J. W. Cronin, R. Kaushal, E. Burdick, J. T. Katz, C. M. Lilly, P. H. Stone, S. W. Lockley, D. W. Bates, *et al.*, "Effect of reducing interns' work hours on serious medical errors in intensive care units," *New England Journal of Medicine*, vol. 351, no. 18, pp. 1838–1848, 2004.

[9] J. L. Halbach and L. Sullivan, "Medical errors and patient safety: a curriculum guide for teaching medical students and family practice residents," *New York: New York Medical College*, 2003.

[10] R. A. Miller, M. A. McNeil, S. M. Challinor, F. E. Masarie Jr, and J. D. Myers, "The internist-1/quick medical reference project—status report," *Western Journal of Medicine*, vol. 145, no. 6, p. 816, 1986.

[11] G. M. Marakas, *Decision support systems in the 21st century*, vol. 134. Prentice Hall Upper Saddle River, NJ, 2003.

[12] D. F. Sittig, A. Wright, J. A. Osheroff, B. Middleton, J. M. Teich, J. S. Ash, E. Campbell, and D. W. Bates, "Grand challenges in clinical decision support," *Journal of biomedical informatics*, vol. 41, no. 2, pp. 387–392, 2008.

[13] R. A. Greenes, *Clinical decision support: the road ahead*. Elsevier, 2011.

[14] J. I. Westbrook, M. T. Baysari, L. Li, R. Burke, K. L. Richardson, and R. O. Day, "The safety of electronic prescribing: manifestations, mechanisms, and rates of system-related errors associated with two commercial systems in hospitals," *Journal of the American Medical Informatics Association*, vol. 20, no. 6, pp. 1159–1167, 2013.

[15] S. Abu-Naser, H. El-Hissi, M. Abu-Rass, and N. El-Khozondar, "An expert system for endocrine diagnosis and treatments using jess," *Journal of Artificial Intelligence*, vol. 3, no. 4, pp. 239–251, 2010.

[16] S. A. Naser, R. Al-Dahdooh, A. Mushtaha, and M. El-Naffar, "Knowledge management in esmda: expert system for medical diagnostic assistance," *AIML Journal*, vol. 10, no. 1, pp. 31–40, 2010.

[17] C. W. Turner, J. Williamson, M. J. Lincoln, P. J. Haug, J. Buchanan, C. Anderson, M. Grant, R. Cundick, and H. Warner, "The effects of iliad on medical student problem solving.," in *Proceedings. Symposium on Computer Applications in Medical Care*, pp. 478–482, American Medical Informatics Association, 1990.

[18] S. Vachhrajani, A. V. Kulkarni, and J. R. W. Kestle, "Clinical practice guidelines," *Journal of Neurosurgery: Pediatrics*, vol. 3, no. 4, pp. 249–256, 2009.

[19] S. Farooq *et al.*, "Clinical protocols: introduction to a useful strategy in clinical practice.," *JPMA. The Journal of the Pakistan Medical Association*, vol. 50, no. 10, pp. 354–357, 2000.

[20] S. Reis and S. Cardoso, "Multimorbilidade em cuidados de saude primarios: o que ha de novo?," *Revista Portuguesa de Medicina Geral e Familiar*, vol. 31, no. 3, pp. 230–231, 2015.

[21] M. E. Tinetti, S. T. Bogardus Jr, and J. V. Agostini, "Potential pitfalls of disease-specific guidelines for patients with multiple conditions," *The New England journal of medicine*, vol. 351, no. 27, p. 2870, 2004.

[22] C. M. Boyd, J. Darer, C. Boult, L. P. Fried, L. Boult, and A. W. Wu, "Clinical practice guidelines and quality of care for older patients with multiple comorbid diseases: implications for pay for performance," *Jama*, vol. 294, no. 6, pp. 716–724, 2005.

[23] S. H. van Oostrom, H. S. J. Picavet, S. R. de Bruin, I. Stirbu, J. C. Korevaar, F. G. Schellevis, and C. A. Baan, "Multimorbidity of chronic diseases and health care utilization in general practice," *BMC family practice*, vol. 15, no. 1, p. 61, 2014.

[24] C. Salisbury, L. Johnson, S. Purdy, J. M. Valderas, and A. A. Montgomery, "Epidemiology and impact of multimorbidity in primary care: a retrospective cohort study," *Br J Gen Pract*, vol. 61, no. 582, pp. e12–e21, 2011.

[25] S. K. Cho, C. O. Kim, E. S. Park, and J.-Y. Chung, "Verapamil decreases the glucose-lowering effect of metformin in healthy volunteers," *British journal of clinical pharmacology*, vol. 78, no. 6, pp. 1426–1432, 2014.

[26] N. Mulyar, W. M. Van der Aalst, and M. Peleg, "A pattern-based analysis of clinical computer-interpretable guideline modeling languages," *Journal of the American Medical Informatics Association*, vol. 14, no. 6, pp. 781–787, 2007.

[27] F. Sonnenberg and C. Hagerty, "Computer-interpretable clinical practice guidelines," *Where are we and where are we going*, pp. 145–158, 2006.

[28] B. Martínez-Salvador and M. Marcos, "Supporting the refinement of clinical process models to computer-interpretable guideline models," *Business & Information Systems Engineering*, vol. 58, no. 5, pp. 355–366, 2016.

[29] S. H. Woolf, R. Grol, A. Hutchinson, M. Eccles, and J. Grimshaw, "Potential benefits, limitations, and harms of clinical guidelines," *Bmj*, vol. 318, no. 7182, pp. 527–530, 1999.

[30] D. Isern and A. Moreno, "Computer-based execution of clinical guidelines: a review," *International journal of medical informatics*, vol. 77, no. 12, pp. 787–808, 2008.

[31] A. A. Boxwala, M. Peleg, S. Tu, O. Ogunyemi, Q. T. Zeng, D. Wang, V. L. Patel, R. A. Greenes, and E. H. Shortliffe, "Glif3: a representation format for sharable computer-interpretable clinical practice guidelines," *Journal of biomedical informatics*, vol. 37, no. 3, pp. 147–161, 2004.

[32] K.-P. Adlassnig, C. Combi, A. K. Das, E. T. Keravnou, and G. Pozzi, "Temporal representation and reasoning in medicine: Research directions and challenges," *Artificial intelligence in medicine*, vol. 38, no. 2, pp. 101–113, 2006.

[33] L. Anselma, P. Terenziani, S. Montani, and A. Bottrighi, "Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines," *Artificial Intelligence in Medicine*, vol. 38, no. 2, pp. 171–195, 2006.

[34] "HL7 Standards - Master Grid," 2015.

[35] M. Samwald, K. Fehre, J. De Bruin, and K.-P. Adlassnig, "The arden syntax standard for clinical decision support: Experiences and directions," *Journal of biomedical informatics*, vol. 45, no. 4, pp. 711–718, 2012.

[36] G. Hripcsak, P. D. Clayton, T. A. Pryor, P. Haug, O. Wigertz, and J. Van der Lei, "The arden syntax for medical logic modules.," in *Proceedings. Symposium on Computer Applications in Medical Care*, pp. 200–204, American Medical Informatics Association, 1990.

[37] R. A. Greenes, A. Boxwala, W. N. Sloan, L. Ohno-Machado, and S. Deibel, "A framework and tools for authoring, editing, documenting, sharing, searching, navigating, and executing computer-based clinical guidelines.," in *Proceedings of the AMIA Symposium*, p. 261, American Medical Informatics Association, 1999.

[38] V. L. Patel, V. G. Allen, J. F. Arocha, and E. H. Shortliffe, "Representing clinical guidelines in glif: individual and collaborative expertise," *Journal of the American Medical Informatics Association*, vol. 5, no. 5, pp. 467–483, 1998.

[39] M. Peleg, A. A. Boxwala, O. Ogunyemi, Q. Zeng, S. Tu, R. Lacson, E. Bernstam, N. Ash, P. Mork, L. Ohno-Machado, *et al.*, "Glif3: the evolution of a guideline representation format.," in *Proceedings of the AMIA Symposium*, p. 645, American Medical Informatics Association, 2000.

[40] M. Peleg, A. A. Boxwala, E. Bernstam, S. Tu, R. A. Greenes, and E. H. Shortliffe, "Sharable representation of clinical guidelines in glif: relationship to the arden syntax," *Journal of biomedical informatics*, vol. 34, no. 3, pp. 170–181, 2001.

[41] Y. Shahar, S. Miksch, and P. Johnson, "The asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines," *Artificial intelligence in medicine*, vol. 14, no. 1-2, pp. 29–51, 1998.

[42] D. Riaño, *Knowledge management for health care procedures*. Springer, 2009.

[43] O. Young, Y. Shahar, Y. Liel, E. Lunenfeld, G. Bar, E. Shalom, S. B. Martins, L. T. Vaszar, T. Marom, and M. K. Goldstein, "Runtime application of hybrid-asbru clinical guidelines," *Journal of biomedical informatics*, vol. 40, no. 5, pp. 507–526, 2007.

[44] S. Miksch, Y. Shahar, and P. Johnson, "Asbru: a task-specific, intention-based, and time-oriented language for representing skeletal plans," in *Proceedings of the 7th Workshop on Knowledge Engineering: Methods & Languages (KEML-97)*, pp. 9–19, Milton Keynes, UK, The Open University, Milton Keynes, UK, 1997.

[45] G. Kong, D.-L. Xu, and J.-B. Yang, "Clinical decision support systems: a review on knowledge representation and inference under uncertainties," *International Journal of Computational Intelligence Systems*, vol. 1, no. 2, pp. 159–167, 2008.

[46] A. Ten Teije, S. Miksch, and P. Lucas, *Computer-based medical guidelines and protocols: a primer and current trends*, vol. 139. Ios Press, 2008.

[47] A. Vollebregt, A. ten Teije, F. van Harmelen, J. van der Lei, and M. Mosseveld, "A study of proforma, a development methodology for clinical procedures," *Artificial Intelligence in Medicine*, vol. 17, no. 2, pp. 195–221, 1999.

[48] J. Fox, N. Johns, and A. Rahmanzadeh, "Disseminating medical knowledge: the proforma approach," *Artificial intelligence in medicine*, vol. 14, no. 1-2, pp. 157–182, 1998.

[49] M. A. Musen, S. W. Tu, A. K. Das, and Y. Shahar, "Eon: a component-based approach to automation of protocol-directed therapy," *Journal of the American Medical Informatics Association*, vol. 3, no. 6, pp. 367–388, 1996.

[50] S. W. Tu, M. A. Musen, *et al.*, "Modeling data and knowledge in the eon guideline architecture," in *Medinfo*, pp. 280–284, 2001.

[51] P. De Clercq, K. Kaiser, and A. Hasman, "Computer-interpretable guideline formalisms," *Studies in health technology and informatics*, vol. 139, p. 22, 2008.

[52] G. Leonardi, A. Bottrighi, G. Galliani, P. Terenziani, A. Messina, and F. Della Corte, "Exceptions handling within glare clinical guideline framework," in *AMIA Annual Symposium Proceedings*, vol. 2012, p. 512, American Medical Informatics Association, 2012.

[53] P. Terenziani, S. Montani, A. Bottrighi, M. Torchio, G. Molino, and G. Correndo, "The glare approach to clinical guidelines: main features," *Studies in health technology and informatics*, pp. 162–166, 2004.

[54] S. R. Abidi, "A conceptual framework for ontology based automating and merging of clinical pathways of comorbidities," in *Workshop on Knowledge Management for Health Care Procedures*, pp. 55–66, Springer, 2008.

[55] D. Riano and W. Ortega, "Computer technologies to integrate medical treatments to manage multimorbidity," *Journal of biomedical informatics*, vol. 75, pp. 1–13, 2017.

[56] B. Jafarpour, *Ontology Merging Using Semantically-Defined Merge Criteria and OWL Reasoning Services: Towards Execution-Time Merging of Multiple Clinical Workflows to Handle Comorbidities.* PhD thesis, Dalhousie University, 2014.

[57] B. Jafarpour and S. S. R. Abidi, "Merging disease-specific clinical guidelines to handle comorbidities in a clinical decision support setting," in *Conference on Artificial Intelligence in Medicine in Europe*, pp. 28–32, Springer, 2013.

[58] S. Wilk, M. Michalowski, W. Michalowski, M. M. Hing, and K. Farion, "Reconciling pairs of concurrently used clinical practice guidelines using constraint logic programming," in *AMIA Annual Symposium Proceedings*, vol. 2011, p. 944, American Medical Informatics Association, 2011.

[59] S. Wilk, W. Michalowski, M. Michalowski, K. Farion, M. M. Hing, and S. Mohapatra, "Mitigation of adverse interactions in pairs of clinical practice guidelines using constraint logic programming," *Journal of biomedical informatics*, vol. 46, no. 2, pp. 341–353, 2013.

[60] V. Zamborlini, M. Da Silveira, C. Pruski, A. ten Teije, E. Geleijn, M. van der Leeden, M. Stuiver, and F. van Harmelen, "Analyzing interactions on combining multiple clinical guidelines," *Artificial intelligence in medicine*, vol. 81, pp. 78–93, 2017.

[61] V. Zamborlini, M. Da Silveira, C. Pruski, A. ten Teije, and F. van Harmelen, "Analyzing recommendations interactions in clinical guidelines," in *Conference on Artificial Intelligence in Medicine in Europe*, pp. 317–326, Springer, 2015.

[62] G. A. Mendoza and H. Martins, "Multi-criteria decision analysis in natural resource management: a critical review of methods and new modelling paradigms," *Forest ecology and management*, vol. 230, no. 1-3, pp. 1–22, 2006.

[63] V. Belton and T. J. Stewart, "Dea and mcda: Competing or complementary approaches?," in *Advances in decision analysis*, pp. 87–104, Springer, 1999.

[64] K. Marsh, M. Goetghebeur, P. Thokala, and R. Baltussen, *Multi-Criteria Decision Analysis to Support Healthcare Decisions*. Springer, 2017.

[65] P. Thokala, N. Devlin, K. Marsh, R. Baltussen, M. Boysen, Z. Kalo, T. Longrenn, F. Mussen, S. Peacock, J. Watkins, *et al.*, "Multiple criteria decision analysis for health care decision making—an introduction: report 1 of the ispor mcda emerging good practices task force," *Value in health*, vol. 19, no. 1, pp. 1–13, 2016.

[66] Z. Philips, L. Ginnelly, M. Sculpher, K. Claxton, S. Golder, R. Riemsma, N. Woolacott, and J. Glanville, "Review of guidelines for good practice in decision-analytic modelling in health technology assessment," 2004.

[67] G. Elwyn, D. Frosch, R. Thomson, N. Joseph-Williams, A. Lloyd, P. Kinnersley, E. Cording, D. Tomson, C. Dodd, S. Rollnick, *et al.*, "Shared decision making: a model for clinical practice," *Journal of general internal medicine*, vol. 27, no. 10, pp. 1361–1367, 2012.

[68] M. Perleth, B. Gibis, and B. Göhlen, "A short history of health technology assessment in germany," *International Journal of Technology Assessment in Health Care*, vol. 25, no. S1, pp. 112–119, 2009.

[69] T. L. Saaty, "Analytic hierarchy process," in *Encyclopedia of operations research and management science*, pp. 52–64, Springer, 2013.

[70] M. Ryan, K. Gerard, and M. Amaya-Amaya, *Using discrete choice experiments to value health and health care*, vol. 11. Springer Science & Business Media, 2007.

[71] A. Benson, T. Bekaii-Saab, E. Chan, Y.-J. Chen, M. Choti, H. Cooper, and P. Engstrom, "NCCN Clinical Practice Guideline in Oncology Colon Cancer," tech. rep., National Comprehensive Cancer Network, 2013.

[72] G. E. Krasner, S. T. Pope, *et al.*, "A description of the model-view-controller user interface paradigm in the smalltalk-80 system," *Journal of object oriented programming*, vol. 1, no. 3, pp. 26–49, 1988.

[73] F. Gonçalves, T. Oliveira, J. Neves, and P. Novais, "Compguide: Acquisition and editing of computer-interpretable guidelines," in *World Conference on Information Systems and Technologies*, pp. 257–266, Springer, 2017.

[74] M. Peleg, "Computer-interpretable clinical guidelines: a methodological review," *Journal of biomedical informatics*, vol. 46, no. 4, pp. 744–763, 2013.

# A

## USE CASE DESCRIPTION

This appendix provides a description of the different use cases of the drug dispute resolution system in tabular format. The first use case presented concerns the choice of the range of each criterion defined by the healthcare professional. Table 18 presents this use case.

Table 18.: Use Case Choose Range of Criteria

| | |
|---|---|
| Super Use Case | |
| Brief Description | Use case that describes the choice of range for each criterion by the healthcare professional medicamentoso |
| Preconditions | Existence of Criteria |
| Post-conditions | Range of defined criteria |
| Flow of Events | |

| | Actor Input | System Response |
|---|---|---|
| 1 | Select the range of each defined criterion | |
| 2 | | Displays the range for Criterion A |
| 3 | | Displays the range for Criterion B |
| 4 | | Displays the range for Criterion C |
| 5 | Choose the range for Criterion A | |
| 6 | Choose the range for Criterion B | |
| 7 | Choose the range for Criterion C | |
| 8 | | Save Criterion A range |
| 9 | | Save Criterion B range |
| 10 | | Save Criterion C range |

Table 19 demonstrates the textual description of the use case Choose the importance of criteria.

Finally, represented in Table 20 , represents the choice of the importance of each alternative in the determined criteria, in case the use case Choose Values for alternatives.

Table 19.: Use Case Choose Importance of Criteria

<table>
<tr><td>Super Use Case</td><td></td></tr>
<tr><td>Brief Description</td><td>Use Use Case that describes choosing the importance of the criteria to calculate the weight of each.</td></tr>
<tr><td>Preconditions</td><td>Existence of Criteria</td></tr>
<tr><td>Post-conditions</td><td>Definition of criteria weights made</td></tr>
<tr><td>Flow of Events</td><td>
<table>
<tr><td></td><td>Actor Input</td><td>System Response</td></tr>
<tr><td>1</td><td>Select the importance of the criteria</td><td></td></tr>
<tr><td>2</td><td></td><td>Displays the criteria</td></tr>
<tr><td>3</td><td>Choose the importance for criterion A</td><td></td></tr>
<tr><td>4</td><td></td><td>Validate and save set value</td></tr>
<tr><td>5</td><td>Choose the importance for criterion B</td><td></td></tr>
<tr><td>6</td><td></td><td>Validate and save set value</td></tr>
<tr><td>7</td><td>Choose the importance for criterion C</td><td></td></tr>
<tr><td>8</td><td></td><td>Validate and save set value</td></tr>
</table>
</td></tr>
<tr><td>Exception1 [value outside defined range] (step 4, 6, 8)</td><td>
<table>
<tr><td></td><td>Actor Input</td><td>System Response</td></tr>
<tr><td>1</td><td></td><td>Value chosen out of defined importance range</td></tr>
</table>
</td></tr>
</table>

Table 20.: Use Case Choose Values for Alternatives

<table>
<tr><td>Super Use Case</td><td></td></tr>
<tr><td>Brief Description</td><td>Use Case that describes choosing the importance of alternatives for each criterion.</td></tr>
<tr><td>Preconditions</td><td>Existence of alternatives and criteria</td></tr>
<tr><td>Post-conditions</td><td>Definition of the scores of the importance of alternatives</td></tr>
<tr><td>Flow of Events</td><td>
<table>
<tr><td></td><td>Actor Input</td><td>System Response</td></tr>
<tr><td>1</td><td>Select the importance of alternatives in each criterion</td><td></td></tr>
<tr><td>2</td><td></td><td>Shows the criteria and alternatives</td></tr>
<tr><td>3</td><td>Choose the importance of alternative 1 in criterion A</td><td></td></tr>
<tr><td>4</td><td></td><td>Validate and save set value</td></tr>
<tr><td>5</td><td>Choose the importance of alternative 2 in criterion A</td><td></td></tr>
<tr><td>6</td><td></td><td>Validate and save set value</td></tr>
<tr><td>7</td><td>Choose the importance of alternative 3 in criterion A</td><td></td></tr>
<tr><td>8</td><td></td><td>Validate and save set value</td></tr>
<tr><td>9</td><td>Choose the importance of alternative 1 in criterion B</td><td></td></tr>
<tr><td>10</td><td></td><td>Validate and save set value</td></tr>
<tr><td>11</td><td>Choose the importance of alternative 2 in criterion B</td><td></td></tr>
<tr><td>12</td><td></td><td>Validate and save set value</td></tr>
<tr><td>13</td><td>Choose the importance of alternative 3 in criterion B</td><td></td></tr>
<tr><td>14</td><td></td><td>Validate and save set value</td></tr>
<tr><td>15</td><td>Choose the importance of alternative 1 in criterion C</td><td></td></tr>
<tr><td>16</td><td></td><td>Validate and save set value</td></tr>
<tr><td>17</td><td>Choose the importance of alternative 2 in criterion C</td><td></td></tr>
<tr><td>18</td><td></td><td>Validate and save set value</td></tr>
<tr><td>19</td><td>Choose the importance of alternative 3 in criterion C</td><td></td></tr>
<tr><td>20</td><td></td><td>Validate and save set value</td></tr>
</table>
</td></tr>
<tr><td>Exception1 [value out of range] (steps 4-20 in even numbers</td><td>
<table>
<tr><td></td><td>Actor Input</td><td>System Response</td></tr>
<tr><td>1</td><td></td><td>Value chosen outside Criteria range</td></tr>
</table>
</td></tr>
</table>

# B

## UI MOCKUPS

This appendix demonstrates the mockups developed. Figure 39 corresponds to the view that presents the alternatives in the MCDA model.

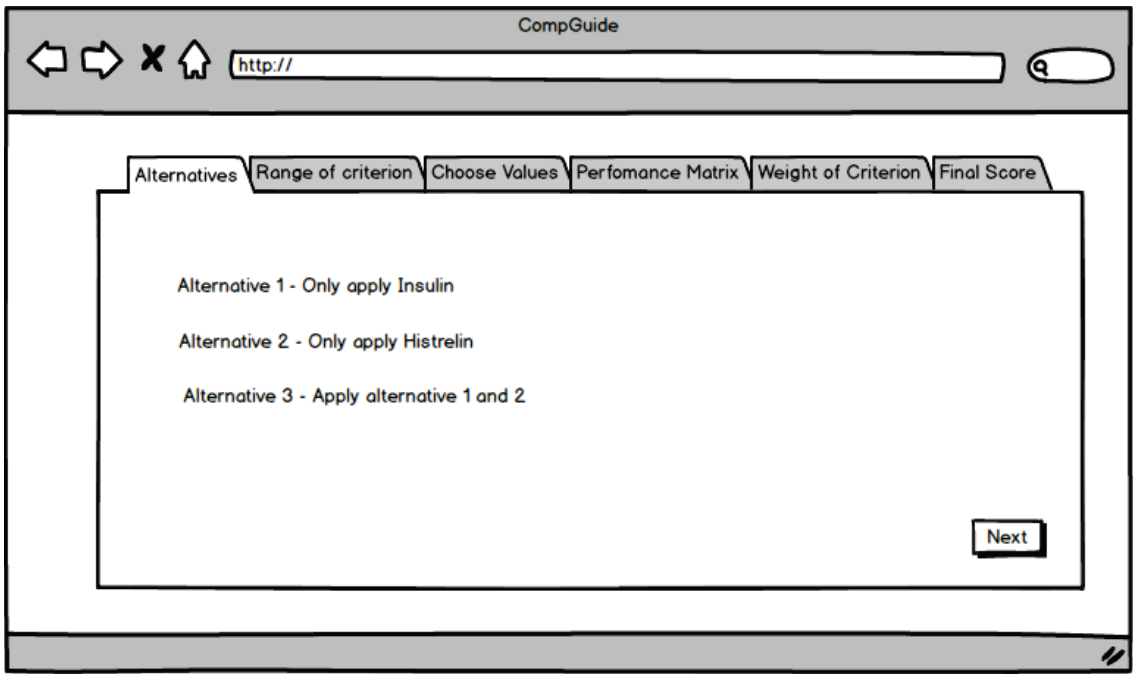



Figure 39.: Mockup with Alternatives Preview

Figure 40 shows the view that will be responsible for choosing the importance of each alternative in a given criterion.

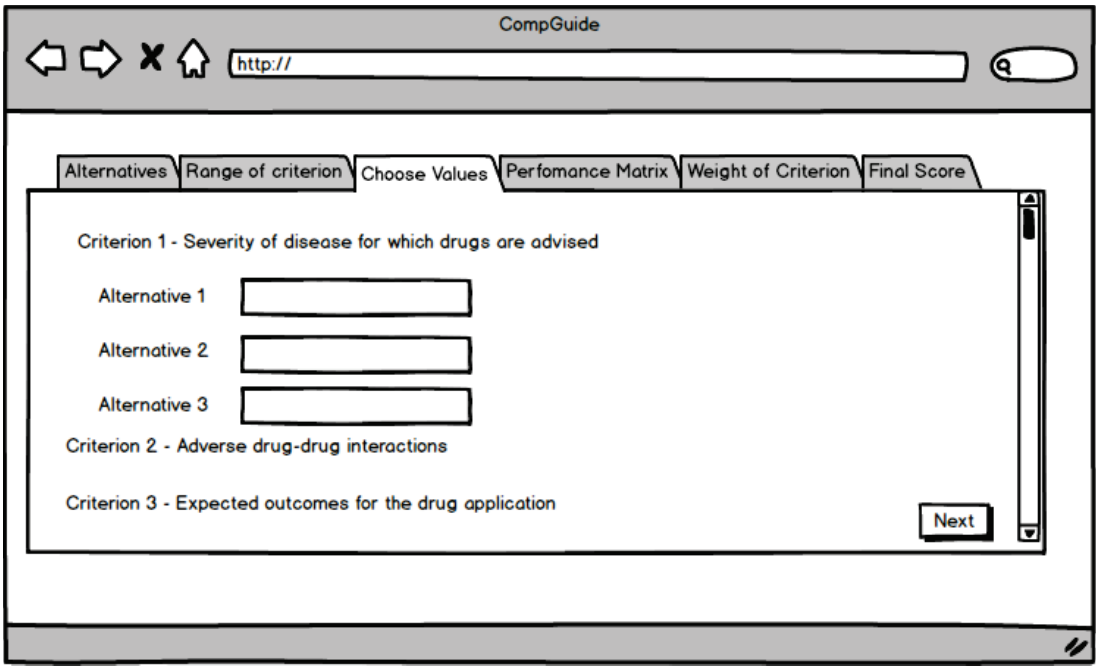



Figure 40.: Mockup of choice of alternative importance in each criterion

Finally, Figure 41 represents the view responsible for the performance matrix presentation.

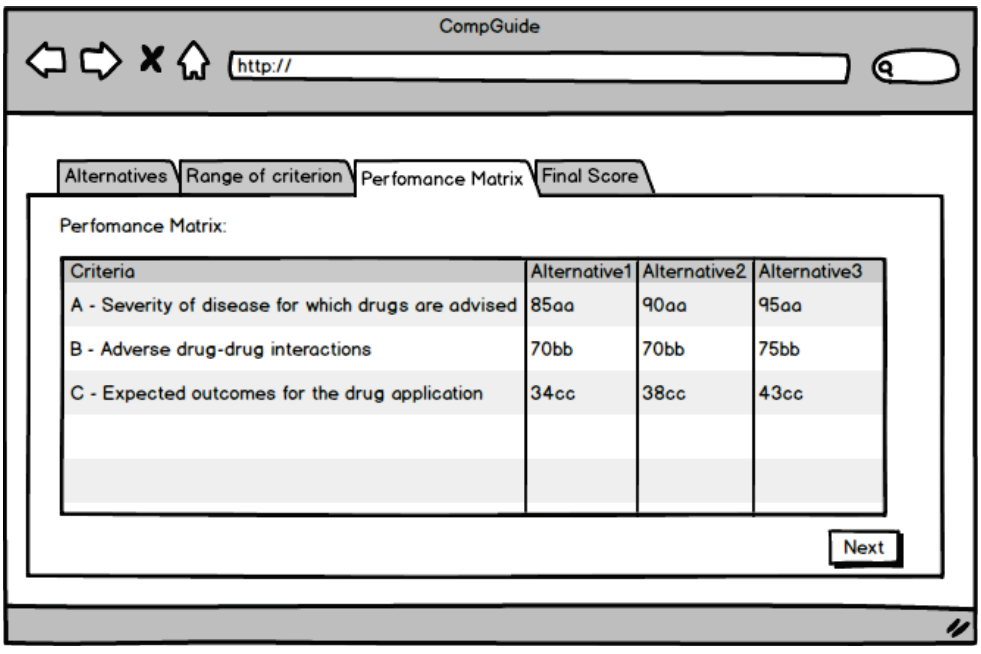



Figure 41.: Performance matrix Mockup

# C

## SEQUENCE DIAGRAM

This appendix deals with the application sequence diagrams. Figure 42 represents the sequence diagram responsible for obtaining interactions using the RxNorm Interaction API.

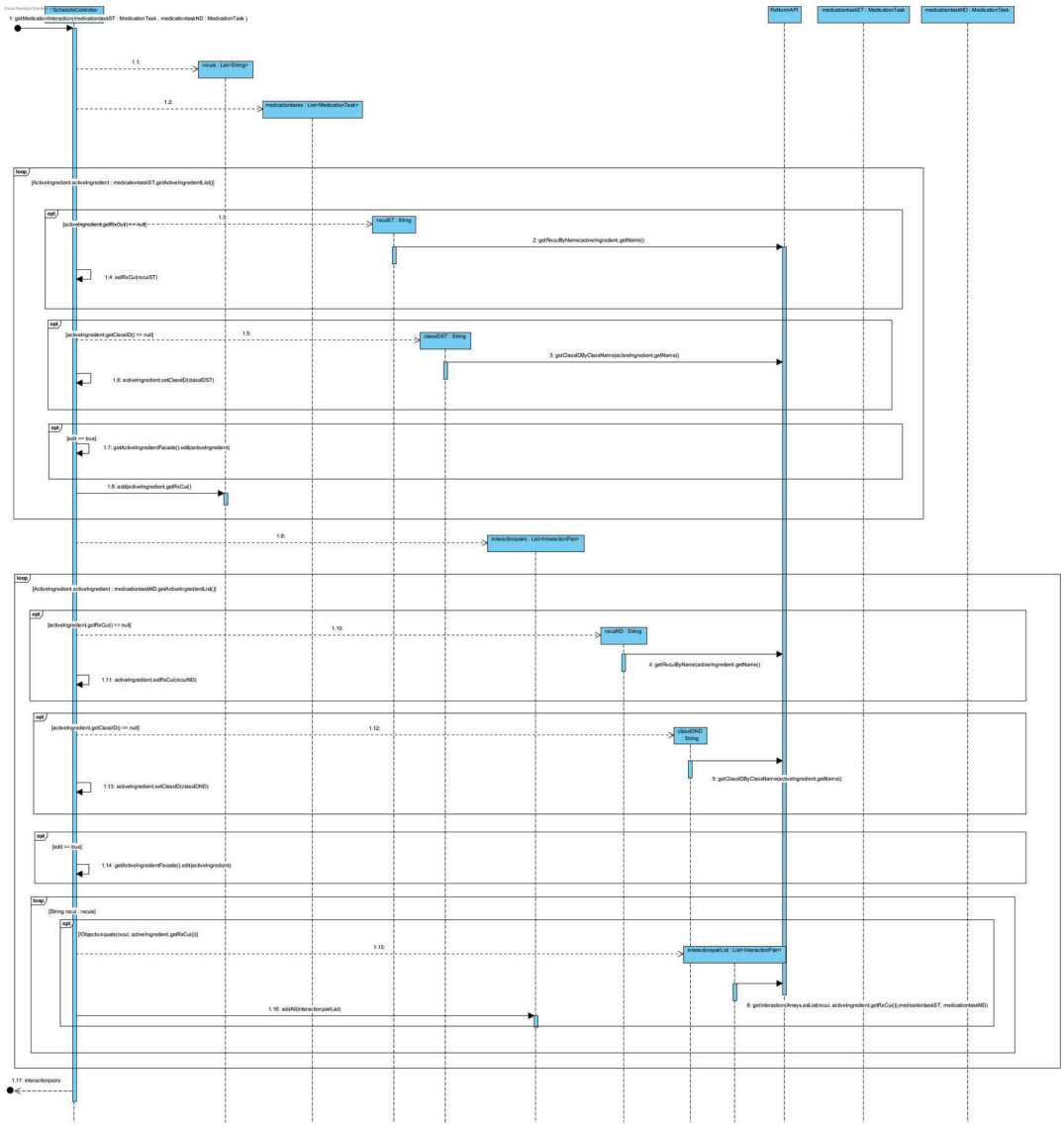


Figure 42.: Get Interaction Sequence Diagram

Figure 43 shows the diagram corresponding to obtaining similar medication using the RxNorm Similar API.

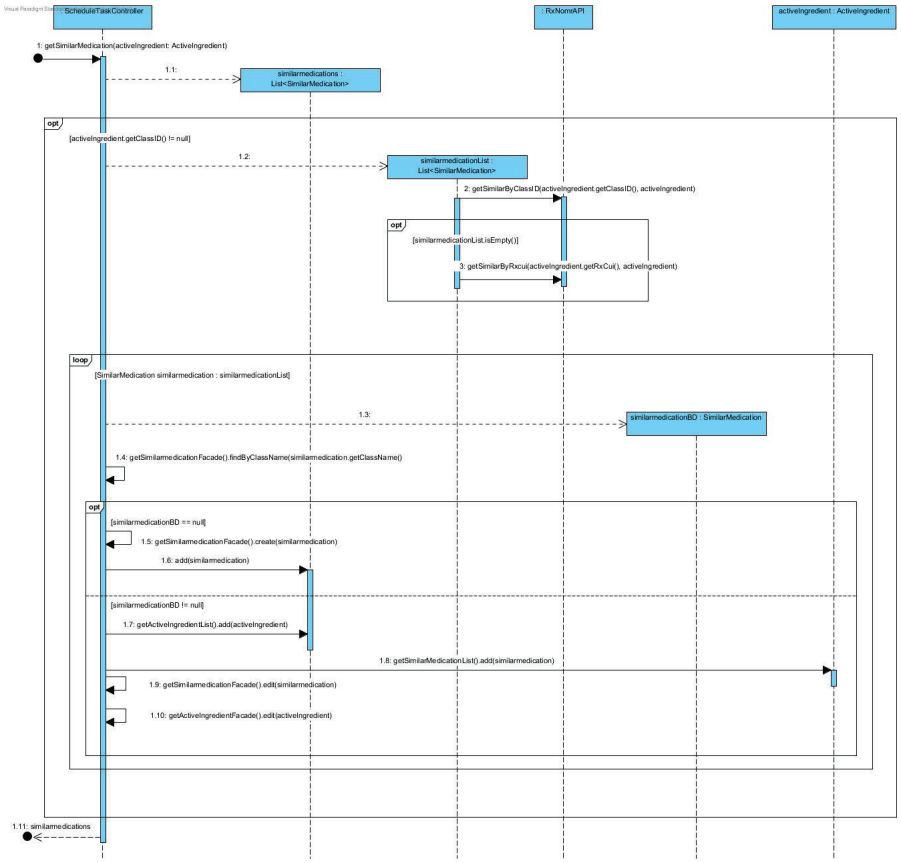



Figure 43.: Get Similar Medication Sequence Diagram

# D

## WEB APPLICATION

This appendix refers to the presentation of the web application, with the main objective of providing more details about the different interfaces available. Figure 44 shows the web page with the details corresponding to task A61.

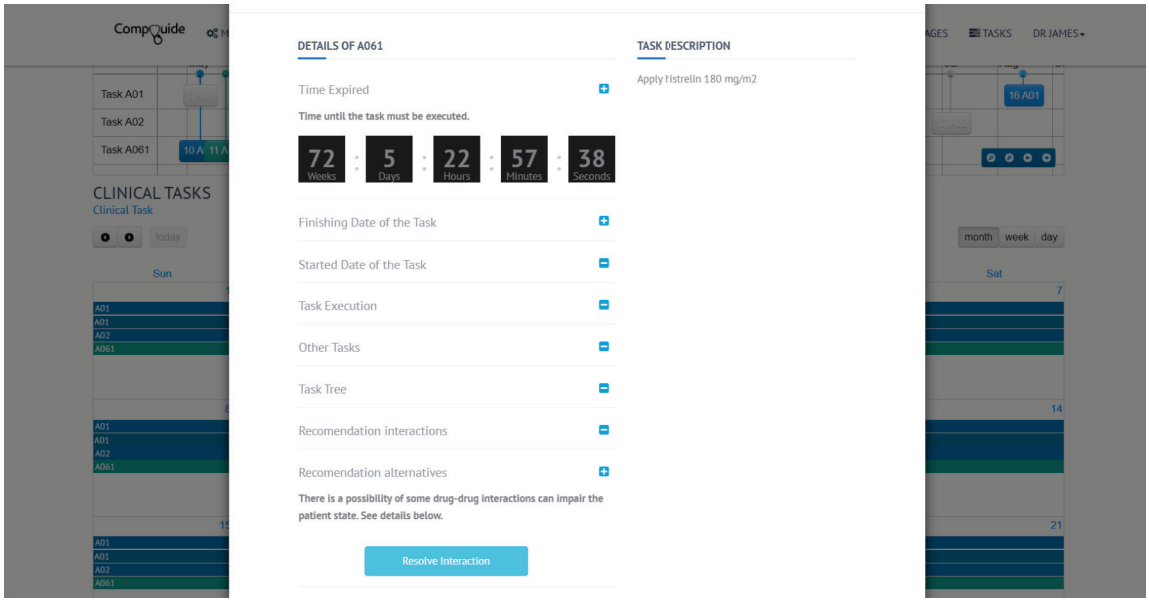



Figure 44.: Task A61 Details

Figure 45 corresponds to the first tab of the MCDA model web page, where you can see the alternatives in the specific case study.

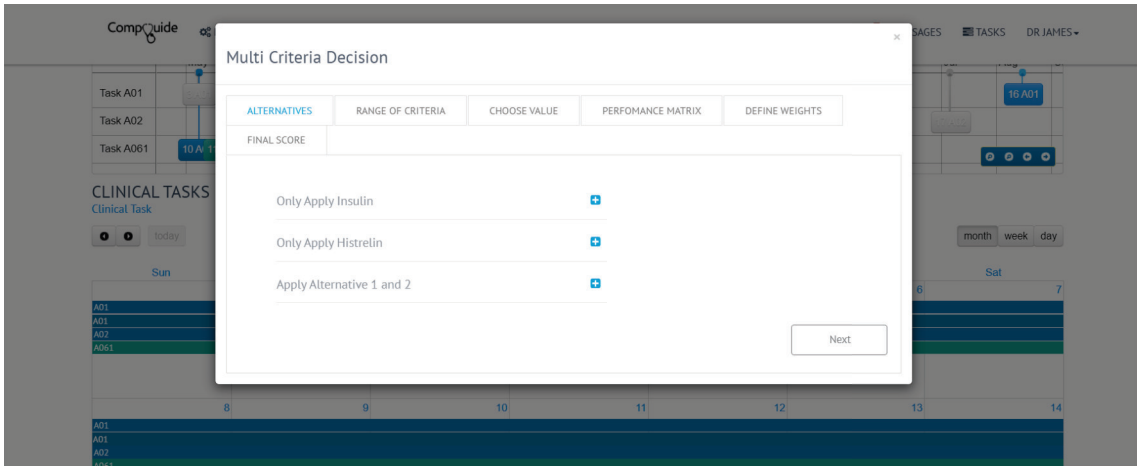



Figure 45.: MCDA model alternatives available for the conflict in question

In the last case, Figure 46 shows the tab where the performance matrix of the chosen case studies is displayed.

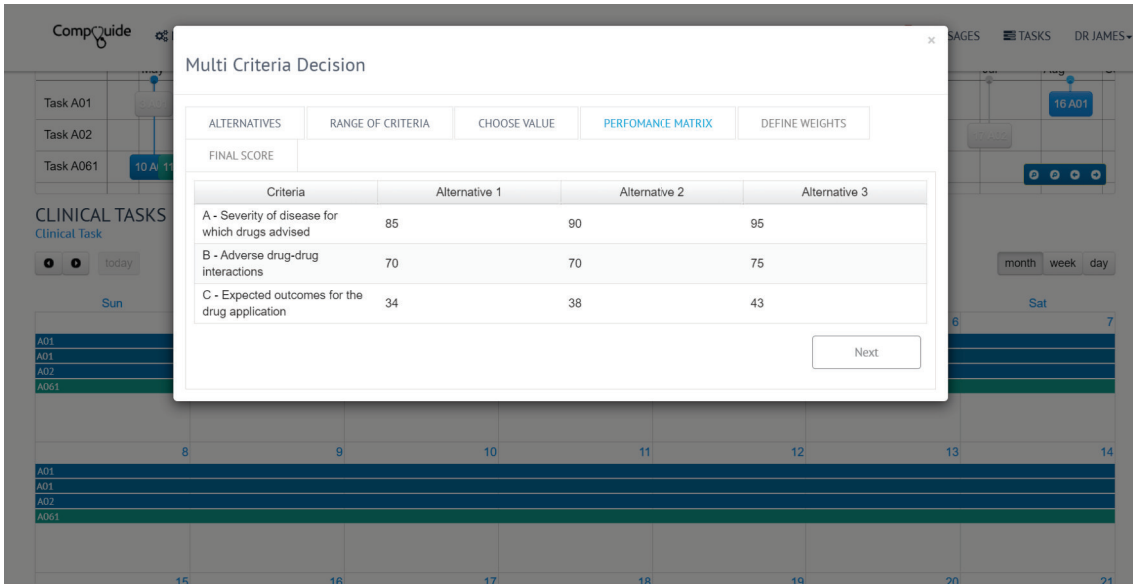



Figure 46.: Performance matrix for conflict between task A02 and A61