Individual Dissertation

COGNITIVE ARGUMENTATION-BASED ASSISTANTS IN GYNECOLOGY

Georgios Procopiou

UNIVERSITY OF CYPRUS

COMPUTER SCIENCE DEPARTMENT

May 2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

COGNITIVE ARGUMENTATION-BASED ASSISTANTS IN GYNECOLOGY

Georgios Procopiou

Supervisor

Professor Antonis Kakas

The individual dissertation submitted for partial fulfillment of the requirements for obtaining Computer Science's Undergraduate degree, Department of Computer Science, University of Cyprus

May 2022

Acknowledgements

I would first like to express my deepest appreciation to my supervisor, Dr. Antonis Kakas, for his inviable contribution and willingness to offer me his guidance whenever I needed it. By trusting me and giving me this extraordinary opportunity to work on this project he helped me to broaden my horizons, enrich my skill quiver and cultivate qualities that I will carry forever.

I would also like to express my gratitude to the medical professional, Dr. Vasilios Tanos, for the precious time and dedication as well as for the insightful comments and suggestions he provided during my research. His immense knowledge and enthusiasm intrigued my interest in the field of Gynecology.

I am also grateful to a master's student, Mr. Ioannis Yiangou, for his valuable advice and practical suggestions as well as for his patience during our daily meetings, which cannot be underestimated.

In addition, I would like to thank my parents and especially my brother, Pantelis Prokopiou, for their unwavering support and belief in me as well as my friends and a fellow student of mine, Sophia Sarri, for their deep understanding through the whole duration of my dissertation and their efforts to keep me motivated to pursue my ambitions.

I would like to extend my sincere thanks to my beloved grandparent, Eleni and Pantelis Panayi, for the countless years of happiness and continual inspiration they gave me with their own wise and unique way.

Finally, I would like to praise and thank God, the Almighty, for the guidance and selfless love but also for the countless blessings, opportunities and knowledge who has graciously given me.

This accomplishment would have not been possible without you. Thank you.

Abstract

In this thesis GAIA, “GynAIcologicA” is going to be presented. GAIA is an Artificial Intelligent Diagnostic System which has the ability to provide support during the clinical diagnostic procedure to the experts of the field with regards to obstetrics and gynecology. This Diagnostic Decision-Support System is based on the Argumentation Framework which is considered to be the ideal platform for developing not only Explainable but also Human-Centered and Law Abiding Cognitive Artificial Intelligent Assistants.

A variety of functionalities were added upon the existing version of the system to enhance its decision-making capabilities and improve its diagnosis in each of the steps that may occur during the examination procedure. Its ability to use knowledge that contradicts any possible conditions or diseases from being possible, also known as Contraindications, and to hide it from the doctor’s radar upon demand as well as its ability to suggest possible contraindications to the user based on the context and the relevant possible disorders are just some forms of the newly implemented capabilities and services offered by the system.

A superabundant and extensive bibliographic review as well as a rich study on the field of medicine with the help of experts in the field were followed to conclude to the basic guidelines and principles that the system must obey as well as to achieve its new goals while maintaining its initial standards and purpose.

Finally, a two-phased thorough testing and evaluation procedure was developed and applied to identify any potential gaps and missing concepts in the knowledge base of the system that are going to be implemented in a future version. The ultimate goal is to create new functionalities to the system which are easy to use by novice as well as senior doctors and can be very handy and useful in real-life scenarios.

TABLE OF CONTENTS

FIGURES

TABLES

Chapter 1

Introduction

1.1 Problem Statement and Motivation

Undoubtedly, we thrive in a digitized world where Artificial Intelligent systems can be used in a wide spectrum of domains and in any kind of applications for humankind to benefit from. This ever-increasing demand for embracing and adopting this technology intrigued our interest into engaging further with it and develop and enhance an Artificial Intelligent Clinical Decision Support System to be applied and used at is maximum potential in the medical field of Gynecology. Despite the fact that all sorts of kinds of fields make use of Artificial Intelligent Systems, in this field such systems are relatively new.

The capabilities of the existing system were illustrated to me and I was fascinated with its ability to provide diagnosis for one hundred and thirty-seven (137) different disorders. However, for GAIA to find real-life applications and enhance the quality of its decision-making capabilities it was a necessity to add on the existing system the feature of contraindications. This feature will allow the system to exclude disorders considered to be possible if enough information is gathered with regards to each disease. If contraindications are used correctly and in a wise manner the potentials and capabilities of the system will be unraveled and without any doubt will bring a handful of advantages for gynecologists to benefit from.

Since healthcare services can have a profound impact on human's lives, our main goal aims to carry away some of the burden that gynecologists have during their diagnostic procedure. Although this system can ease the experts of the field during a routine checkup, our approach and rational does not aim to replace the doctor with its intelligent assistant but to embrace the users by maximizing their capabilities and enriching their decisions.

1.2 Thesis Overview

The structure of this thesis is organized in seven (7) distinct chapters each of which is considered important in its own unique way to build the fundamentals that are going to be discussed in the later chapters. Some chapters introduce necessary background information while others give detailed description of the procedures and solutions considering the aforementioned problem.

Chapter 2 states the necessary theoretical information with regards to the field of Computer Science and more specifically the branch related with Artificial Intelligence such as general concepts, principles and in some cases concerns and guidelines.

Chapter 3 presents the background theory regarding the field of Medicine and to be more precise and in-depth explanation of the rationale behind the creation and implementation of Clinical Decision Support Systems is illustrated. Additionally, in this chapter there are references to the field of Gynecology and Obstetrics as well.

Chapter 4 presents both the existing theoretical model as well as the theoretical model expansion related with the thesis problem, including the way in which information are collected and used in the "diagnostic" procedure and the modelling of "contraindications" within the context of Argumentation with high-level descriptions and explanations.

Chapter 5 combines a variety of technologies and methods that were used to implement the theoretical expansion model on top of the existing model and create an easy-to-use working computer application.

Chapter 6 illustrates the two-way approach that was used and going to be used in future for evaluating the system accompanied with their results as well as the methods that are going to be used to enhance and enrich its knowledge base.

Finally, in Chapter 7 the conclusions of the thesis are summarized along with potential limitations that can be used in future work to enable GAIA to improve and find applications in the real world and to become a well-known tool with wide range of use cases in the medicine industry.

Chapter 2

Background Theory in Artificial Intelligence

2.1 Artificial Intelligence

Artificial Intelligence; (AI), is a broad term which is used to describe any machine or computer system, that has the ability to emulate human-like intelligence, behavior and reasoning to enhance its decision-making and problem-solving capabilities [17]. An Artificial Intelligent System scans the environment, receives stimuli from its sensors and forms a generalized perception about it. Based on its rational it will then quickly respond to the problem, while enriching its knowledge base from its new experiences, as a way of adaptation to the always-changing environment [9].

The idea of such machines, which can think on their own, dates back to ancient Greece. According to Greek mythology, Hephaestus the god of metallurgy and fire created Talos a huge robot made from bronze and Pandora a robot replica of a woman [28]. In the mid-20th century, Alan Turing, a British mathematician and logician, in his paper “Computing Machinery and Intelligence” offered to answer the question; “can machines think?”. As a result, the well-known Turing Test was introduced to determine if human intelligence can be reproduced from computer systems [17], [8]. Two years later, the first implementation of an Artificial Intelligent System was launched by Cristopher Strachey, director of the Programming Research Group at the University of Oxford. The British Computer Scientist created a program that was capable of completing a game of checkers at a reasonable pace, which was one of its kind [7].

Artificial Intelligent Systems have come a long way since their early introduction in 1951. Behind the rapid growth of Artificial Intelligent Systems in the last decade, are laying two major reasons. Starting with the enormous quantities of available data thanks to the Internet, as well as the great advancement in computing power, and the recent addition of hardware designed specifically for Artificial Intelligent models, led to groundbreaking discoveries in the field [20].

Nowadays, Artificial Intelligent Systems have a variety of applications in a wide range of fields. Such technologies can benefit both the scientific and industrial worlds in the process of making crucial decisions that tend to be time consuming for humans. Some of the most popular examples of these systems are Siri and Alexa, as well as Cortana and Google Assistant which use sound, image, and speech recognition to help individual users with their everyday tasks. It is worth mentioning that Tesla's autopilot which uses Artificial Intelligence, combined with the car's sensors, is responsible for self-driving and steering the vehicle, and to predict and prevent crashes and fatal accidents [23]. With the technological advancements and Artificial Intelligence backing our existence the future seems bright and as Nick Bostrom claims, "Machine intelligence is the last invention that humanity will ever need to make", but we still have a long way to go [2].

2.2 Human-Centered, Ethical and Law Driven Artificial Intelligence

Dreams and nightmares of Artificial Intelligence are illustrated throughout a variation of books and movies along with technology's mesmerizing advancements and the terrifying possibilities of out-of-control intelligent robots. Machines eventually rule and us, humans, can do nothing about it. However, there is the possibility for an alternative future, a future which is packed with intelligent devices that have the potential of amplifying human abilities [27]. This rather compelling prospect which promises this appealing future, is called Human-Centered Artificial Intelligence (HCAI) also known as human-in-the-loop (HILT) and has a single goal, to achieve what neither a human nor a machine can attain on their own [3].

The HCAI framework can bridge the gap between ethics and practice and exalt humans rather than replace them, leading to a safer and more reliable future. The human-in-the-loop approach, supports and respects human values, raises appreciation for the humankind and serves only the human needs. A human-centered future will eliminate any possibility of humans been overthrown by intelligent systems, humans' privacy and security threats will be neutralized and finally, unemployment due to intelligent robots will be prohibited [27].

In summary, the human-centered artificial intelligence framework defines the standards and builds the fundamentals of intelligent and moral-driven machines. As of the laws, they can be hardwired into the "brains" of the artificial intelligent systems which will be programmed to obey and follow them.

In accordance with the aforementioned, one can think, that humans are faced with moral dilemmas almost daily, but these intelligent machines how are they supposed to act when they find their selves in such situations? What will happen if their choices are either to "be ethical while breaking the law" or "be unethical but abiding the law"?

"The Footbridge Dilemma" can be considered as an example which was proposed by philosophers to understand in depth the problem. In this scenario, the intelligent robot is standing on a footbridge above the tram tracks. In the distance a trolley can be observed rushing its way towards a group of unsuspecting people. On the bridge, next to the robot there is a man standing and the robot is confident that the size of the men can stop the tram, if pushed over. The dilemma proposes: "Shall the intelligent system push the man off the bridge on to the railway, sacrificing him to stop the tram and therefore save the masses?". With other words, if the intelligent robot decides to act based on its morals and save as many people as possible, it is going to break the laws as an intentional act of murder by pushing the man off the bridge but again, this action promotes an even greater good. On the other hand, the robot can decide to stay passive by simply just to not pushing the man and in this case a group of people is going to die instead [13].

This, and much more dilemmas where proposed and discussed in order to understand humans' behavior and their rational based on their morals. It was immediately apparent that from those studies, peoples' answers were not the same and, in the occasions, where they agreed upon the same act, their justifications why, vary [13]. That is exactly why Artificial Intelligent Systems need to explain and justify their rational, and the reasons behind their actions.

In conclusion, it is important for researchers and developers to find a middle ground where ethics, morals and laws are implemented and to ensure that the system can make compromises between those so its acts will bring more benefits than destructions.

2.3 Explainable Artificial Intelligence

Technological advancements made during the last decades led to the invention of revolutionary artificial intelligent systems that posed new challenges to the humankind to solve. "Black box" is the most common term that is used to describe this kind of systems and machines because their process, which they use to calculate a result, is nearly impossible to be interpreted and comprehended by humans. Even the developers who created the algorithm cannot understand or explain what exactly is happening inside them nor how it reached to a certain outcome [18]. The need for machines with the capability to explain their actions emerges.

Explainable Artificial Intelligence (XAI) can be considered as a set of methods and processes that give the opportunity to humans to take a grasp on how the output and the results are generated based on the rational and the "thinking process" (algorithm) of the intelligent system. Explainability in intelligent machines allow the developers and the daily users to understand which factors had led the system to a specific output because their choices and logic will be justified in natural language text. The generated explanations can also be used in a later stage to calculate the model's accuracy and fairness as well as to find any potential biases that affect the overall performance of the system and its ability to generalize. With that said, is obvious that the key requirement for responsible Artificial Intelligence lies under its ability to explain itself [18].

Several differences can be identified between a traditional intelligent model (AI) and a model with the ability to explain its choices (XAI). Firstly, and more importantly, Explainable Artificial Intelligent models can provide reasonable explanations on "why" and "what" influenced its decision-making process, as opposed with the normal Artificial Intelligence, which suffers from absence of an explanation. Moving on, explainability mitigates legal and security compliance, essential qualities that lack from the traditional models [18]. Although explainable machines can be a powerful tool for the users and businesses, they often come with a cost on complexity and somewhat on time performance [14].

With interpretability marching slowly but steadily towards artificial intelligence, a handful of benefits can be achieved. Starting with the most obvious one, which is the ability to explain their decisions which is essential to guarantee that a system can perform its tasks as expected. In addition, the system's transparency and traceability can increase with the use of explainability, building trust between humans and machines [18].

Nowadays, Explainable Artificial Intelligent systems have a variety of applications and use cases in numerous domains. To begin with, in the Financial Service sector these systems can be used to accelerate the resolution of complaints thus improving the customer's experience. Criminal Justice sector also benefits from their usage. They are mainly used for accelerating DNA analysis results as well as for optimizing the processes for risk assessment. Finally, and most importantly, explainable artificial intelligence systems also contribute to the Healthcare sector. They are used to speedup, optimize, and enhance the medical diagnosis of a patient while improving the transparency of the patient's care [18].

2.4 Argumentation

“Argumentation” can be considered as an alternative form of programming, and it is an extension which is built on top of the Logic Programming Framework. It can be related with common sense programming and its major components, the arguments, can produce the results using reasoning. Semantically, “Argumentation”, is related as one of the most important aspects of human’s intelligence. It is inextricably linked with the ability of people to create and build ideas, to understand complicate statements and to perform reasoning to express their opinions and thoughts [11]. Therefore, a system that has the ability to argument its way and conclude to a decision, is like a human with God-tier capabilities.

Let us consider a set of arguments represented by a set of symbols A_U and a relation R_U on A_U x A_U. This pair (A_U, R_U) is called universe and allows us to illustrate and represent a set of possible arguments along with their interactions. In other worlds, a knowledge base is the set of all arguments and attacks, which can be derived from the formulae of the base [15].

Now, imagine the argumentation framework as a graph; G<A, R> and consider the following abstract example of such graphs:

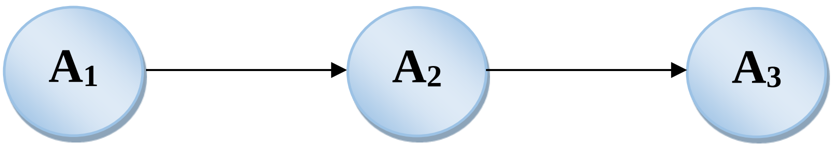

Figure 2.1. Abstract Argumentation Graph

From this example one can observe that, the first argument, attacks the second argument and the second argument attacks the third. With other words “A1 attacks A2” and “A2 attacks A3”.

Moving on with a more complicated example regarding a trial with the set of arguments A_U = [a_1, a_2, a_3, a_4, a_5] and the following relation R_U.

- a1: X is not guilty of premeditated murder of Y
- a2: X is guilty of Y's premeditated murder
- a3: X has an alibi, during the murder X was at work with colleagues
- a4: X is the boss of the business; the alibi induces suspicions about the testimony
- a5: X was so deeply in love with Y and asked for a marriage 5 times, each time the proposal was turned out
- a6: X loved Y; common sense suggests that a person who loves someone does not mean harm to his/her loved one

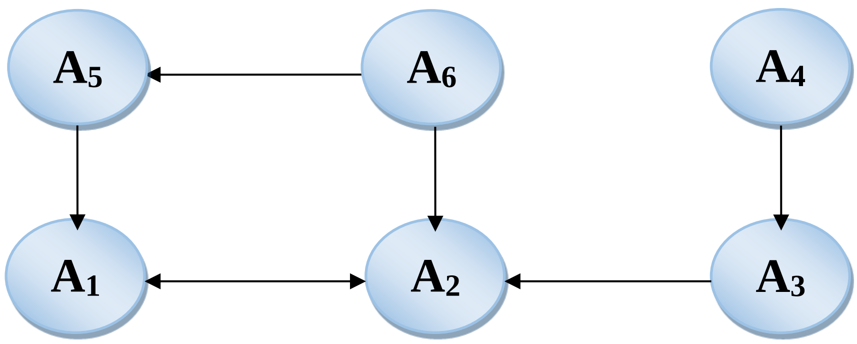

Figure 2.2. Trial Graph Representation using Argumentation

The graph derives directly from the knowledge base (A_U, R_U) and can differ slightly from the definition by the fact that relationships and arguments are built accordingly [15]. Within the concept of "attacks", the arguments can now form groups which support the same position and together can strengthen an argument and defend it against the opposing ones.

All in all, an argumentation graph G on (A_U, R_U) can be considered as a pair (A, R) where:

- $A \subseteq A_U$ a finite set of vertices called "arguments"
- $R \subseteq R_U \cap (A \times A)$ a set of edges, called "attacks"

Several prominent domains use computational models of argumentation for the development of real-life applications. Starting with the most obvious, the law department exploit these models during trials because legal reasoning is argumentative to a large extend. Applications of argumentation models can also be found in debates because of the need of decision support as well as in e-democracy where individuals can critique and debate about government's decisions. Last but not least, the health care domain exploits the use of argumentation models because it allows for important conflicts to be highlighted and analyzed as well as for patient's diagnosis due to the complexity and heterogeneity of the health-care information [19].

2.5 Admissibility and Explainability in Argumentation

Under the context and through the process of arguing a distinct advantage emerges which is the ability to generate explanations using some proponent and opponent debate game. However, this framework does not only provide the ability to create concrete justifications but also to identify an "acceptable only" subset of arguments throughout a set of arguments [16].

To introduce the notion of explainability and admissibility within the context of the argumentation framework let me start by explaining a very simple principle: "For every argument, A, which is admissible, an explanation of why it is accepted should be available based with A's attack relations with other arguments" [21].

Imagine that we have a set of n arguments, A, and a subset of A, which is an admissible set of arguments, A_{adm} with $[a_1, a_m] \in A_{adm}$ where $m \leq n$. This admissible only subset of arguments can exist if and only if, the following criteria are meet [6]:

1. All arguments in A_{adm} do not oppose a threat to other arguments of A_{adm} and do not have any attack relationships between them. With other worlds A_{adm} is a "conflict-free" set of arguments [6].

2. An argument a_x is admissible with respect to a set of arguments A_{adm} if for all arguments the following holds: If a is attacked by a_y, then there is an argument a_z in A_{adm} such as a_y is attacked by a_z thus a_y is a defeater [6].

Now, let us consider the following scenario, a conversation between a group of friends that argue on what to eat:

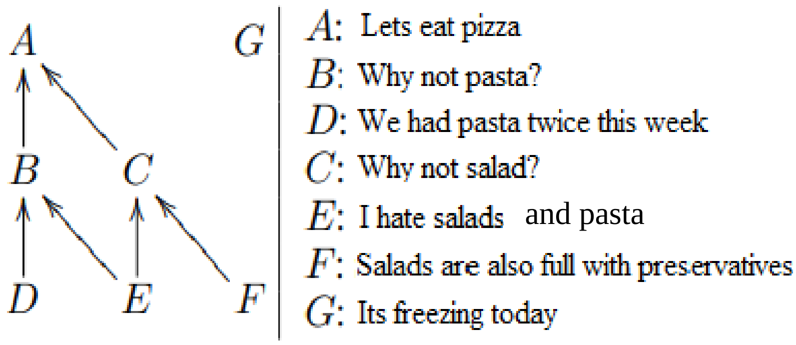

Figure 2.3. Admissible Arguments in Argumentation Graphs

From the graph, it is immediately apparent that arguments E defends A and thus it fully justifies argument A. However, D and F also justify A because F attacks C and D attacks B, which both B and C attack A. On the other hand, G is also a valid piece of knowledge although it has nothing to do with A. In this example two admissible sets can be found, A_{adm1} with {A, D, E, F} and A_{adm2} with {G}. Both A_{adm1} and A_{adm2} are conflict-free, and any attacks are countered by arguments within the appropriate set.

The trustworthiness of a resulting decision due to an admissible set is enhanced because there are no other arguments to contradict the admissible set's arguments. As a result, the integrity and robustness of the outcome is guaranteed in any way [16], [21].

Moving on, as discussed earlier, the argumentation framework also has the property to generate explanations from the admissible sets on how and what led to a decision. By iterating through the admissible set the explanations can be retrieved since all the arguments in the set have an explanation. As an additional step to improve the justification of an outcome, the explanations of each accepted argument can be put together in a humanly manner to structure a sentence or perhaps a paragraph using natural language processing.

2.6 Gorgias Framework

Gorgias is an implementation of a logic programming framework based on argumentation that combines both the principles of preference reasoning as well as abduction. From a technical perspective Gorgias can be considered as a modular meta-interpreter embedded on the top of the object-oriented Logtalk preprocessor using SWI-Prolog [1].

The major functionality of this framework is to provide answers to queries, Q(X), asked within a logic program using priorities on rules and constraints on variables and as the name suggests, it also make use of abductive predicates. The result generated by Gorgias given a query Q(X) is in the form of $<\Delta, A, C>$ where [1]:

1. Δ is a set of rules
    a. Where rule consists of label, head and body, rule(Label, Head, Body)
2. A is a set of adducible hypotheses
3. C is a set of constraints on X and variables in $\Delta \cup A$

Gorgias has the potential to make use of reasoning in dynamic environments even with the lack of information (knowledge) while using policies that can adapt based on certain preferences. Therefore, it has also the ability to model higher level principles by considering the personal preferences of the user. As a result, different users can define policies of their own adjusted on their individual preferences to fulfill their personal requirements. For example, a cognitive “Notification Assistant” can be instructed by user X to generally allow any notifications but during the night to block any notifications, while user Y can instruct its assistant to generally block any notifications but during the weekend to allow the notifications to pass through.

Gorgias is the main argumentation framework that was used in the development of the problem stated in this thesis with regards to the contraindications.

Chapter 3

Background Theory in Medicine

3.1 Clinical Decision Support Systems

The amount and the quality of clinical data has exponentially increased during the last decades, including more disease registries as well as more electronic health records. Although digitalization and big data can be very beneficial does not necessarily mean that better patient treatment and care are offered [4]. However, "Clinical Decision Support Systems" can be used to harness the vast amounts of data and to transform them into meaningful information for health care and health services to thrive in this digitized world.

3.1.1 Overview

To begin with, "Clinical Decision Support Systems", (CDSS), can be considered as computer systems and on some occasions, computer clusters, designed to impact the clinical and medical diagnosis of patients while improving health care delivery [10]. To enhance the system's capabilities, the data of the patient are often linked with an electronic knowledge base commonly known as database [22]. Clinical knowledge and patient information are combined to produce an outcome which is going to be used in a later stage by an expert to enrich and reinforce its medical decision-making process.

Clinical Decision Support Systems can be divided in two types with regards to the algorithm they use. To be more precise implementations of CDSS utilize either Argumentation or Machine Learning algorithms [10] [4] [22].

The first type is referred as "Knowledge-Based" Clinical Decision Support System, (KBCDSS). This approach utilizes the principles and functionalities of Argumentation Frameworks, with rules acting as claims while the strongest set of arguments is going to be the admissible set which will be presented as the winner.

During an early stage in development, these systems used to be designed to imitate human thinking and to simulate the decision-making process that an expert in the field often uses to diagnose a patient. However, in the last two decades the goal and the aim of KBCDSS alter and a different approach is now in use. Currently, the user is expected to interact much more with the system as well as to be more active as opposed with the earliest versions where the user was passive. In addition, the system is expected to be used as an assistant for the doctor to exploit during its decision making as well as to provide information for the clinician to use, rather than to come up with a solid answer like it was used to in the past [10].

The Knowledge-Based Clinical Decision Support Systems consist of three parts; the mechanism to communicate with the user, the reasoning engine, and the knowledge base [10]. The reasoning engine utilizes the concepts and principles of Argumentation while the knowledge base consists primarily of rules. This is the type of CDSS that is used in the context of this thesis.

Unlike “Knowledge-Based” systems which make use of the reasoning engine, the second type of systems utilize machine learning, a form of artificial intelligence, to recognize patterns in the data and can learn from past experiences. This type of systems is commonly referred as “Nonknowledge-Based” CDSS [10].

3.1.2 Architecture

The architecture of Clinical Decision Support Systems can vary based on their purpose as well as the algorithms and heuristics they implement in order to accomplish their goals. Although there is a wide range of CDSS architectures the one that is going to be used in this thesis consists of three major components:

1. “Communication Device”: Usually the personal computer of the user; can be a laptop or a desktop and even a mobile phone. It can be considered as the main interface where inputs from the user are given and the results are presented
2. “Engine”: The median between the inputs and outputs. Can combine both new information with stored information which can be located in the database. An algorithm or a heuristic can use the information to generate the output.
3. Database: Consists with all the available data that can be found not only in the knowledge base of the system but also in the patient record history

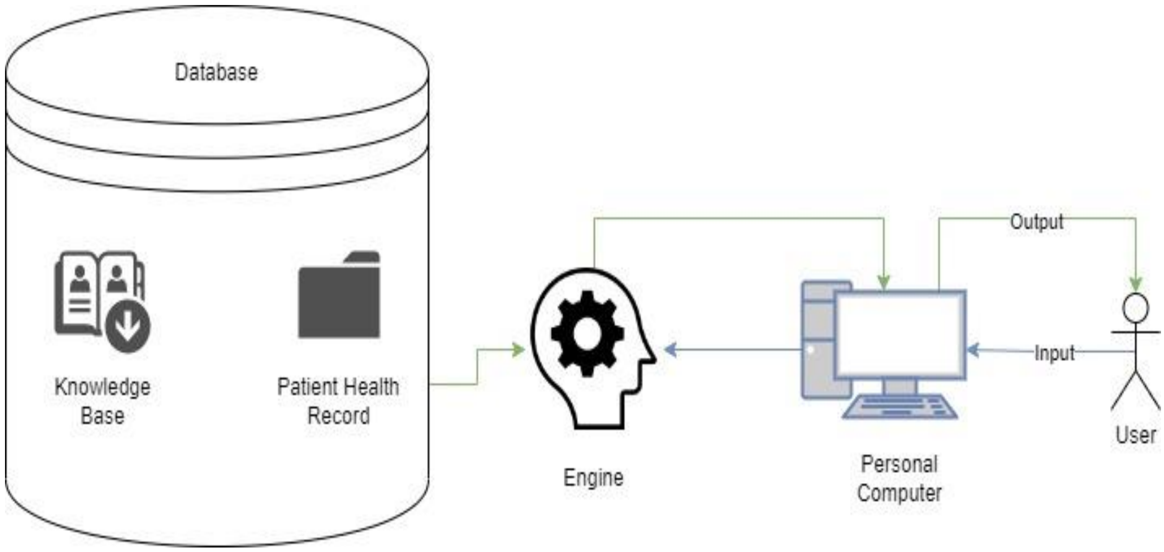

Figure 3.1. Clinical Decision Support System Architecture

3.1.3 Functionalities and Applications

The functionalities of CDSS can be categorized based on two different aspects. The first aspect is the user prompting. The system has the potential to notify, remind and alert the user about critical information and certain events during its interaction with the software. In addition, a very important functionality, which is related with the second aspect of such systems is their ability to provide medical and clinical diagnosis. The presenting complaints and relevant patient record information of an individual can be interpreted by the system which will then suggest possible answers (diseases or medications) during a checkup [22].

A wide range of services and applications includes:

- Medical and clinical diagnosis [24]
- Medication prescription and treatment [24]
- Alarms and reminders for disorders related with emergency conditions that need immediate treatment [22]
- Order facilitators for laboratory tests [22]

3.1.4 Guidelines and Design Principles

It is vital for Clinical Decision Support Systems' success that important considerations are taken into account during their development and design phases. Thus, developers must strictly follow a set of guidelines and principles to ensure the acceptance of such systems from the medical community as well as their harmonious integration into hospitals and clinics.

First and foremost, because of the unique nature of health data, a clinical decision can be influenced by a variety of factors. For example, physical, clinical or laboratory examination results as well as the patient history can alter the outcome of a diagnosis.

Therefore, the system must be able to combine these different forms of information and knowledge before concluding to its final diagnosis [24].

Moving on, a CDSS should imitate the cognitive process and abilities of a human. The enormous amounts of data stored into the system's knowledge base can be combined with the methodology and thinking process of an expert in the field. As a result, its reasoning will be pure human-like and a better differential diagnosis can be achieved [24].

As an addition to the aforementioned, knowledge must be gathered from reputable sources and potential biases and outdated resources must be avoided. Additionally, validation process must be performed once the rules are coded into the system's "brains". The validation procedure should not only extend to test whether the rules fire appropriately but also to whether the rules themselves as knowledge are appropriate and valid [10].

Finally, once the system is implemented and a thorough testing process is performed, the knowledge base of the system should be updated in a timely manner. As more and more diseases are discovered and new methods of diagnosing a disorder are coming to the spotlight it is important that the system will keep up to date with the recent discoveries on the matter [10].

3.1.5 Related Work

Nowadays, there is a handful of available CDS systems for the public to use. Some of them are free while others require only email registration and validation. Individuals can access these systems via the internet or their mobile devices without the need to visit a health care center. However, their existence is not to substitute the doctors and the health service providers but to act as tools for humanity to benefit from.

For example, "Symptoma Digital Health Assistant" is designed to provide its users with information about possible causes of their symptoms [26]. In addition, Symptoma's chatbot has the ability to engage into conversation with the user to ask for more relevant symptoms to narrow down its search and to be more “confident” about the diagnosis.

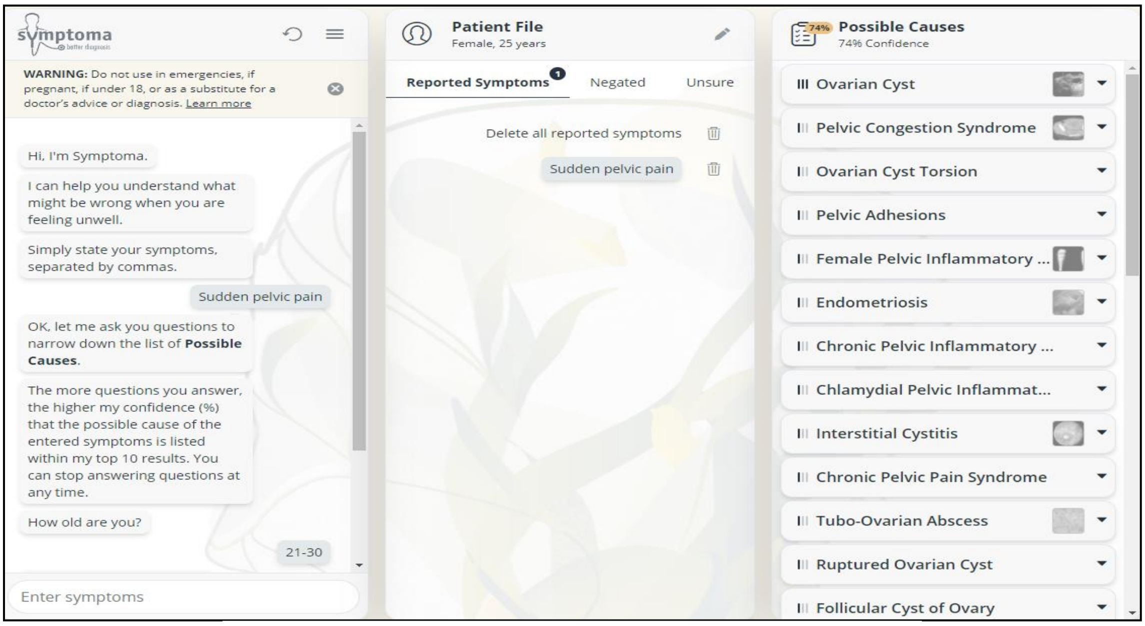

Figure 3.2. Symptoma
(https://www.symptoma.com/en/medical-device)

Another similar tool is “Symptomate” which claims that can provide individuals with fast and accurate health assessments based on their complaints. Additionally, it has the ability to ask the patient-user simple questions related with the initial symptoms to enhance and enrich the diagnosis [25].

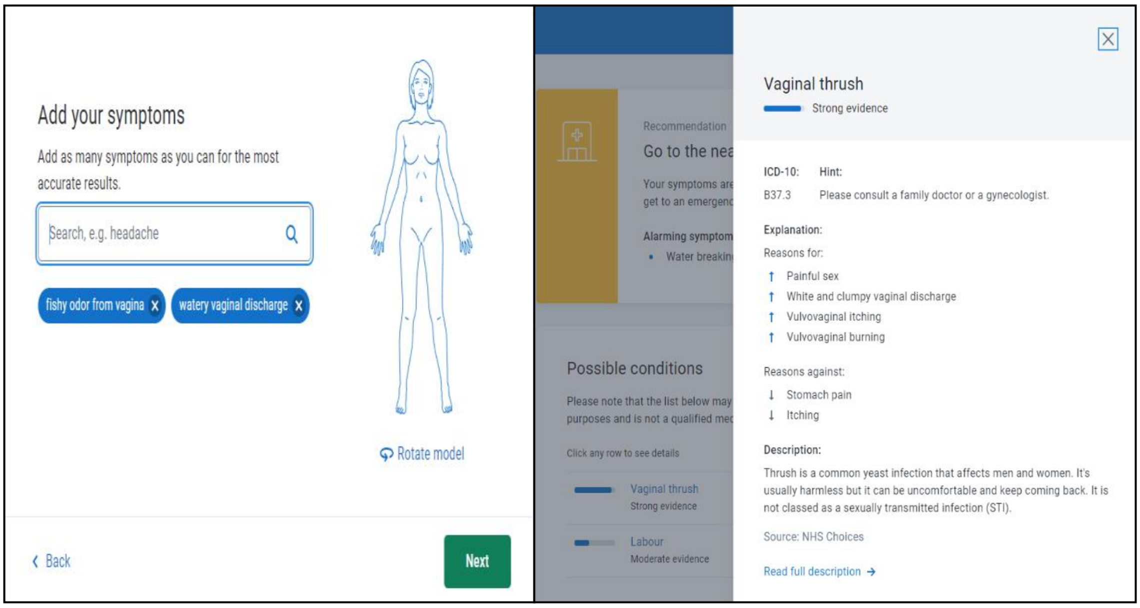

Figure 3.3. Symptomate
(https://symptomate.com/)

Last but not least, “Symptom Checker” is another example of CDSS where the symptoms of the user can be inserted while the diagnostic procedure will run in the background to provide the patient with a set of possible diseases. If a potential disorder is considered as a “red flag” the system alarm the user to seek immediately medical advice.

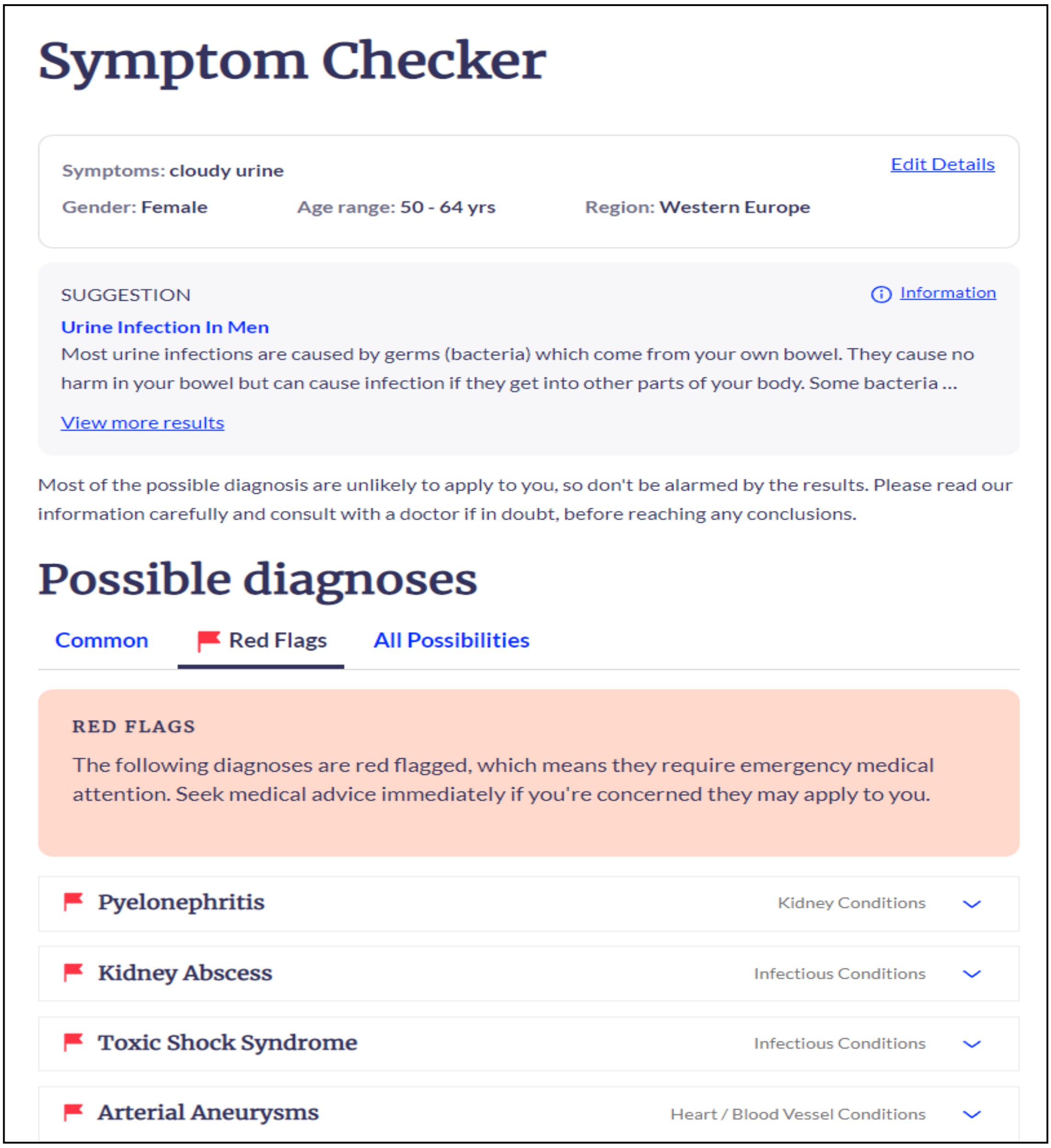

Figure 3.4. Symptom Checker
(https://patient.info/symptom-checker)

3.2 Gynecology

Gynecology or Gynaecology is a field of medicine which specializes in the female reproductive system and includes the ovaries, the fallopian tubes, the uterus as well as the breasts. A doctor with a specialization in gynecology is commonly referred as "Gynecologist". They can offer diagnosing and treating services for disorders relevant with the reproductive system of a women [12].

Gynecologists have mastered the necessary skills to perform any kind of examination related to their field [30]. To begin with, they can conduct physical examinations for the breasts, pelvis, vagina etc. to inspect and find possible abnormalities. Moving on, they also have the knowledge and the expertise to carry out screenings or imaging tests such as ultra-sound which includes transvaginal or endovaginal ultra-sound, abdominal or pelvic ultra-sound as well as breast screenings such as mammography. In addition, they can perform laboratory tests such as, culture collection for biopsies (breast, cervical, endometrial etc. [29]), PAP test and many more.

Moreover, gynecologists are also eligible to offer services that aim towards the diagnosis and treatment for disorders relevant with the female reproductive tract. Some experts in the field are called as obstetrician-gynecologists or "OB-GYN" and can provide care during pregnancy or childbirth [12]. Finally, they can also act as consultants on topics with regards to sexual health, birth control as well as menstruation and contraception.

Chapter 4

Theoretical Model

4.1 Existing Model

The existing model that is going to be illustrated and expanded to some extend is based on the principles and fundamentals explained earlier in the theoretical chapters two and three.

To begin with, the system was built using the argumentation framework of GORGIAS and follows the guidelines proposed for clinical decision support systems. The knowledge base of the system that is associated with gynecology was provided by an obstetrician-gynecologist. To ensure that any information was beyond the shadow of a doubt, the expert himself not only gathered but also evaluated the information to ensure their integrity and its world-wide acceptance.

The system covers a wide range of disorders related with gynecology, almost one hundred and forty (140), and each and every one can be used during the diagnostic procedure. They are divided into nine (9) categories / groups and some of the traits and criteria used to separate them into their relative groups was:

- The nature of the disorder (e.g transmitted upon sexual contact)
- The organs as well as processes related with the female reproductive system that are affected by the disorder (e.g endocrinology, vulva)

Group Name	Brief
1. Bleeding and Vascular Problems	Conditions and disorders that occur in the blood (e.g coagulopathy, Von Wilerbrand) as well as vascular disfunctions and problems (e.g leg varicose veins, thrombocytopenia)
2. Endocrinology	Conditions and disorders that affect the production of hormones as well as the malfunction of the organs producing them such as ACTH secreting, hyperinsulinemia etc. (elevated levels of hormones called cortisol and insulin respectively)
3. Gynecological Cancers	Tumors that are evaluated as cancerous and are observed in the female reproductive system, for example, ovarian and vaginal cancer
4. Infertility	Conditions and disorders related with the inability of a woman to conceive, such as primary and secondary infertility
5. Pelvic Pain	Conditions and disorders that cause pain on the pelvis (bellow belly button and above the legs) such as appendicitis, salpingitis etc.
6. Sexually Transmitted Infections	Conditions and disorders that pass from one person to another during sexual intercourse (e.g chlamydia, anogenital warts etc.)
7. Urogynecology and Prolapses	Conditions and disorders related with both the reproductive and urinary system (e.g cystocele, pyelonephritis etc.) as well as prolapses of those organs such as uterine prolapse
8. Vulva Pathologies	Conditions and disorders related with the outer part of the female genitals, the vulva, for example VIN and VAIN
9. Sundries	Conditions and disorders that do not fulfill the criteria of the aforementioned groups

Table 4.1. Disorders by Group

It is worth mentioning that a disorder may belong in more than one groups based on its nature and the potential symptoms that can be caused. For example, nephrolithiasis commonly known as kidney stones, belongs to the Urogynecology and Prolapses group as well as to the Pelvic Pain group due to the lower and central pelvic pain that can be caused to patients suffering from it.

Moving on, the theoretical decision-making process of the existing model is based on the same natural way that experts in the field follow during a routine checkup. First and foremost, the patient record information is collected or imported from previous checkups (e.g demographics, family history, past operations, medications etc.) and will be then combined with the second stage's information, which include the presenting complaints of the patient. Then, clinical tests can be performed by the doctor and the results can be imported into the system to enrich its knowledge and enhance its diagnosis.

During each step of the diagnostic procedure of the system, its radar focuses only on relevant disorders and the search can either be narrowed down or broaden with the addition of each new piece of information. The system also has the ability to alert the user (doctor) for possible emergency conditions that may need immediate treatment.

Additionally, the system mitigates the ability of a human; to interpret information in order to extract high-level concepts also known as common sense knowledge. For example:

- Given the height and weight of a patient higher level concepts can be produced by the system such as "overweight" or "obese" using the formula of Body Mass Index (BMI). In some cases, like ovarian cancer obesity is considered as a major risk factor.
- Given the age of a patient the system can determine the group(s) that the patient belong. For example, if the age of a patient is greater and equal than twenty (20) and less than forty-five (45) then the patient is considered as an "adult". Assumingly that the age of the patient is thirty (30) years old it is immediately apparent by the system that the patient is during the "reproductive age" (more than eleven (11) and less than fifty-two (52) years old). Thus both "adult" and

"reproductive age" are interpreted by the system by only using the age of the patient.

Finally, and most importantly, explanations are offered by the system and are used to justify why diseases are considered to be possible, in other words the system has the potential to account for its decision-making procedure. From a theoretical perspective, the admissible argument set, which includes the argument(s) that support the possible disorder, is processed by a natural language heuristic (algorithm) to generate a "colorful" explanation in as humanly way as possible.

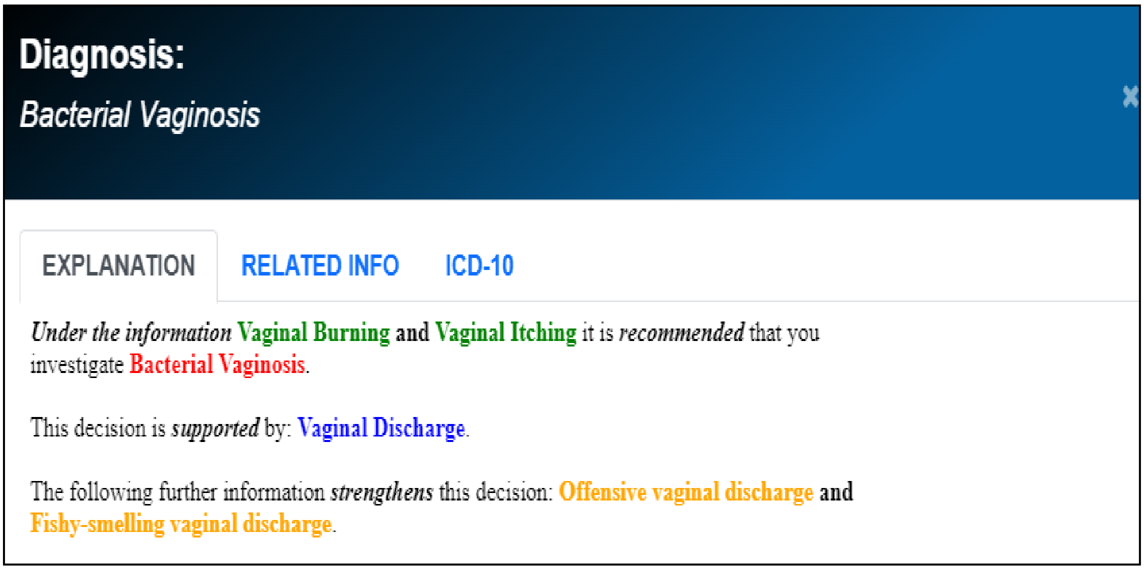

Figure 4.1. System's Explanation

To conclude, the goal of the system is to recommend a set of possible conditions or disorders, which are relevant, rather than to pinpoint the exact one. There are several reasons behind this rational but the most important one is to cast light to diseases that are not common but on the other hand cannot be excluded just because their unique nature. Given a certain set of symptoms these uncommon disorders can be triggered and highlighted if considered relevant by the system to help doctors avoid any pitfalls with regards to uncommon disorders. Therefore, what is usually referred as "Differential Diagnosis" is also achievable with this implementation.

4.2 Architecture

An important characteristic that has the potential to influence not only the usability but also the speed of decision-making systems is their architecture. An in-depth explanation of the major components which are contained in the architecture of the implemented system is going to follow.

To begin with, this diagnostic tool is hosted in a website to make the user experience easier and to provide accessibility to a variety of machines, regardless their operating system, such as laptops and desktops which have an internet connection and a web browser. Ideally a one-to-one communication between the patient and the doctor during the information collection procedure is the primary goal of the system. In other words, the system shall provide the doctor with the ability to interact with both the patient and the system at the same time conveniently, by getting information about the complaints, symptoms and concerns of the patient and use them as an input in the system.

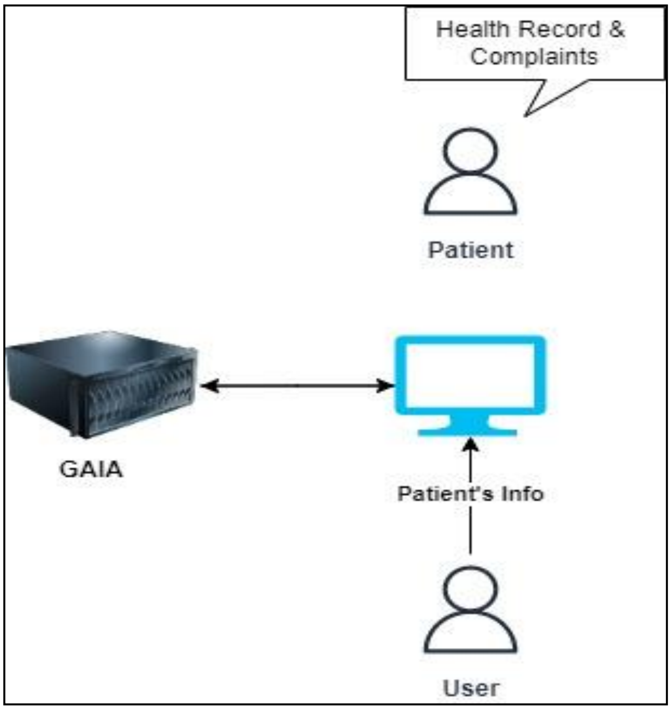

Figure 4.2. Abstract Architecture – Communication and Information Gathering

Moving on, in order for the diagnostic procedure to be executed, both new and old information with regards to the patient need to be gathered together. The old information is used to enhance the diagnosis of the system and it may consist with relevant patient record information such as medications, habits (alcohol abuse, excessive smoking), family history etc. The engine of the system act as the median which concatenates both new and old information together and send them to the "knowledge base" in order to generate the diagnosis.

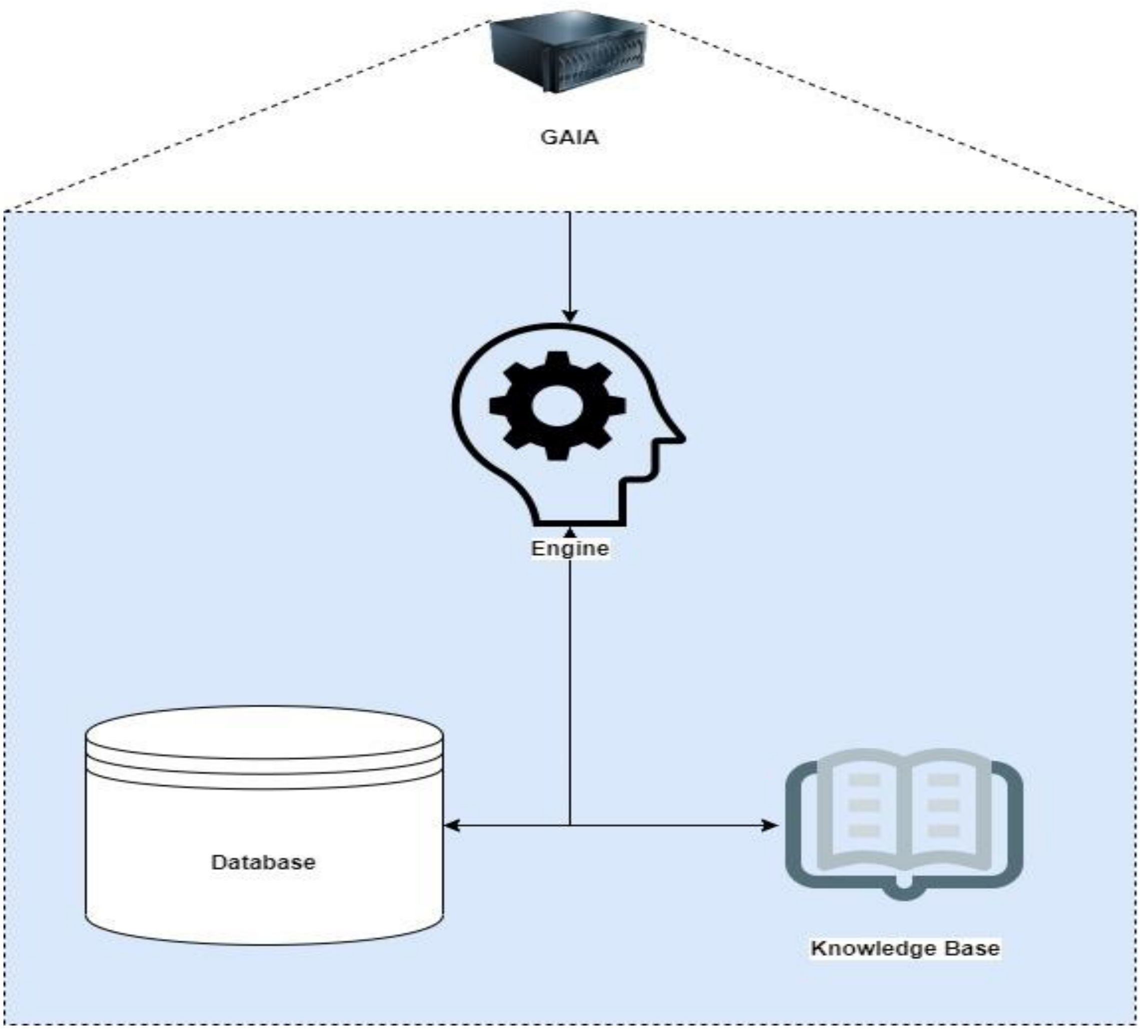

Figure 4.3. Abstract Architecture - GAIA

4.3 Model Expansion

Although the existing version of the system has a robust architecture as well as an elegant and optimized approach to the diagnostic procedure there was still some room for improvements. During a checkup, a senior doctor often uses information to contradict the likelihood of disorders being possible and thus can be excluded. As a result, the initial list of disorders can be shrunk and the attention can be turned elsewhere. This is exactly what lacks from the system and if implemented new possibilities and potentials will arise for the user to benefit from.

4.3.1 Contraindications

"Contraindication" is commonly referred to the medical reason, that can either be a symptom or perhaps a patient record information but also a laboratory or a clinical test, for not giving to an individual a particular drug or a medical treatment. Alternatively, those medical reasons can also be considered as a way of diagnosing a patient because by definition, they eliminate the chance of an individual suffering from a disorder. Hence, a smaller subset of the initial disorders set is left for the expert to focus on and decide. Three different types of contraindications will be implemented and each of which has its own characteristics.

1. An "Absolute" contraindication should be considered as a standalone and strong argument where if exists can eliminate the possibility of a patient having a certain disorder. For example, a non-pregnant patient cannot suffer from ectopic pregnancy or any other disorders that can occur during pregnancy (e.g placenta abruption, placenta previa). Thus, in this example "non-pregnant" is a contraindication for the disease of "ectopic pregnancy" and can be considered as an "absolute" one.
2. An "Almost Absolute" contraindication also refers to a strong argument but not as strong as the one presented on the absolute contraindication. If a medical reason exists and it is evaluated as an almost absolute contraindication it means that; there is a slim chance of a patient having a disorder even if this piece of information exists. For example, in seven out of ten cases (70%) of "Corpus

Luteum Insufficiency" patients discovered to have "hyperprolactinemia". Thus, "not" having "hyperprolactinemia" can be considered as an "almost absolute" contraindication for "Corpus Luteum Insufficiency".

3. A "Context Based" contraindication can be defined as an argument that can only exist if and only if another argument exists as well. This type of contraindications is strong and if exist can eliminate the chance of a patient having a certain disorder. Therefore, the contraindication, as the name suggests can exist depending on the context. For example, for "Delayed Puberty" you need both arguments "age less than fifteen (15)" and "normal breast growth" to form a contraindication that will evaluate the disorder as not possible. So, the combination of "age less than fifteen (15)" and "normal breast growth" can be considered as a "context based" contraindication for "Delayed Puberty". In other words, under the context of "age less than fifteen (15)" and "normal breast growth" this disorder is not possible. However, if only one of those two arguments is known then the contraindication is not solid and "Delayed Puberty" cannot be excluded.

The information regarding the contraindications was gathered in the form of tables with the use of reputable sources [31] [32] [33]. Then they were thoroughly reviewed and expanded by a senior doctor with years of expertise in the field of obstetrics and gynecology. Following this, one to one conversation where then performed to evaluate for a final time the trustworthiness of the gathered knowledge.

For each of the nine group of disorders a dedicated table with the following features was created to speed up and tide the process due to the large number of disorders supported by the system.

<table>
<tr><th colspan="5">GROUP X</th></tr>
<tr><th>Disease</th><th>Contraindication</th><th>Medical Reason</th><th>Strength – Type</th><th>Explanation</th></tr>
<tr><td>D_1</td><td></td><td rowspan="3">Patient Record
/
Symptoms
/
Clinical Examination Findings
/
Laboratory Test Results</td><td rowspan="3">Absolute
/
Almost Absolute
/
Context Based</td><td></td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
<tr><td>D_n</td><td></td><td></td></tr>
</table>

Table 4.2. Contraindication Table - Structure

More than one thousand two hundred and fifty (1250) rows were created to host this enormous number of contraindications. An in-depth example and explanation of the rational of the tables will be illustrated next.

Disease	Contraindication	Medical Reason	Strength – Type	Explanation
D1	C1	Patient Record	Absolute	-

Table 4.3. Contraindication Table – Example 1

From the first example it can be observed that C1 is an absolute contraindication for D1 and it is an information related with the individual's patient record. For example, consider D1 as "MRHKS" (Mayer-Rokitansky-Küster-Hauser Syndrome) and C1 as "normal menstrual cycles". In this scenario if the patient's menstruation is normal then MRHKS can be excluded.

Disease	Contraindication	Medical Reason	Strength – Type	Explanation
D2	C2 AND C3	Patient Record And Symptoms	Context Based	-

Table 4.4. Contraindication Table – Example 2

However, in the second example both C2 and C3 are needed to contradict the possibility of D2. We can further observe that C2 or C3 on their own cannot be considered as a valid contraindication and if and only if both exist their unification has the potential of eliminating D2. For instance, let us assume that D2 is "intussusceptions", C2 is "child" and C3 is "no sudden abdominal pain". Under the context that the patient is a child, absence of sudden abdominal pain, can disprove the likelihood of intussusceptions.

For simplicity reasons for the current scenario, "Medical Reason" is going to be left as empty and can be considered as any of the possible options showed in Table 1.

Disease	Contraindication	Medical Reason	Strength – Type	Explanation
D3	C4	-	Almost Absolute	-
	C5	-	Absolute	-

Table 4.5. Contraindication Table – Example 3

In this example we have C4 and, or C5 which can be used to exclude D3. Although C4 can be used on its own to contradict D3 there might be scenarios where this is not the case. However, if both exist, C5 due to its "absolute nature" can support C4 forming an alliance to eliminate D3. In a hypothetical scenario where D3 is "Vulva Cancer", C4 is "younger than 50 years old" (patient record) and C5 is "vulvar biopsy – culture shows negative results" (laboratory test) can be used on their own to contradict cervical cancer but if combined can form a stronger contraindication to remove any doubts.

Assuming that D1, D2 and D3 are disorders relevant with GROUP Y then the final table for this group should look like this:

GROUP Y				
Disease	Contraindication	Medical Reason	Strength	Explanation
D1	C1	Patient Record	Absolute	-
D2	C2 AND C3	Patient Record And Symptoms	Context Based	-
D3	C4	Patient Record	Almost Absolute	-
	C5	Laboratory Test	Absolute	-

Table 4.6. Contraindication Table for Groups

4.2.2 Contraindications as an Argumentation Problem

With the basic concepts of contraindications now covered and explained, the next step is to model the "contraindications" since the goal of this thesis is to enrich the knowledge base and enhance the diagnostic procedure and decision-making capabilities of the existing system.

To begin with, let us define the tuple <D, P, C>:

- D is the set of disorders supported by GAID
    1. Consists of one hundred and thirty-seven (137) unique conditions and diseases

- P the set with disorders proven to be possible
    1. Subset of D generated from the first phase of the diagnostic procedure the "diagnosis"
- C the set with disorders proven as impossible
    1. Subset of P before altering generated from the second phase of the diagnostic procedure the "contraindications"

First and foremost, a set of various inputs such as patient record information, symptoms, clinical and laboratory results are gathered. Then, they are going to be used by the Argumentation Framework to execute the "diagnosis" that will generate the set of disorders, P, that have the likelihood of being possible. The same inputs can also be used in the "contraindication" procedure later on to from the arguments to contradict the existence of the disorders included in P.

The general policy used during the second phase is to consider all disorders of P as possible until the opposite is proven. Once the reasoning procedure is completed, a set will be constructed with the admissible arguments and the supported claims can exclude diseases from P and can be transferred to C. However, it is important to keep the basic design principles of the existing version of the system and not alter the architecture and methodology used in the diagnostic procedure.

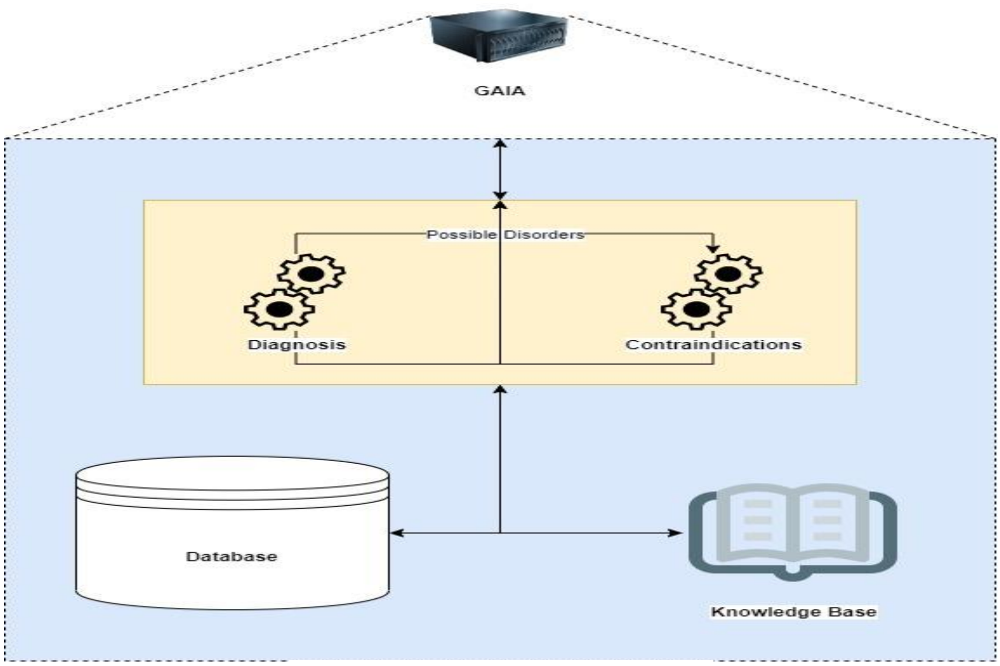

Figure 4.4. New Architecture

Chapter 5

Technical Model

5.1 Software Technologies

For the completion and implementation of the problem stated in this thesis a wide range of techniques were used and accompanied with different coding approaches due to the system's unique nature and variety of components.

Despite the fact that the webpage of the system was already implemented I contribute my bit as well; from fixing and improving certain functionalities all the way to creating new features for the user to use and benefit from. Various popular web development tools were used such as:

- HyperText Markup Language (HTML): Client-side language used for designing the structure and components of the pages
- Cascading Style Sheets (CSS): Client-side language used for the styling and the responsiveness of the pages across different port views
- JavaScript (JS) and jQuery: The first is a client-side language used for creating dynamic and interactive content (pop ups, alerts etc.) while the latter is a JavaScript library used for the enhancement of event-handling (click, type etc.)

- PHP: Sever-side language used in the diagnostic procedure for calling other processes (middleware) with regards to JAVA and Prolog.

Additionally, the Visual Studio Code (VS Code) integrated development environment (IDE) was used due to the wide-range support of languages and ability to use extensions that are publicly available.

For hosting the website, the following Apache Server technologies were used:

- Local Host: XAMPP and WampServer
- Online Host: The department offered two online repositories on a web server, one for development and one for production

Moreover, an extension of Prolog, a logic programming language framework, was used and is commonly referred as Gorgias. Thereby, the argumentation framework was developed with the use of Gorgias which allow programmers to write argument-based rules, create scenario-based preferences and then generate explanations which can be derived from the admissible set of arguments.

Finally, the Java programming language was used with the JPL library to connect the argumentation framework with the website. This library uses SWI-Prolog's and Java's interface to allow a bidirectional communication between them. Since Gorgias is an extension of Prolog it can also be used in Java with the JPL library for query executions and explanation collection from Gorgia's delta (admissible argument set).

5.2 Contraindication Tables Implementation in GORGIAS

All the important and necessary information with regards to the various contraindications have been collected in the format of tables, creating the "Scenario Tables", as mentioned earlier in the fourth chapter of the thesis. The structure of the tables allows the programmer to define an easy-to-use methodology and translate the text, which is in formal language, to code which is required for Gorgias to run and execute.

Before modelling the tables using the framework of Gorgias, decision options and object-level rules are needed to be implemented first as well as a general policy.

The “Decision Options” are the possible results that can derive from the argumentation procedure. In other words, they are the decisions that can be chosen by Gorgias with the winner having the admissible argument set. Within the frame of contraindications, we have two options for Gorgias to “choose” based on the given knowledge:

- A disease is possible (poss(D))
- A disease is not possible (not_poss(D))

where “D” is the relevant disease.

To introduce the counterfactual state between those two options, “complement” was used to implement that a disorder can either be possible or not possible but not both.

- *complement(poss(D), not_poss(D)).*
- *complement(not_poss(D), poss(D)).*

Moving on, the “Object-level Rules” can be considered as a set of rules that are used to model whether each disease is possible or not, given an information which may include symptoms, patient record etc. For every disorder marked as possible from the diagnostic procedure we create dynamically these rules using the decision options mentioned above:

- *rule(possible_disorder(D), poss(D), []).*
    - *“Disorder D is possible”*
- *rule(not_possible_disorder(D), not_poss(D), []).*
    - *“Disorder D is not possible”*

The “General Policy” used by our agent was built in such a way to make use of the already known information that any disorder that is going to be examined in the contraindication procedure is initially possible due to the first phase performed by the diagnosis part of the argumentation.

- *rule(policy(D), prefer(possible_disorder(D), not_possible_disorder(D)), []).*

For each disorder a file was created to host the possible contraindications that can exclude the possibility of being possible. In each file only the relevant rules were implemented with regards to each disease and some of their characteristics are:

1. The name of the rule starts with the type of the contraindication
    a. symptom
    b. patient_record
    c. clinical_exam
    d. laboratory_test
2. Followed by the ID (number) of the contraindication

For example, let us assume that for the first contraindication we have disorder D and a laboratory test argument A that can exclude D. The rule will be:

- *rule(laboratory_test_cr1, prefer(not_possible_disorder(D), policy(D)), []) :- A*

With this way, we advise Gorgias to prefer that the disorder D is not possible over the general policy, which supports that D is possible, given a set of argument A. However, to avoid this line of code being executed by all of the disorders, in the argument set A this argument a_1 must be written:

- (D = name of the disorder)

For example,

rule(clinical_test_cr3, prefer(not_possible_disorder(D), policy(D)), []) :-

noSmallCauliflowerLumps,

(D = anogenital_warts).

This example illustrates a contraindication regarding “Anogenital Warts”. It is immediately apparent that both arguments in A must exist to disprove the general policy and eliminate the possibility of it. Additionally, in the same file, multiple rules can be written in order to implement all the available contraindication information of each disorder.

Furthermore, in the argument set A, multiple arguments can be listed separated with a comma to implement the conditional AND.

For instance,

rule(laboratory_test_cr36, prefer(not_possible_disorder(D), policy(D)), []) :-

no_labTest(highT3),

no_labTest(highT4),

no_labTest(highTSH),

(D = hyperthyroidism).

For the elimination of “Hyperthyroidism” three (3) hormone readings from blood sample need to be taken; T3, T4 and TSH. The results must be less than the allowed highest limit with regards to each hormone to discard hyperthyroidism. One can observe that three (3) arguments are listed, plus the argument that let us identify that we are talking about hyperthyroidism and are separated by comma.

Now the file implementing the contraindications for “Neisseria Gonorrhoeae” disorder will be illustrated in which all of the techniques mentioned above are used.

```
%------------------------------------------------------------------------------------------
% SCENARIO DESCRIPTION: Neisseria Gonorrhoeae
% RULE: patient_record_cr10(D)  - contrainidication rule 10 - context based (100%)
% RULE: laboratory_test_cr11(D) - contrainidication rule 11 - absolute (>thershold)
% RULE: laboratory_test_cr12(D) - contrainidication rule 12 - absolute (>threshold)
% PREFERENCE OVER: Any other rule

% Notes:
%------------------------------------------------------------------------------------------

% If contraindication exist
rule(patient_record_cr10, prefer(not_possible_disorder(D), normal_policy_rule(D)), []):-
    mother(not_neisseria_gonorrhea_carrier),
    sexualHistory(never_had_sex),
    (D = neisseria_gonorrhoeae).

rule(laboratory_test_cr11, prefer(not_possible_disorder(D), normal_policy_rule(D)), []):-
    no_labTest(pcr_neisseriaGonorrhoea),
    (D = neisseria_gonorrhoeae).

rule(laboratory_test_cr12, prefer(not_possible_disorder(D), normal_policy_rule(D)), []):-
    no_labTest(urine_neisseriaGonorrhoea),
    (D = neisseria_gonorrhoeae).
```

Figure 5.1. Contraindication File – Neisseria Gonorrhoeae

5.2 Contraindication Recommender Engines Implementation in JAVA

As discussed earlier, the system was not only implemented to be used by senior doctors but also by novice and resident doctors. Having a system that can use a piece of information to exclude disorders and narrow down the diagnosis is very important. However, the possibility to suggest any contraindications with regards to the relevant disorders that may be presented on the doctor's radar is a necessity to have if we want to create a robust and convenient system. As a result, the idea of a recommender engine was suggested for the user to benefit from.

The initial goal was to help a novice doctor that does not have as much experience as an expert doctor has. With only one click, a list of possible contraindications is going to be presented for the doctor to consider and with an additional click they can be used as an input and be inserted in the reasoning of the system.

To achieve this ambition, a list of all the possible contraindications listed by the name of their rule was generated in Java using the data structure of a hash map. Once the contraindication procedure, mentioned above, was done with the execution, Gorgias returned the name of the rules that were included in the admissible argument set. The rules that were not mentioned in the admissible set are going to be the ones to be presented in the recommender engine.

However, instead of one recommender engine proposing all medical types of contraindications (symptoms, patient record, clinical exam, laboratory test) to the user, was decided in a later stage to make use of two distinct engines. One for proposing the symptoms, patient record and laboratory test related contraindications and one for proposing only those relevant with clinical examinations. The rational was based on the method that doctors follow during a diagnostic procedure which includes the collection of the symptoms and patient record information before proceeding to any clinical examinations.

The hash maps were a necessary data structure to use to be able to map the name of the rule to the medical reason of the contraindication with $\Theta(1)$ time complexity. Generally, a hash map consists of a list of items with the key-value pair property. In our case the key, which must be unique, is the name of the rule while the value is the medical reason of the contraindication

As an additional feature the search engines have the ability to change their button's color. If a suggestion is available, with regards to the button type, the color of the button will be orange, otherwise it will be white and vice versa.

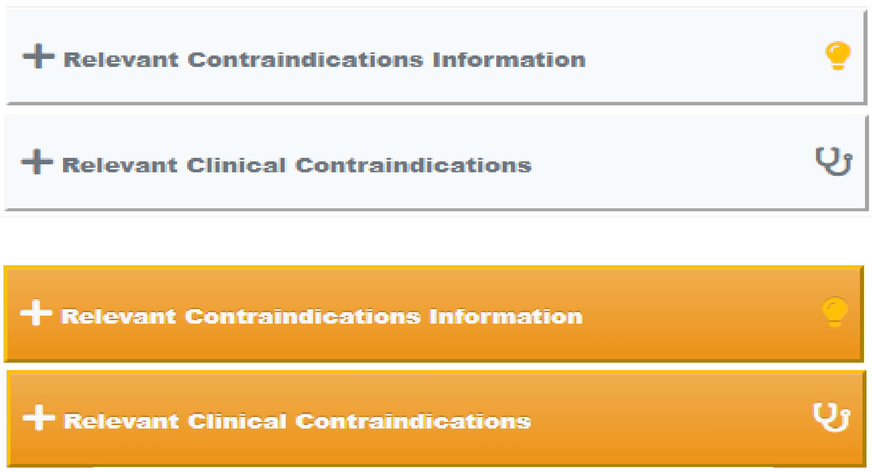

Figure 5.2. Search Engine Button States

As for the suggesting part, those buttons can be clicked at any given time and a pop up will appear with the relevant information to be chosen or perhaps to be asked to the patient.

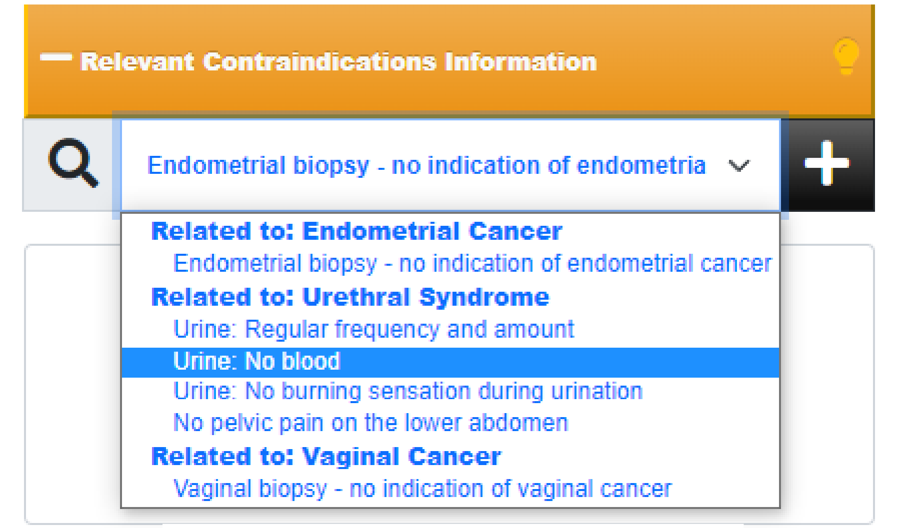

Figure 5.3. Search Engine Suggestions

5.3 Explanation Generation and Enhancement

One of the most important advantages that can be offered by a system that uses the Argumentation Framework is their ability to generate reasonable and understandable explanations to justify their actions.

Generally, the construction of a solution for a query that is given to Gorgias starts from the initial argument and the addition of any suitable arguments in its defend. If the argument set defends itself against each attack, then it is admissible. The admissible argument set is stored in Gorgia's Delta which can derive with the use of the "prove" function. The prove function takes two arguments, the goal and the delta thus "prove(Goals, Delta)". In our case when we want to prove that a disorder D is not possible, we execute:

- *prove([not_poss(D)], Delta)*

Assuming that we want to eliminate the possibility of “Chlamydia” and in the system’s knowledge base we have the following information:

1. *mother(not_chlamydia_carrier)*
2. *no_labTest(urine_chlamydia)*

As well as the following three rules that can exclude chlamydia:

rule(patient_record_cr5, prefer(not_possible_disorder(D), policy(D)), []):-

mother(not_chlamydia_carrier),

sexualHistory(never_had_genital_contact),

sexualHistory(never_had_sex),

(D = chlamydia).

rule(laboratory_test_cr6, prefer(not_possible_disorder(D), policy (D)), []):-

no_labTest(pcr_chlamydia),

(D = chlamydia).

rule(laboratory_test_cr7, prefer(not_possible_disorder(D), policy(D)), []):-

no_labTest(urine_chlamydia),

(D = chlamydia).

If we run *prove([not_poss(chlamydia)], Delta)*, in the variable Delta the admissible set will be returned with the following information:

1. laboratory_test_cr7
2. not_possible_disorder(chlamydia)

With the use of a hash map, again, we can in $\Theta(1)$ time complexity map the rule names with comprehensive explanation in natural language. As a result, the previous explanation given by the diagnosis can be enhanced by appending the explanation given by the contraindications procedure.

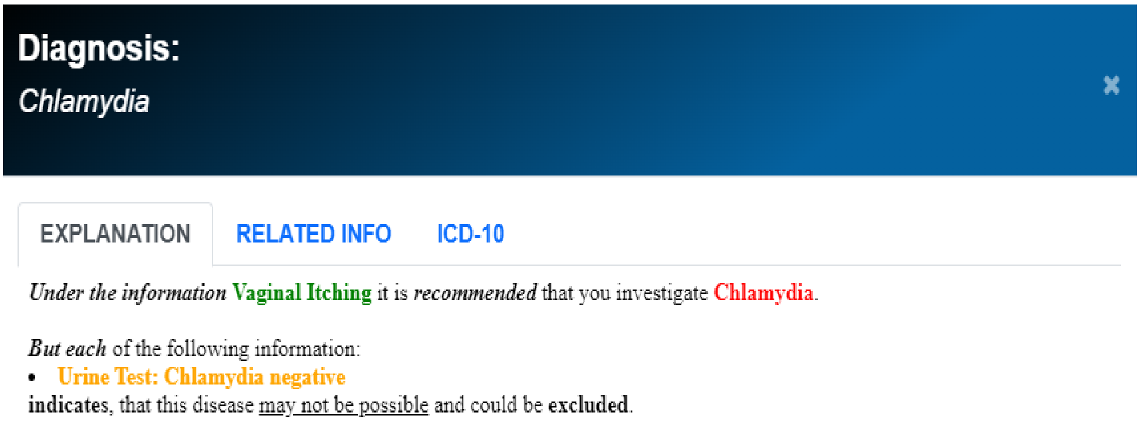

Figure 5.4. Explanation Enhancement - Contraindications

However, since we have in our possession the recommended contraindications from the recommender engine implementation, we can use them to further enrich the explanation.

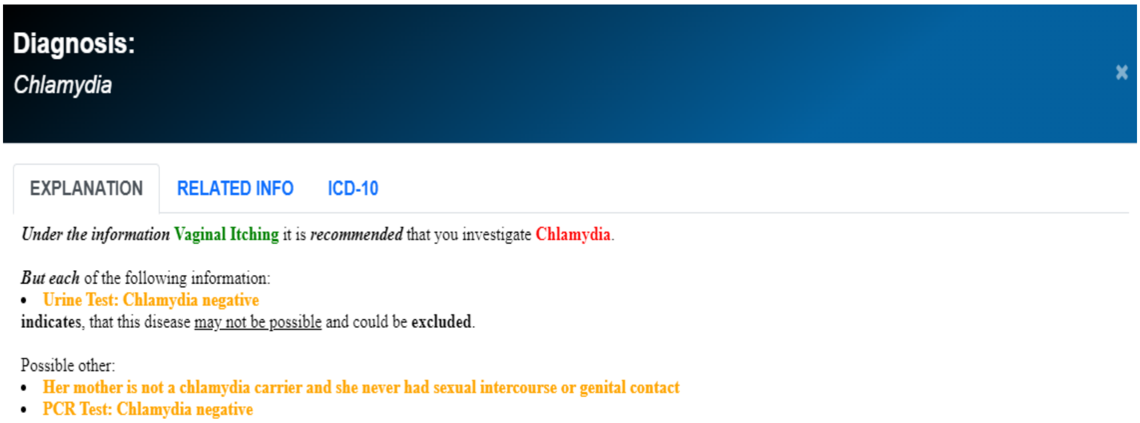

Figure 5.5. Explanation Enhancement – Contraindications and Suggestions

Chapter 6

Evaluation and Adaptability

6.1 Evaluation Methods

Software testing; is the process of verifying and evaluating that a software application does, what it is supposed to do. Throughout this phase bugs will be identified and in our case any gaps in the system's knowledge base will be pointed out. In this chapter, two (2) different evaluation approaches are going to be illustrated and discussed.

6.1.1 Past Real-Life Test Cases

To begin with, the first method that is going to be used to evaluate the system's accuracy involves the collection of past real-life scenarios. With the help of the same doctor that provided his knowledge and support during the contraindication collection, cases of real patient will be gathered and summarized in separate files based on the disorder. Generally, in this procedure five (5) steps are required to be performed for every test case, starting with the selection of a random disorder and the creation of its template file, and culminating in the calculation of the results and the collection of the doctor's feedback.

Before we proceed to the illustration of the procedure that is required in order to create a digitized test case, which is based on the doctor's file, a template file and some guidelines must be established first and strictly followed to protect any sensitive patient information. This will ensure that the subject's data privacy will be secured and will also help us to create and maintain a consistent record of cases for future reference.

First and foremost, during this procedure each test case will be created and hosted by the system and therefore the subjects are going to be registered to its database. Although some fields such as the ID, first name and last name are necessary for the creation of a new patient in the system, for this occasion only, they are going to be replaced with representative values which are going to be used for the identification of the case as well as to keep safe the actual patient's information. However, some other fields regarding the subject's date of birth as well as height and weight must be real because they are necessary for the commonsense module functionality, which translates them into high-level concepts that are going to be used as a part of the diagnosis. Thus, the first rule we must follow is to exclude the personal information of the subject and to substitute it with something along those lines.

Additionally, an equally important rule is to include only the information given by the doctor and not to add anything irrelevant or extra to boost or enhance the system's diagnosis. However, there might be some cases in where the given information does not exist in the system's vocabulary. For these occasions only, we are allowed to ignore their existence while a record is going to be kept with the missing knowledge, which are going to be included in a later version of the system.

Therefore, with that in mind the template file must be separated into different sections for the relevant information to be written. The first section will consist of the demographics of the patient followed by the relevant health record information. Then the initial complaints of the patient are going to follow with the doctor's first list of suspicious disorders. Before proceeding with the doctor's initial diagnosis, any additional symptoms and information with regards to previous clinical examination findings and laboratory test results are going to be included as well. Finally, any clinical examinations performed by the doctor during the checkup will be written and are going to be followed by the doctor's final diagnosis.

Patient's ID:

Demographics

- First Name:
- Last Name:
- Date of Birth:
- Height:
- Weight:
- Ethnic / Origin: -

Relevant Patient Record Information

- Patient Record Information - Example 1
- Patient Record Information - Example 2

Presenting Complaints

- Presenting Complaints - Example 1
- Presenting Complaints - Example 2

Doctor's First Suspicions:

Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.

GAID's suspicions: []

Additional Symptoms & Information

- Additional Symptom - Example 1
- Additional Patient Record Information - Example 2

Doctor's Initial diagnosis:

Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.

GAID's initial diagnosis: []

Clinical Examination - (Examination: Findings)

- Clinical Examination & Finding – Example 1
- Clinical Examination & Finding – Example 2

Final diagnosis:

Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.	Choose an item.

GAID's final diagnosis: []

Figure 6.1. Template's Structure

From this template one can observe that there are three phases which require the doctor's empirical diagnosis including any first suspicions, an initial diagnosis, and a final diagnosis. Apparently, the system diagnosis during each phase is going to be extracted as well and is going to be compared with that of the doctor at the current stage. Therefore, three metrics are going to be collected but more on that later.

Now, that we have a template file to work with, let us move with the steps that are involved during the testing procedure. The first step, which is an obvious one, is to randomly decide a disorder that we may want to work with from the list of the one hundred and thirty-seven (137) supported diseases. Once the disorder is decided, the file is going to be filled by the doctor with the relevant information of a past real-life visit. In the annotated file that is going to be created by the doctor any interesting events as well as any suspicions are going to be included and pointed out. Then a preprocess of the annotated file is going to be performed because the ID, first name and last name fields of the demographic section are going to be empty. As a result, those fields are going to be filled with relevant values regarding the disorder and are going to be used as identifiers for the test case.

Following that, the information and knowledge included in the file is going to be used to create a new patient in the system. Then, any patient record information such as medications or surgical history are going to be imported first. A new visit for the current patient is going to start and it may include the insertion of the information starting from the presenting complaints all the way to the clinical examination findings. At the end of each step, the system's diagnosis is going to be written to the appropriate section in the template file and in a later stage is going to be compared with that of the doctor.

Moving on, once the information insertion and diagnosis gathering step is done, the metrics are going to be computed and recorded in an excel spreadsheet. As a final stage, the template file, and the disorders at each part of the diagnosis are going to be discussed with the doctor and any extra disorders derived by the system's diagnosis are going to be marked as relevant or irrelevant and any disorders that are missing are going to be noted down along with the doctor's opinion on why they should exist.

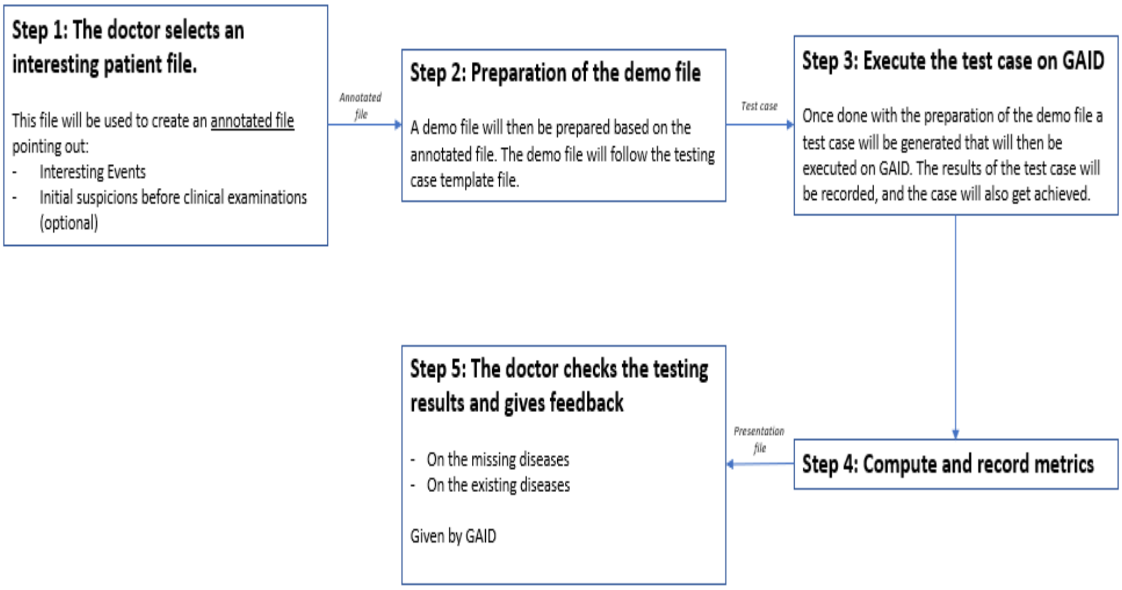

Figure 6.2. Testing Procedure

6.1.2 Pilot Clinical Evaluation

As opposed to the aforementioned method for evaluating the system, the second procedure involves cases which are going to occur during the checkup of the patient. Therefore, instead of past cases, we are going to have a series of events happening in the present. Additionally, in the first method we have two primary figures, the doctor, and the evaluator, however, during this procedure two doctors are going to be involved instead, one senior and one novice.

To be able to perform this evaluation method, the patient's consent must be given first. Additionally, any related procedures that involved the protection of the patient's privacy will be applied as well. Furthermore, the three phases in which the doctor's empirical diagnosis is going to be needed in order to make any comparisons with that of the system's, are going to be the same but with the slight difference; that are going to be exported one by one using the "Export Diagnosis" functionality of the system.

Moving on, during the checkup the senior doctor is going to perform the standard traditional way of patient examination, while the novice doctor will record all the information and patient answers in the system. When the point where the diagnosis needs to be compared is reached then the novice doctor will just simply export it for comparisons to be performed in the future. This is going to occur three times, during the initial suspicions and the initial and the final diagnosis which are also comprised by the same steps explained earlier.

It is important, that the results and the diagnosis of the system will be hidden from the senior doctor to avoid any biases during the diagnostic procedure that may lead into asking more questions with regards to a disease or perhaps altering the doctor's diagnosis. However, it is important for the novice doctor to have the opportunity to ask some further questions at the end of each step when it is required by the process.

6.1.3 Metrics

The metrics are very important during an evaluation procedure and are usually performed by comparing the predicted results with the actual results. A common guideline for such metrics should be; the higher the yield of the metric, the better the prediction should be [5].

Although there are various metrics that can be used for evaluation the most common are "accuracy" and "precision" [5]. Illustratively, when thinking about accuracy metrics, one can imagine a target which is used in archery. The closest to the bullseye the arrow lands, the better the accuracy of the archer. Generally, accuracy is used to measure how many data are predicted correctly [5]. In our case, accuracy is going to be used to score the system's diagnosis based on the doctor's by checking how many of the predicted disorders are considered as suspicious by the expert. On the other hand, precision is used to calculate the consistency of the system. For example, it can be used to calculate how often the system returns the same output based on the same inputs [5].

However, in our case the measurement regarding the precision of the system is not that necessary and useful due to the fact that the system utilizes its knowledge base and its rules. As a result, for the same inputs the same outputs will derive and there is not a meaningful reason to implement such metric.

The main metric that we are going to use to evaluate the system's capabilities is the accuracy. In our case, this metric will show how close is the diagnosis made by the system compared to the one made by the doctor. As discussed earlier, in each test case three diagnosis are going to be taken from the doctor and three from the system, and therefore this metric needs to apply three times.

Abstractly, we want to measure the relative accuracy with respect to the predictions of the senior doctor. The relative accuracy can be measured if we take the number of disorders that are common between the doctor and the system divided by the number of diseases suggested by the doctor. However, we also want to penalize the system depending on the number of diseases that are suggested. For this, a more complex approach is going to be used. We calculate first, the difference between the disorders suggested by the system and the disorders suggested by the doctor. We subtract this amount from the total number of disorders (137) and we divide it with the same number. Now both the first and second formulas are going to be combined to create the main accuracy metric of the system.

Main Accuracy Metric (Closeness):

$$\boldsymbol{Closeness} = \left(\frac{\boldsymbol{c}}{\boldsymbol{e}}\right) * \left(\frac{\boldsymbol{D} - (\boldsymbol{s} - \boldsymbol{e})}{\boldsymbol{D}}\right)$$

Where:

- s: number of disorders predicted possible by the system
- e: number of disorders predicted possible by the expert
- c: number of common disorders between the system and the expert
    - $\boldsymbol{s} \cap \boldsymbol{e}$
- D: number of disorders supported by the system (137)

Therefore, if the formula is correct, in the instance where the system suggests that every disorder is possible while the doctor suggests only one, the accuracy should be near zero. Let s = 137, e = 1 and therefore c = 1 then

$$\left(\frac{\mathbf{1}}{\mathbf{1}}\right) * \left(\frac{\mathbf{137-(137-1)}}{\mathbf{137}}\right) = \frac{\mathbf{1}}{\mathbf{137}} = \mathbf{0.007} \approx \mathbf{0}$$

Moving on, it is also useful for us to create and use a robust metric that is going to evaluate the relevance of the explanations that are generated by the system for the suspicious diseases. This will not only help to enhance the accuracy of the explanations but to also point any gaps or any information that it was intended to be included during the diagnosis.

As we mentioned earlier, the ultimate goal of the system is to provide support to both senior and junior doctors. By providing a reasonable explanation to the doctor we can assure that the system's accuracy and information that uses during its decision-making procedure does what it is supposed to do. There are plenty of disorders and there are much more symptoms and information that one must remember in order to diagnose a patient. By any means, the explanations also have the potential to be used as a point of reference and to refresh the doctor's memory with regards to a certain disorder.

6.1.4 Results

Now that we have talked about the fundamentals of the procedures, and we discussed about the metric of the system, it is finally the time to test the system and evaluate its accuracy.

A good and wise initial approach could be considered to test at least once each disorder using both the past real-life test cases and pilot clinical evaluation procedures. Although this would be an ideal scenario, it was decided for a start to pick randomly fifty (50) disorders that are going to be tested and used as reference point.

The performance of the system was assessed by comparing its decision-making ability on differential diagnosis after two rounds of questioning and one round of clinical examination findings with that of the specialist gynecologist. This retrospective cohort study was done in accordance with the Helsinki declaration and the standards of the Cypriot national committee on ethical and reliable artificial intelligence.

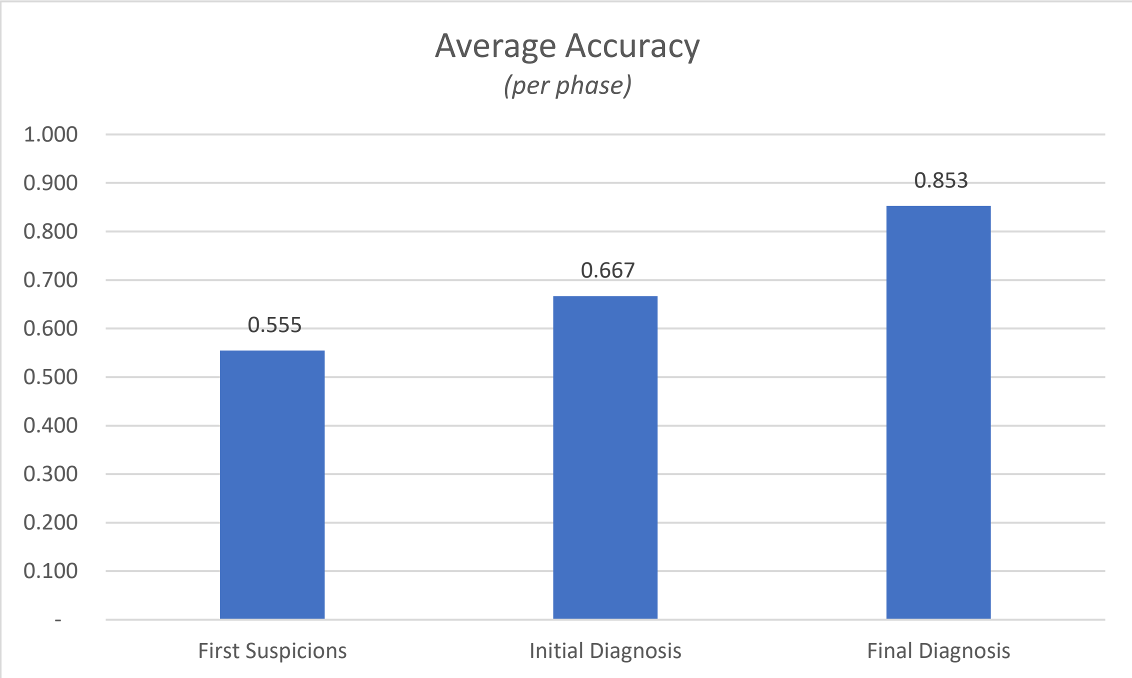

Figure 6.3. Average Accuracy

The bar chart illustrates the average accuracy (or closeness) of the system's diagnosis based on fifty randomly chosen disorders with respects to the doctor's during the three aforementioned phases, which are; the first suspicions and the initial and final diagnosis. It is immediately apparent that the further we proceed with the diagnosis the more similar results we have with the doctor.

To begin with, from an abstract perspective this increase in accuracy is the outcome of the inclusion of additional information to the system, which is mainly related with symptoms and relevant health record history. One can observe that during the first round the system diagnosis is 55% similar with that of the doctor, while the initial diagnosis is 11% more similar. However, during the final diagnosis is evident that the similarity of the doctor's and system's diagnosis has increased almost 20% after the previous round totaling at a whopping 85%. This increase is related with the clinical examination finding information, which is included in the system after the second round and are also considered by their nature very strong pieces of evidence.

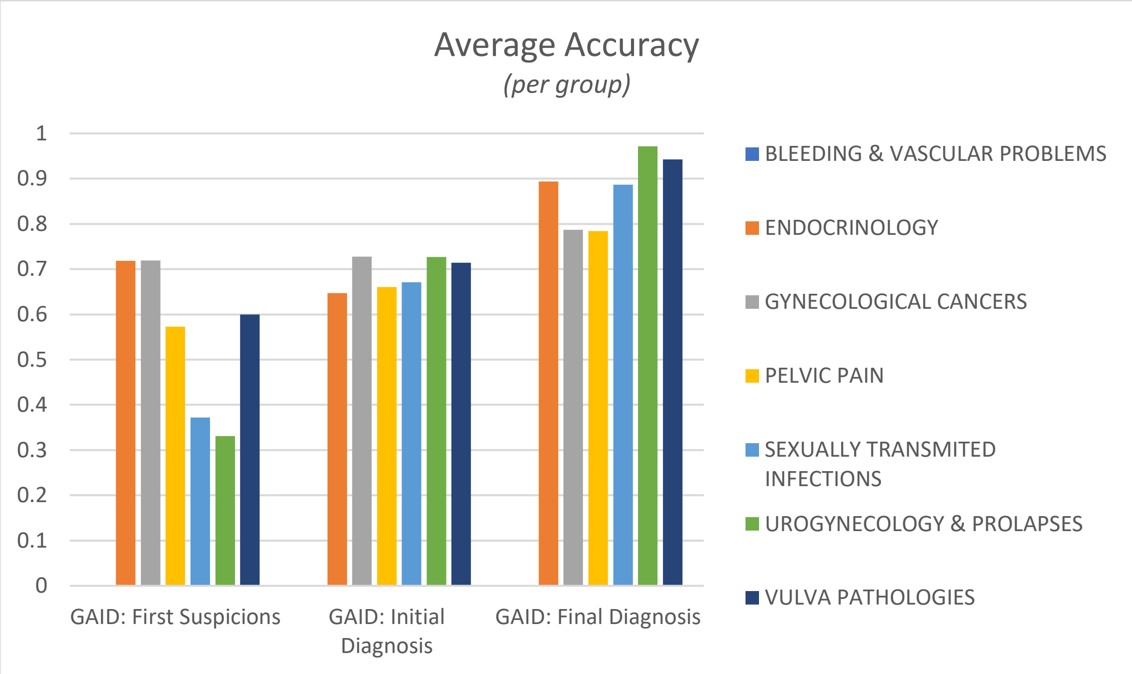

Figure 6.4. Per Group Average Accuracy

On the other hand, the bar charts in Figure 6.4. demonstrates how close the system's possible list of disorders are with regards to that of the doctor based on the relevant group of disorders that they belong. It is worth mentioning that although the group "Urogynecology and Prolapses" had the least accurate diagnoses during the round of the first suspicions it also managed to surpass any other groups by having the closest diagnoses with the doctor during the final stage. Additionally, we can also observe that every group of disorders showed an increase in accuracy at each step of the procedure except "Endocrinology" which had approximately a 5% decrease between the first suspicions and the initial diagnosis phases.

By taking a closer look to the “Endocrinology” related disorders we can observe that the sample of disorders is fairly small, which is mainly the reason behind the unexpected decrease in accuracy and it is caused by the “Adrenal Tumor” disorder. In contrast, the “Pelvic Pain” group had the largest sample of disorders with almost twenty (20).

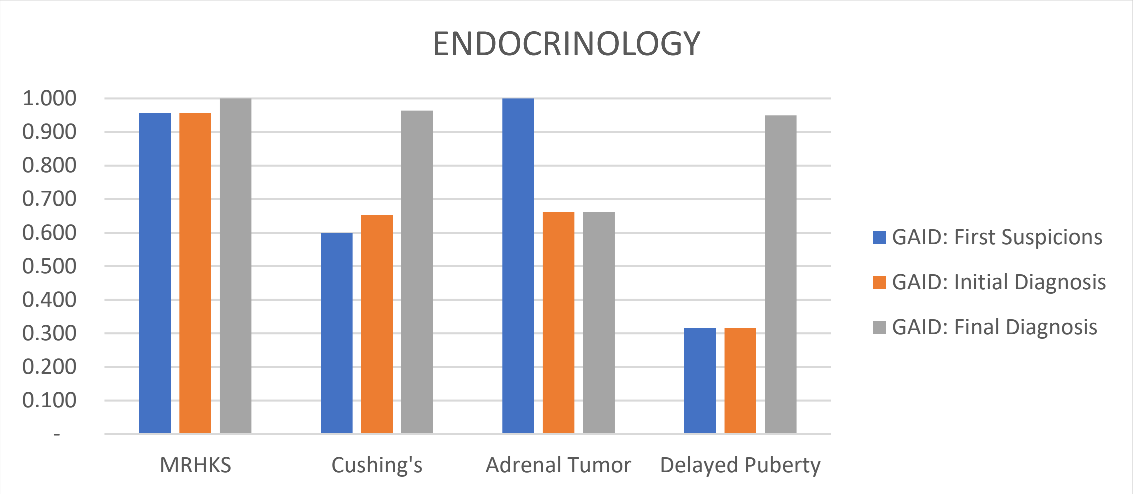

Figure 6.5. Endocrinology Accuracy

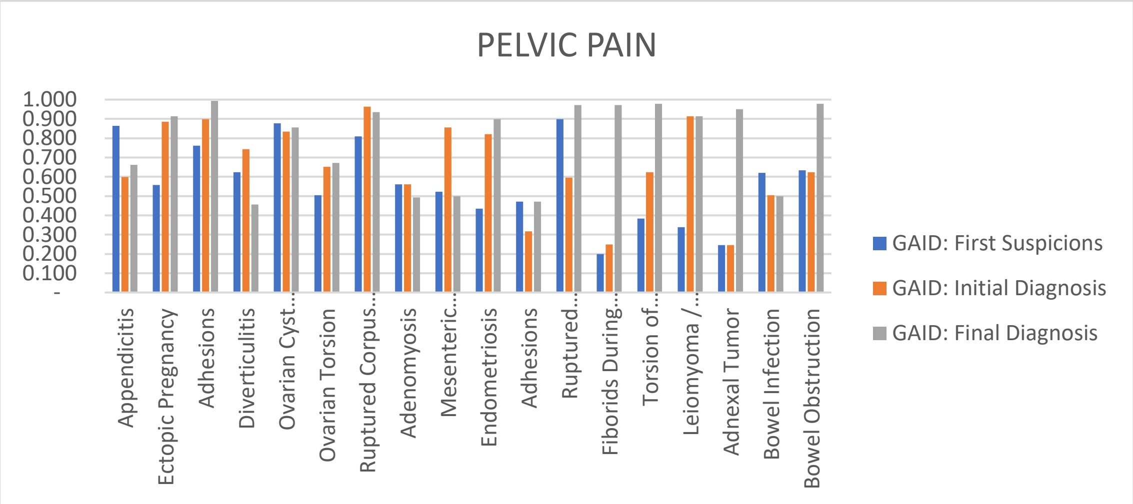

Figure 6.6. Pelvic Pain Accuracy

All in all, the system had successfully demonstrated a decent result in accuracy (how close its diagnosis is with that of the doctor) with its decision-making capabilities and also proved to be a practical database which can be used as a structured history taking assistant and friendly patient record keeping database. However, larger-scale studies are required to further evaluate the system and identify specifics preventing the achievement of a system with a fully complete knowledge base.

6.2 Knowledge Revision and Refinement

During the testing procedure, it was very important not only to score the system based on how close its diagnosis was with that of the doctor, but to also identify any missing information that will help us to enrich the system's knowledgebase and improve its decision-making capabilities.

To achieve this goal, the test case files, which were initially filled with the doctor's empirical diagnosis, are now hosting the system's diagnosis as well. The approach that we are going to use to evaluate the diagnostic capabilities of the system will be performed with a one-to-one communication with the doctor. During these meetings, the doctor will help us spot any missing information with regards to the contraindications and in some rare occasions with the diagnosis that are going to later integrated into the system. The new information was also collected in the forms of tables in which the system's final diagnosis was strictly evaluated by the doctor.

As a point of reference, we are going to randomly choose the first case file for "Appendicitis" a disorder in the "Pelvic Pain" group. Now an illustration of the procedure that was followed is going to be presented.

Final Diagnosis (APPENDICITIS)	
Appendicitis	Good
Cervical Cancer	1. Not reasonable with recent normal PAP test 2. Never been infected with HPV
PID	Good
Vaginal Cancer	1. Not reasonable with recent normal PAP test 2. Never been infected with HPV

Table 6.1. Test Case Final Table

From the above table, one can observe that four (4) disorders were included in the system's final diagnosis for "Appendicitis" two of which were considered by the doctor as irrelevant. The new contraindications are on the right column of the table and they are going to be used to enhance the knowledge base of the system.

```
%-----------------------------------------------------------------------------------------------
% SCENARIO DESCRIPTION: CERVICAL CANCER
% PREFERENCE OVER: Normal Policy
%-----------------------------------------------------------------------------------------------

%If contraindication exist
rule(laboratory_test_cr74, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    biopsy(cervix, negative),
    (D = cervicalCancer).

rule(patient_record_cr75, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    vaccine(hpv, before_first_intercourse),
    (D = cervicalCancer).

rule(patient_record_cr76, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    virgin,
    (D = cervicalCancer).

rule(patient_record_cr314, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    hasRecentPAPTest,
    (D = cervicalCancer).

rule(patient_record_cr315, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    no_history(hpv),
    (D = cervicalCancer).

rule(patient_record_cr318, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    noPainWhenApplyingPreassure,
    (D = cervicalCancer).

rule(clinical_test_cr361, prefer(not_possible_disorder(D), normal_policy_rule(D)), []) :-
    normalUltraSound,
    (D = cervicalCancer).
```

Figure 6.7. Cervical Cancer Contraindications

In Figure 6.7., in the red box, the new contraindications that derived from the test case file of “Appendicitis” were implemented and can be observed while in the orange box, there are two (2) new contraindications regarding “Cervical Cancer” that derived from other test case files. The necessary vocabulary for the new contraindications was also implemented.

Once all the contraindications were implemented with their appropriate vocabulary, the cases were executed in the system for one more time in order to collect the new diagnoses and calculate the metric once again. The new files share the same structure with the old ones. Additionally, pieces of information that have unknown vocabulary are highlighted in yellow while those that were unknown in the past but are known now are in pink. However, in pink, we can also observe the disorders that are now excluded due to the implementation of the new contraindications. Figure 6.8., 6.9. are following as a graphical representation of the information stated above.

Patient's ID: test_001

Demographics

- First Name: Appendicitis
- Last Name: 001
- Date of Birth: 12 Jan 1995
- Height: 170cm
- Weight: 75kg
- Ethnic / Origin: Caucasia (White)

Relevant Patient Record Information

- Married with 2 children
- Last delivery cesarean section
- Anxiety
- Regular periods 30 / 5
- LMP 12 days ago
- Generally healthy
  - **= No psychiatric disorder**
  - **= No cognitive difficulties**
  - **= Medicine: Never used injectable horomones**
  - **= Recent PAP Test Normal results**

Presenting Complaints

- Pelvic pain since yesterday. Started as mild pain which after few hours became stronger = **Pelvic Pain**
- Cramping = **Pelvic Cramping**
- Nauseas on / off
- No vomits
- Loss of appetite

Doctor's First Suspicions:

Ovarian Cyst Torsion	Appendicitis	Infectious Colitis	Choose an item. Ovulation	Choose an item.	Choose an item.

GAID's suspicions: [Appendicitis, Bowel Infection, Bowel Obstruction, Cervical Cancer, Crohn's Disease, Fibromyalgia, Gastrinoma, Infectious Colitis, Inflammatory Bowel Disease, Irritable Bowel Syndrome, Mild Ovarian Hyperstimulation Syndrome, Ovarian Cyst Hemorrhage, Ovarian Cyst Rupture, Ovarian Cyst Torsion, Ovarian Torsion, Para-Ovarian Cyst Rupture, Pelvic Venous Congestion, Psychological Pain, Ruptured Corpus Luteum, Ruptured Ovarian Follicles, Salpingo-ovarian Abscess, Vaginal Cancer] 22

1 | Page

Figure 6.8. Page 1 – Appendicitis Test Case

Additional Symptoms & Information

- Does not feel her ovulation
- No food poisoning
- No diarrhea, tenesmus
  - = **Normal bowel movment**
- No fever, shivering
- No pain when jumping
- Normal vaginal secretions = **Normal Vaginal Discharge Color and Smell**
- No dyspareunia
- Last bowel movement 2 days ago

Doctor's Initial diagnosis:

Adhesions	Appendicitis	Ovarian Cyst Rupture	Ovarian Cyst Torsion	Ovarian Cyst Haemorrhage	Pelvic Inflammatory Disease (PID)

GAID's initial diagnosis: [Appendicitis, Bowel Infection, Bowel Obstruction, Cervical Cancer, Crohn's Disease, Fibromyalgia, Gastrinoma, Infectious Colitis, Inflammatory Bowel Disease, Irritable Bowel Syndrome, Mild Ovarian Hyperstimulation Syndrome, Ovarian Cyst Hemorrhage, Ovarian Cyst Rupture, Ovarian Cyst Torsion, Ovarian Torsion, Para-Ovarian Cyst Rupture, Pelvic Venous Congestion, Psychological Pain, Ruptured Corpus Luteum, Ruptured Ovarian Follicles, Salpingo-ovarian Abscess, Vaginal Cancer] 22

Clinical Examination - (Examination: Findings)

- Soft abdomen
- Sensitivity and guarding in the right abdominal and pelvic side
- No lymph nodes palpated
- PV examination: Sensitivity in the central abdominal region when moving the uterus
- Uterus and ovaries normal appearance, shape, and size
  - = **Normal size Uterus**
  - = **Normal size Ovaries**
- Free fluid in pouch of Douglas 50 ml = **Ultra-Sound Scan shows collection of free fluid in Douglas Pouch**

Final diagnosis:

Appendicitis	Adhesions	Pelvic Inflammatory Disease (PID)	Choose an item.	Choose an item.	Choose an item.

GAID's final diagnosis: [Appendicitis, Cervical Cancer, PID, Vaginal Cancer] 4

2 | Page

Figure 6.9. Page 2 – Appendicitis Test Case

The overall closeness metric of the system got better once all the test cases were executed in the system and their accuracy was recalculated.

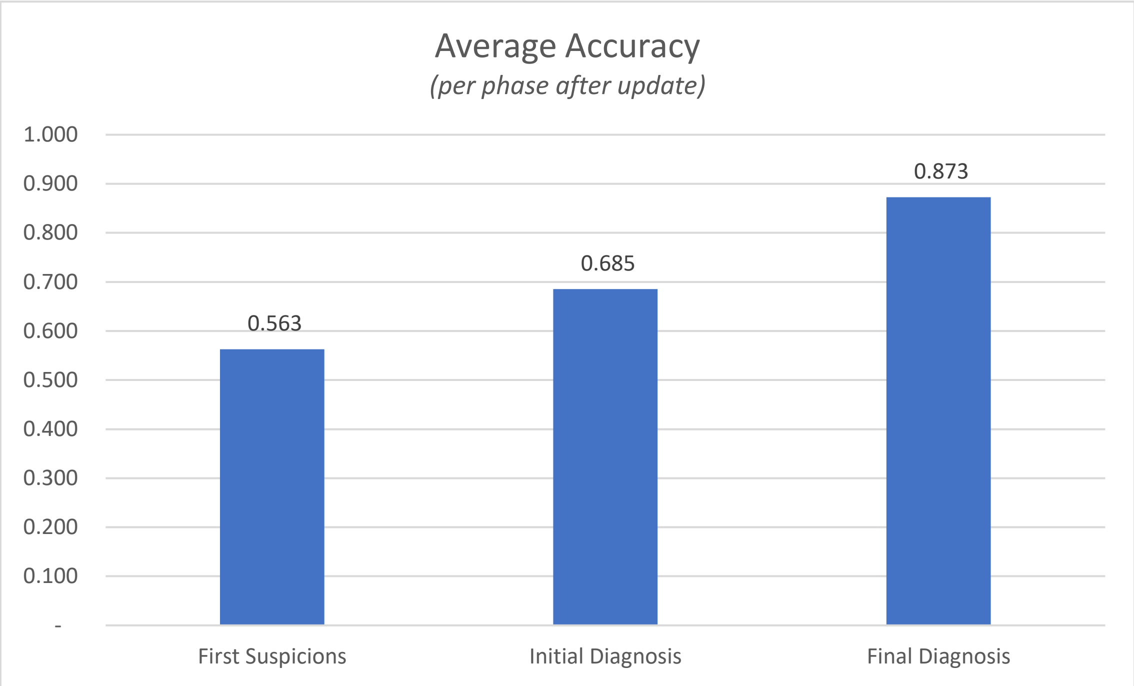

Figure 6.10. Average Accuracy after Update

Although the increase in accuracy may not be that much, we still have plenty of disorders to test and a handful of improvements to make. Overall, we manage to increase by 5% how close the system's diagnosis is with that of the doctor.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

All in all, from an abstract perspective this thesis can be considered as a detailed description that provides all the necessary guidelines and principles that systems must follow and embrace during their implementation to be able to solve real-life problems. Following an extensive bibliographic review, the resume of an in-depth approach on the diagnostic procedure with regards to gynecology is illustrated as well as the way in which one can extend an already implemented software without altering its initial purpose. The theoretical and technical chapters of the thesis can be used to solve a handful of problems that humans encounter during their life span. Additionally, the evaluation procedures and metrics can be adjusted and used as a point of reference for evaluating any kind of cognitive assistant.

However, from a more precise point of view, the outcome of the work involved during the context of this thesis was to enhance and improve the diagnostic capabilities of an Artificial Intelligent Diagnostic Decision Support System which is based on the Argumentation Framework with respects to the field of Gynecology and Obstetrics. As a result, the knowledge base of the system was also enriched with new information to cast light to any doubts concerning the lack of knowledge as far as the exclusion of disorders is considered. Now, doctors of various experiences can benefit from the use of GAIA and can utilize the system to its maximum potential while making use of contraindications and the relevant recommender engines to narrow down and exclude disorders upon demand to improve their diagnosis during a checkup.

Finally, a two-phased evaluation procedure was created with the goal of scoring the system's accuracy with regards to its diagnosis. Although it is important to know how the system perform in real-life scenarios the ultimate goal of the evaluation was to identify any gaps in the knowledge base of the system and pinpoint any bugs that will occur during the scenario execution.

7.2 Future Work

The vast majority of the functionalities that were discussed in-depth and stated in previous chapters are fully implemented into the system and thoroughly tested. However, due to the always-advancing field of gynecology and the breakthroughs that will occur as technology progresses, the demand for keeping the knowledge base of the system up to date arises.

With that in mind, the first aspect that developers must follow once they are done with the creation of such systems that are relevant with the field of medicine is to keep track of new methods of diagnosing a disorder. As recent events showed, new diseases may arise as well, therefore, the system must be updated to support any kind of newly introduced disorders and diagnosing procedures.

Moreover, we can also try and implement a module to translate laboratory test results to high-level concepts in order to provide higher accuracy and efficiency. In this module we can also integrate a system that can automatically order laboratory tests for a patient if the doctor consider it as a necessary examination that must be performed before concluding to a final diagnosis.

Additionally, a very important aspect that will be very useful in a system like this, is the ability to provide treatments and medications that may need to follow up after a successful diagnosis. So far, we discussed about the diagnosis and contraindication argumentation problems but in future version of the system, treatments and procedures that need to follow after the diagnosis must be suggested to the doctor using the same rational that was used for the implementation of the aforementioned.

Furthermore, due to the lack of time and the limitations to submit the thesis during the dissertation deadlines only fifty (50) of the disorders supported by the system have been tested with the past real-life testing method. However, in the future, we are not only planning to increase the amount of test cases but to also add additional metrics to it to identify any potential dark spots in the system's knowledge while the clinical pilot evaluation procedure will also be performed. For example, the doctors can perform a large scale clinical trial in which they have the potential to evaluate online the generated diagnoses of the system. Finally, any bugs that are going to be found during the diagnostic procedure will be noted down and will be fixed in a later version of the system.

To conclude, any possible limitations that may appear in the future are going to be used in our advantage as additional features not only to enhance the diagnosis and the user experience but also to enrich and try to optimize the system's knowledge base that will host any possible disorders and procedures used in the field of gynecology. However, one rabbit at a time.

Bibliography

[1] A. Kakas et al, “GORGIAS: Applying Argumentation”, Argument and Computation, vol 10, pp 55-81, 2019.

[2] A. Romero, “7 Famous AI Quotes Explained”, Towards Data Science, https://towardsdatascience.com/7-famous-ai-quotes-explained-782dda72d2c5.

[3] A. Wolfewicz, “Human-in-the-Loop”, Levity, https://levity.ai/blog/human-in-the-loop.

[4] A.T.M. Wasylewicz and A.M.J.W. Scheepers-Hoeks, "Clinical Decision Support Systems", Fundamentals of Clinical Data Science, pp 153-169, 2019.

[5] B. Srinivas et al, "Evaluation metrics for generation", INLG’2000 Proceedings of the First International Conference on Natural Language Generation, pp 1-8, 2000.

[6] B. Verheij, "Two Approaches to Dialectical Argumentation: Admissible Sets and Argumentation Stages", Proc. NAIC 96, pp 357-368, 1996.

[7] B.J. Copeland, “Alan Turing and the Beginning of AI”, Encyclopædia Britannica, https://www.britannica.com/technology/artificial-intelligence/Alan-Turing-and-the-beginning-of-AI.

[8] B.J. Copeland, “Alan Turing”, Encyclopedia Britannica, https://www.britannica.com/biography/Alan-Turing.

[9] B.J. Copeland, “What Is Artificial Intelligence?”, AlanTuring.net, http://www.alanturing.net/turing_archive/.

[10] B.W. Mamlin et al, "Clinical Decision Support Within the Regenstrief Medical Record System.", Springer New York, pp 190-214, 2007.

[11] C. Álvaro and C.A. Iglesias. "A systematic review of argumentation techniques for multi-agent systems research." Artificial Intelligence Review, vol 44, pp 509-535, 2015.

[12] D. Brennan, “What is a Gynecologist?”, WebMD, https://www.webmd.com/women/what-is-gynecologist.

[13] D. Edmonds, Would You Kill the Fat Man?: The Trolley Problem and What Your Answer Tells Us about Right and Wrong, Princeton Univ Press, Princeton, vol 1, pp 3-69, 2015.

[14] D. Lazaridis, “Explainable AI (XAI) and Interpretable Machine Learning (IML) Models”, https://ambiata.com/blog/2021-04-12-xai-part-1/.

[15] F. Dupin et al., "Argumentation update in YALLA (yet another logic language for argumentation)", International Journal of Approximate Reasoning, vol 75, pp 57-92, 2016.

[16] F. Toni and X. Fan, "On computing explanations in argumentation", Association for the Advancement of Artificial Intelligence, pp 1496-1502, 2015.

[17] IBM Cloud Education, “What Is Artificial Intelligence (AI)?”, IBM, https://www.ibm.com/cloud/learn/what-is-artificial-intelligence.

[18] IBM, “Explainable AI”, IBM, https://www.ibm.com/watson/explainable-ai.

[19] K. Atkinson et al, “Towards Artificial Argumentation”, AI Magazine, vol 38, pp. 25-36, 2017.

[20] M. Surya, “The Decade of Artificial Intelligence.”, Towards Data Science, https://towardsdatascience.com/the-decade-of-artificial-intelligence-6fcaf2fae473.

[21] P. Baroni and M. Caminada and M. Giacomin, “An Introduction to Argumentation Semantics”, The Knowledge Engineering Review, vol 26, pp 365-410, 2011.

[22] P.E. Beeler et al, “Clinical Decision Support Systems”, Swiss Medical Weekly, vol 144, 2014.

[23] R.L. Adams, “Powerful Examples of Artificial Intelligence”, Forbes, https://www.forbes.com/sites/robertadams/2017/01/10/10-powerful-examples-of-artificial-intelligence-in-use-today/?sh=35b2c14420de.

[24] R.T. Sutton et al, “An Overview of Clinical Decision Support Systems: Benefits, Risks and Strategies for Success”, NPJ Digital Medicine, vol 3, pp 1-10, 2020.

[25] Symptomate, “Medical App for Online Diagnosis”, Symptomate, https://symptomate.com/.

[26] Sypmtoma, “Medical Device Symptoma”, Symptoma, https://www.symptoma.com/en/medical-device.

[27] S. Ben, "Tutorial: Human-Centered AI: Reliable, Safe and Trustworthy.", 26th International Conference on Intelligent User Interfaces-Companion, vol 36, pp 495-504, 2021.

[28] S.H. Ather, “The History of Artificial Intelligence”, Aristoyofai, https://ahistoryofai.com/antiquity/.

[29] University of Rochester, “Gynecology - Tests and Procedures.”, University of Rochester Medical Center, https://www.urmc.rochester.edu/encyclopedia/collection.aspx?subtopicid=30935.

[30] Wooster Community Hospital, ”Difference Between OB/GYN and Gynecology” https://www.woosterhospital.org/what-is-the-difference-between-ob-gyn-and-gynecology/.