

May 2021

Individual Diploma Thesis

Ethics in Artificial Intelligence and Argumentation

Eleni Ioakeim

University of Cyprus



Department of Computer Science

May 2021

University of Cyprus
Department of Computer Science

Ethics in Artificial Intelligence and Argumentation

Eleni Ioakeim

Supervisor
Prof. Antonis Kakas

The Individual Diploma Thesis was submitted for partial fulfilment of the requirements for obtaining the degree of Computer Science of the Department of Computer Science of the University of Cyprus

Acknowledgements

I would like to express my sincere gratitude to my supervisor Prof. Antonis Kakas for providing guidance and feedback throughout this research. His invaluable advice, continuous support while I was carrying out this thesis, was of a greater importance. In addition, I would like to thank my family and close friends for their wise counsel and sympathetic ear.

Περίληψη

Με αυτήν τη διατριβή, προτείνουμε μία μεθοδολογία μέσω της οποίας αξιολογούμε κατά πόσο ένα σύστημα τεχνητής νοημοσύνης (ΑΙ) ακολουθά τους ηθικούς κανόνες της κοινωνίας μας. Αυτό το επιτυγχάνουμε μέσω της Επιχειρηματολογίας. Είναι σημαντικό να είμαστε σε θέση να ελέγξουμε εάν τα συστήματα τεχνητής νοημοσύνης είναι ηθικά ή όχι επειδή, καθώς οι επιχειρήσεις χρησιμοποιούν αυτήν την τεχνολογία όλο και περισσότερο, θέλουμε να διασφαλίσουμε ότι τα δικαιώματα των καταναλωτών και της κοινωνίας, γενικά, δεν παραβιάζονται.

Η σημασία αυτού τονίζεται επίσης από το γεγονός ότι έθνη από όλο τον κόσμο έχουν καθιερώσει κατευθυντήριες γραμμές και νομοθεσίες για να διασφαλίσουν ότι τα συστήματα τεχνητής νοημοσύνης δε χρησιμοποιούνται εις βάρος της ανθρωπότητας. Η διατριβή επικεντρώνεται στις κατευθυντήριες γραμμές που έθεσε η Ευρωπαϊκή Ένωση (ΕΕ).

Το γεγονός ότι θέλουμε να διασφαλίσουμε τη συμμόρφωση των συστημάτων τεχνητής νοημοσύνης και ότι έχουμε αναπτύξει ένα επεξηγηματικό εργαλείο, καθιστά σαφές ότι προωθούμε μια ανθρωποκεντρική προσέγγιση για την τεχνητή νοημοσύνη. Τα συστήματα ΑΙ δημιουργούνται από ανθρώπους, για ανθρώπους. Τα συστήματα ΑΙ δεν πρέπει να προκαλούν αρνητικά αποτελέσματα στον άνθρωπο, επομένως ελέγχουμε τη συμμόρφωση. Αυτά τα συστήματα πρέπει να σχεδιάζονται με τέτοιο τρόπο ώστε οι άνθρωποι να είναι in-the-loop (HITL), on-the-loop (HOTL) και in-command (HIC).

Με αυτήν την έρευνα, συμβάλλουμε στον τομέα της ηθικής τεχνητής νοημοσύνης και εργαζόμαστε για την εξασφάλιση αξιόπιστης τεχνητής νοημοσύνης, η οποία αποτελεί ζωτική προϋπόθεση για την ασφαλή πρόοδο της τεχνητής νοημοσύνης.

Abstract

With this thesis, we provide ways to validate the Ethics of an Artificial Intelligent (AI) system through Argumentation. It is important to be able to check if AI systems are ethical or not because as businesses use this technology more and more, we want to ensure that the rights of the consumers and society, in general, are not being violated.

The importance of this is also highlighted by the fact that nations across the world have established and are working on guidelines to ensure that AI systems are being utilized the best for humans and our societies. This thesis focuses on the guidelines set by the European Union (EU).

The EU guidelines consist of 7 key requirements which we formalized into an argumentative representation and then developed a Web App where one can validate if the design of its AI system adheres to the requirements. At the end of the validation, we provide the user with the requirements that the system complies with and which it does not, along with the explanations for each of the cases. A key component of our tool is that we provide explanations to the user so, in case of non-compliance with a requirement, s/he can see where the problem is and with our explanation, we help her/him resolve it.

The fact that we want to ensure the compliance of AI systems and that we developed an Explainable tool, makes it clear that we promote a human-centric approach to AI; AI systems shall be made by humans, for humans. AI systems should not cause negative effects to humans, thus we check compliance, and AI systems should be designed in a way that humans are in-the-loop (HITL), on-the-loop (HOTL) and in-command (HIC).

With this research, we contribute to the field of Ethical AI and work towards Trustworthy AI, which is a vital requirement for the safe advancement of AI.

Contents

Chapter 1	Introduction.....	1
	1.1 Artificial Intelligence: An Overview	1
	1.2 Why We Need Ethics in Artificial Intelligence	2
	1.3 Argumentation	3
Chapter 2	Ethics in Artificial Intelligence.....	4
	2.1 Current Approaches	4
	2.1.1 International Approaches	4
	2.1.2 European Union Approach	5
	2.2 Similarities & What Is Missing	8
	2.3 Our Approach	9
Chapter 3	Argumentation.....	11
	3.1 What is Argumentation?	11
	3.2 Argumentation and Artificial Intelligence	12
	3.2.1 Why use argumentation in Artificial Intelligence?	12
	3.2.2 GORGIAS: Applying argumentation	12
Chapter 4	Our Approach: Ethical Assessment of AI Systems Design.....	17
	4.1 Methodology	17
	4.1.1 Argumentative Input File	17
	4.1.1.1 How-to Create Your Input File	18
	4.1.2 Policy-maker Questions	23
	4.1.3 Argumentative Compliance System	26
	4.1.3.1 Types of Checks	26
	4.1.3.2 Values and Requirements	27
	4.1.3.3 Processing the Inputs	28
	4.1.3.4 Output - Explanations	31

Chapter 5	System Development Cycle.....	34
	5.1 System Requirements	34
	5.2 System Architecture & Design	35
	5.2.1 HTML/CSS - Front-end	36
	5.2.2 JavaScript - Middleman	39
	5.2.3 Back-end	39
	5.2.3.1 PHP	40
	5.2.3.2 Argumentative Compliance System	40
	5.2.3.2.1 Java	41
Chapter 6	System User Guide.....	44
	6.1 Policy Assessment Guide	44
	6.2 Policymaker Questionnaire Guide	50
Chapter 7	Evaluation.....	53
	7.1 Cognitive Evaluation	53
	7.1.1 Process of Evaluation	53
	7.1.2 The Questionnaire	54
	7.1.3 Results	64
Chapter 8	Conclusions.....	66
	8.1 Summary	66
	8.2 Future Work	67
	Bibliography.....	69
	Appendix A.....	A-1
	Appendix B.....	B-1
	Appendix C.....	C-1

Chapter 1

Introduction

1.1 Artificial Intelligence: An Overview	1
1.2 Why We Need Ethics in Artificial Intelligence	2
1.3 Argumentation	3

1.1 Artificial Intelligence: An Overview

There is no universal definition of what Artificial Intelligence (AI) is and depending on the context one is talking, the definition varies. There are general definitions such as the one by John McCarthy, in 1995, that AI is “the science and engineering of making intelligent machines”.

What depends on the context is what we define as intelligent. Yet, again by John McCarthy, a general definition of to be intelligent is the ability to achieve goals in the world. Some examples of “intelligence” are

- for an Object-Recognition system to successfully identify objects
- for a Recommender system to recommend something to a user that indeed matches her/his preferences, and last but not least
- for an Autonomous Driving Vehicle, to use the transport network without causing any accidents and follow the rules and laws of using the transport network.

We won’t give a definition for what AI is nor what intelligence is, but we will take the above-mentioned definitions as our stepping block, to elaborate on why Ethics matter in AI. As the field of AI advances, we need to make sure we, humans, gain the most out of it by enforcing a human-centric approach to it; we have to talk about Ethics, *in* and *for* AI. We will discuss this, briefly, in the next subsection and in the last subsection of this chapter, we give an overview of how we enforce Ethics, *in* and *for* AI, through Argumentation.

1.2 Why We Need Ethics in Artificial Intelligence

One might wonder what Ethics have to do with such a field but in this section we will explain exactly why AI and Ethics are so important and vital for the safe advancement of AI. We will highlight the importance by going through some scenarios of designing and deploying AI systems.

Before we continue with our scenarios, it is important to note that to establish Ethics in Artificial Intelligence reflects a fundamental part of how we humans live; we have and follow values that act as pillars for our life. If we want to have Artificial Intelligent agents interacting with humans, then we have to incorporate ethics as for both parties to contribute to what is best for human beings. However, there are various interpretations for *what is* best for human beings but as what Aristotle stated, to do the best for human beings is to do something that what will offer the most eu zên to humans [1]. Thus, our Artificial Intelligent systems/agents should promote a well living to the ones that interact with it, in our case humans.

To begin with our scenarios, we draw attention to the design phase of an AI system in which we define the world/environment and goal(s) of our system. Let's say that one of your goals favours a specific group of people. If you define this goal for your system, then it is expected that it will as well favour that specific group of people. Imagine you are a user of your AI system, and you realize this discrimination, would you be happy about it? Probably not, and thus we have to enforce the practice of *designing* ethical AI designs for AI systems.

Now, let's see the case when your design phase is completely ethical, and you don't have any unethical goals in your system but your system after being deployed and used in the world, its decisions are unfair, discriminating or promoting unethical acts. In such cases, there has to be a mechanism where your system can be frequently assessed if it promotes such negative actions or not, and in case it does, the assessment tool should provide explanations as to help you eliminate these actions.

In essence, we have to establish ethical values and assessment of ethical compliance, at all development phases of AI systems; design, development, deployment and

post-deployment. This will contribute to creating human-centric AI systems, that are safe for the public, and they won't harm humans in any way.

In the next, and last, subsection, we explain how we use Argumentation to ensure that the design phase of AI systems is ethical and that it can be safely developed into a system.

1.3 Argumentation

We define argumentation as “the action or process of reasoning systematically in support of an idea, action, or theory” and we utilize this act to check whereas a design of an AI system supports some specific ethical values.

Via an argumentation framework, which we will describe in detail in a later on chapter, we first support that a given design policy is not supporting any of the specified ethical values. Thus, we set an argumentation between our tool and a given policy.

The policy, which has to be in a format specified by us, describes the environment and goals of our to-be AI system. Our tool, based on the environment values, which is the set of parameters that our system will have and their values, it initiates an argumentation with the policy, and it tests it against various of its parameter values. By testing it, it checks whether a goal, that results by some parameter values, promotes an unethical action.

The policy wins the argument in case no unethical goals are detected. In case an unethical goal is found, the policy loses the argument and our tool will let the user know what parameter values caused the unethical goal, prompting the user to revise its policy and come back later to check it again.

In [Chapter 4](#), we explain, exhaustively, our methodology and state exactly what checks our tool performs, what unethical actions we look out for and what happens if we find that an unethical action is promoted.

Chapter 2

Ethics in Artificial Intelligence

2.1 Current Approaches	4
2.1.1 International Approaches	4
2.1.2 European Union Approach	5
2.2 Similarities & What Is Missing	8
2.3 Our Approach	9

2.1 Current Approaches

There is a common effort among companies, organizations and nations, across the globe, to set guidelines and standards to enforce Ethics in AI. It is universally acknowledged that Ethics are important and that we should work on incorporating them in the development of AI. Currently, there are various approaches, which we will mention in the next subsection but for the sake of this thesis, we focused on the guidelines provided by the European Union (EU).

2.1.1 International Approaches

There are companies, like IBM [2] and Microsoft [3], that have established departments specifically for researching and contributing to Ethical AI. They promote Ethics in AI by infusing in their own products, ethical designs and approaches for their AI systems. Also, other than setting in-house guidelines, they also provide open-source toolkits and guides for anyone to use, as to ensure we *all* work towards Ethical AI.

Other than companies, there are individual nations that established their own guidelines. Singapore, in 2018, established its advisory council on the Ethical Use of AI and Data, consisting of members from diverse backgrounds [4]. The council is designated to ensure the responsible development and deployment of AI in Singapore via developing and publishing guidelines, codes of conduct and overall promoting the ethics standard for a safe advancement of AI within Singapore.

Another, European, country that has been very active regarding the matter is Denmark. In March 2019, the Danish government introduced to the public its national strategy for artificial intelligence [5] and then by July 2020, it was the first country to enforce a mandatory legalization for AI and Data Ethics [6].

Last but not least, standard associations have been also working on forming standards to motivate technologists to use more human-centric approaches when developing and deploying their AI systems. IEEE's 7000™ series of standards [7], is dedicated exactly to this motivation; there are projects regarding ethics in the design of AI systems to projects about ethics on the data used to train AI systems and how companies should handle their data.

The aforementioned approaches, by some means they all promote these, and not only, core values of Ethical AI: develop explainable, transparent and fair AI systems, where humans are able to intervene at any phase the AI system might be at. In the next subsection, we will focus on the approach set by the European Union and the latest regulation proposal.

2.1.2 European Union Approach

The European Union (EU) in turn formed its own guidelines for Ethical AI systems. In 2018, the EU formed a High-Level Expert Group (HLEG) on Artificial Intelligence where members are experts from academia, civil society and industry. This group was assigned to provide insights on the EU's AI strategy. By July 2020, the group published the final deliverable regarding Trustworthy AI [8], which our tool was build upon.

The deliverable is an Assessment List for Trustworthy Artificial Intelligence (ALTAI) for self-assessment. This assessment list, brings in question 7 key requirements for Ethical AI:

- Human Agency and Oversight;
- Technical Robustness and Safety;
- Privacy and Data Governance;
- Transparency;
- Diversity, Non-discrimination and Fairness;

- Societal and Environmental Well-being;
- Accountability.

For each requirement, they explain why it is important and provide some questions for one to answer as to check if their system adheres to the specified requirement or not (thus it is for self-assessment).

In short, here is what each requirement supports:

- Human Agency and Oversight: AI systems should *support* human agency and human decision-making, and never violate these fundamental values. Mentioning the practice of Human-in-Command (HIC), Human-in-the-Loop (HITL) and Human-on-the-Loop (HOTL).
- Technical Robustness and Safety: AI systems have to be secure and safe to use at any time. They should be resilient to (cyber)attacks and never be dangerous to use e.g. cause harm to a human due to not proper training (for Machine Learning AI systems). Threats and weaknesses of the system have to be identified and known. In case of a failure, there has to be a fall-back plan as to prevent unwanted and possibly dangerous side effects e.g. failure of the AI system of an autonomous car.
- Privacy and Data Governance: The data used to create an AI system and data used by an AI system, should never violate the privacy of the users. This section mentions best-practices that anyone developing AI systems should considerate and apply, such as, and not limited to: data anonymization, encryption and compliance with data protection regulations.
- Transparency: It should be clear what data have led to a decision of the system (traceability) and the AI system owners should be able to explain technical processes and reasoning behind a decision of the system (explainability). It is crucial for users to be able to understand how the systems has come to a decision.
- Diversity, Non-discrimination and Fairness: Throughout the entire life cycle of an AI system, it should be ensured that the action taken are not discriminating and

unfair. The design should be as such, so anyone will be able to use the system, especially regarding people with disabilities and the decisions of the system, should not promote an unfair and discriminating stance e.g. provide different decisions due to different gender, nationality etc. To eliminate such consequences, it is advised that all people involved in the creation of the system, follow an ethical approach when taking their actions.

- **Societal and Environmental Well-being:** As mentioned before, AI systems should enhance and support humans as individuals and as a society furthermore. Thus, we must ensure that the AI systems out there do not promote unethical actions, suggest decisions that harm the democracy or any decisions that go against our society or prompt actions that harm the environment. Furthermore, the development and deployment of AI systems should be as environmental friendly as possible e.g. not requiring a lot of (energy) resources to develop and deploy.
- **Accountability:** This last requirement, addresses the importance of having a clear image for who is accountable for what regarding an AI system e.g. have track who and what was used to design it, and also wants to ensure that the owners of AI systems, work for continuously ensuring their AI systems are ethically aligned and that they comply with the regulations. This requirement depends a lot on the risk management and auditability processes followed by the owners of AI systems.

If one goes through the assessment list, it is clear that the HLEG on AI emphasizes a lot that when designing an AI system, the opinion, and agency of all stakeholders is important. We also believe that this is a crucial requirement because there are out there AI systems that might favour the company deploying it, but its functionality might be unfair towards the users of the system. A good example is when Air Berlin went bankrupt, Lufthansa's automated pricing algorithm increased prices up to 30%, right away after the bankruptcy of their competitor [9]. We can see that the system was designed to offer maximum profit to the company even if that would result into an unfair act towards some stakeholders e.g. the users of the system.

Finally, it is important to mention that the EU, in April 2021, proposed a regulation on Artificial Intelligence; the Artificial Intelligence Act [10]. For the scope of this thesis, we won't elaborate more about it other than this humble mention.

2.2 Similarities & What Is Missing

Disregarding the number of approaches, all of them focus on the same values. We see numerous, individual, approaches to Ethical AI which all structure their approach around the same values and requirements e.g. Transparency, Accountability, Fairness, Non-Discrimination etc.

All nations want to be the leading power regarding AI, but if one considers the impact AI has nowadays in our lives, maybe it is needed for all entities working towards Ethical AI, to team up and form a universal law. For example, a company might develop AI systems in country X, follow the guidelines of the specified country, but the system will be used by people from other countries. Then, the AI system has to comply with the Ethical AI guidelines from all the countries that their system can be used from. Thus, we believe there has to be a common, unified approach to Ethical AI.

Furthermore, there is a lot of emphasize on the philosophical aspect on Ethical AI and the technical aspect is not as concrete as it should. Admittedly, there are tools and specific ethical design patterns out there for one to use, but we believe there is still a gap between *saying* we have to follow an ethical approach to AI and *implementing* an ethical approach to AI.

It is anticipated that this gap will be reduced now that regulations, not just guidelines, are being enforced. Ideally, there could be a tool whereas anyone can test its AI system against and check if it adheres to all the core Ethical AI values. There are such tools, for example the frameworks companies like IBM and Microsoft provide to their customers, and there is a web tool of ALTAI, in a questionnaire format, that companies can use to assess their AI systems. However, there isn't any tool that can be universally used to automatically assess the design of AI systems and the end AI system itself. In the next subsection, we elaborate on how we attempted to create such a tool, elaborating on our methodology and how we then implemented it in a web application.

2.3 Our Approach

After studying the EU guidelines, we set a goal to develop a tool where one can test if a policy, which a future AI system will be based upon, adheres to some specific values. We chose to work with the *design* of the systems rather than the end system, because we support the idea that the design phase of an AI system's life cycle is critical and predisposes unethical AI systems if the design is itself unethical. For more insight why the design phase is the fundamental for an AI system, we suggest reading the publication by Theodorou A., Why Artificial Intelligence is a matter of Design [11].

Due to the fact that we chose to focus on the policy, we had to eliminate some requirements, as they don't apply for a policy as they do for an AI system. For example, the Technical Robustness and Safety requirement, does not make sense in the context of a policy. The values which we worked with are:

- Privacy and Data Governance;
- Transparency;
- Diversity, Non-discrimination and Fairness;
- Societal and Environmental Well-being;
- Accountability.

In the methodology chapter ([Chapter 4, Section 4.1.3.2](#)), we describe in detail how each requirement fits the policy context and state when a policy adheres or not to the requirement. It is important to state that the interpretation we gave to the ethical values and requirements, are just our approach to the issue. The interpretation could change between contexts e.g. a policy and an AI system, and also between stakeholders e.g. what is a fair to a company might be different to what customers believe it is fair. For this reason, it is important that all stakeholders have a say into the design phase of a system, to ensure that all views are taken in consideration.

After determining which values we will focus with, we had to find a way we can actually check adherence. Our tool, is an implementation of what was described, by Aler Tubella, A. et al. , in Governance by Glass-Box [12]. Our tool acts as a glass-box around an argumentative representation of a policy, and we evaluate the various outcomes of the policy to check if there are results which violate the values.

In the next subsection, we explain what we mean by “argumentative representation” and later on, in [Chapter 4](#), we get in more detail on how the tool works.

Chapter 3

Argumentation

3.1 What is Argumentation?	11
3.2 Argumentation and Artificial Intelligence	12
3.2.1 Why use argumentation in Artificial Intelligence?	12
3.2.2 GORGIAS: Applying argumentation	12

3.1 What is Argumentation?

When we search “What is Argumentation?”, and follow the resource from Wikipedia [13], it states the following:

“Argumentation theory, or argumentation, is the interdisciplinary study of how conclusions can be reached through logical reasoning; that is, claims based, soundly or not, on premises. It includes the arts and sciences of civil debate, dialogue, conversation, and persuasion.”

The main takeaway for us from this definition is that Argumentation is a form of dialogue in which, through logical reasoning, we reach some conclusions. Logical reasoning happens through making claims based on premises, which are propositions upon which an argument is based or from which a conclusion is drawn.

Furthermore, Argumentation is a fundamental part of how, we, humans reason. Based on our knowledge, beliefs etc, we form arguments and use them to defend or support ourselves or someone else. It’s a form of *intelligence*. Lastly, it’s important to state that the arguments we form, have a “weight” to them. Some arguments are more strong regarding one conclusion and others are stronger for a different one. At any given time of an argumentation, usually, we choose the argument which favours our goal conclusion.

3.2 Argumentation and Artificial Intelligence

3.2.1 Why use argumentation in Artificial Intelligence?

Artificial Intelligence is approached from 4 main point of views: Thinking Humanly, Thinking Rationally, Acting Humanly and Acting Rationally. The Thinking Rationally perspective, talks about how we can create systems from a logicist, where they demonstrate a *logical reasoning* behaviour. We define that logical reasoning, is when one given some premises, it can apply deduction and result into a conclusion.

Argumentation, is a form of logical reasoning; arguments are being stated (the premises) and then through a deduction process we try to win the argumentation by reaching our favoured conclusion. If we are able to apply this form of local reasoning to a machine, then we are describing the “Thinking Rationally” approach, and this approach is important as it captures how humans, usually, think. Artificial Intelligence is not just about identifying objects or being able to classify things. Reasoning is intelligence.

Another important reason why Argumentation is useful in AI, it’s because of its explainable nature. When we conclude after an argumentation, we can trace back every argument that led to the specific conclusion. This is important because we want AI to become more and more Explainable, and by having Argumentative (or any other Logic based) AI, it promotes this requirement by design.

In the next section, we explain how we can implement argumentation on a machine, using an argumentative framework. We used this argumentation framework to implement our tool and in [Chapter 4](#), we explain in detail how we applied it in our case.

3.2.2 GORGIAS: Applying argumentation

The GORGIAS: Applying argumentation framework [14] was proposed by A.C. Kakas et al., in 2018. They capture argumentation in the sense of *preference-based arguments* and this tool was designed as for anyone to be able to use it, due to its very intuitive nature. We will elaborate on how one can use it, as an introduction for later on where we explain how we used it. If you want to learn more about the framework per se, the publication is publicly available.

As mentioned, the tool is based on arguments with preference. For example, if we are asked if we like ice cream, and we say that we like ice cream during the summer only, then we have a *preference* over when do we like to have ice cream. Specifically, we just created a *scenario* -the scenario where we like ice cream “We like ice cream during the summer only”. We can create several scenarios and then set preferences between them, in case they are conflicting. This is the main idea behind GORGIAS; there is a structure called “Scenario-Based Preferences” with which the argumentation happens between scenarios based on the preference the user defines.

We will now explain in more detail what a scenario-based preference is and how we can use them to create an argumentation application. Our explanation will be based on formalizing the following natural language text into GORGIAS code.

“In general, I like to drink coffee. In the morning, I like to drink a cup of cappuccino. After I have my lunch, I like to have a shot of espresso and at night I do not drink coffee.”

The above text is an example “policy” for when someone likes to drink coffee which we will formalize into Scenario-Based Preferences (SBPs). A Scenario-Based Preference is a tuple of 2 sets: S and O

$$SBP = \langle S; O \rangle$$

where the set S, describes our scenario and the set O, contains which options are enabled in the specified scenario.

From our coffee policy, there are 4 scenarios:

1. The general case where the use might drink any coffee.
2. In the morning, s/he drinks cappuccino.
3. After lunch, s/he drinks espresso.
4. At night s/he does not drink coffee.

The options enabled at each scenario, are the coffee types s/he might drink. For the general case, all mentioned coffee types are options, for the specific cases the only option is the specified coffee type and for scenario 4, there is no option enabled.

To represent our scenarios and options in the tuple notation, we have to give them a descriptive alias. For each scenario and its options, here is a possible alias:

1. For the general case, the scenario set is an empty set $\{\}$ because it's not a specific scenario and the options are $\{cappuccino, espresso, french\}$ because s/he likes to drink coffee, thus any of the mentioned coffees are an option.
2. In the morning, the scenario set will be $\{morning\}$ and the option set $\{cappuccino\}$.
3. After lunch, the scenario set will be $\{after_lunch\}$ and the option set $\{espresso\}$.
4. At night, the scenario set will be $\{night\}$ and the option set is an empty set $\{\}$ because s/he doesn't drink coffee.

From the above analysis, we get the following SBPs

1. $SBP_1 = \langle \{\}; \{cappuccino, espresso, french\} \rangle$
2. $SBP_2 = \langle \{morning\}; \{cappuccino\} \rangle$
3. $SBP_3 = \langle \{after_lunch\}; \{espresso\} \rangle$
4. $SBP_4 = \langle \{night\}; \{\} \rangle$

Notice that the options of SBPs 2-4 are a *refinement* of the options of the general case. We refine what the options are for specific scenarios. And note that the scenario set is getting more populated which makes our scenarios more specific.

Now, we have to formalize the SBPs structures into GORGIAS code. GORGIAS has a structure called a *rule*; a rule is used to define the overall options available, define our SBPs and define preferences between our SBPs.

To begin with, we have to define the goals of your policy. For our coffee policy, the goal is to decide which coffee the user will drink. Thus, the first rules will be:

```
rule(r1(cappuccino),drink(cappuccino),[]).
rule(r2(espresso),drink(espresso),[]).
rule(r3(french),drink(french),[]).
rule(r4(none),drink(none),[]).
```

Each rule has the structure $rule(r_i(option), drink(option), [])$, which define the coffee the user wants to drink. We will use these basic rules as our building block for formalizing our SBPs.

For the universal case, the user likes to drink coffee thus we *prefer* the rules 1-3 over the 4th rule. This can be formalized as such:

```
rule(pr1(cappuccino),prefer(r1(cappuccino),r4(none)),[]).
rule(pr2(espresso),prefer(r2(espresso),r4(none)),[]).
rule(pr3(french),prefer(r3(french),r4(none)),[]).
```

Now, the rules have the structure

$rule(pr_i(option), prefer(r_i(option), r_k(option)), [scenario_set])$

We use this structure to define preferences between our base rules. With the rules pr1 to pr3, we define that in general we prefer to drink one of the 3 coffees over not drinking any coffee at all. The scenario set in the universal case is empty, thus there is nothing between the brackets.

For the second scenario, which describes the morning time, we have a scenario value and the preference rules are:

```
rule(pr4(cappuccino),prefer(r1(cappuccino),r2(espresso)),[morning]).
rule(pr5(cappuccino),prefer(r1(cappuccino),r3(french)),[morning]).
rule(pr6(cappuccino),prefer(r1(cappuccino),r4(none)),[morning]).
```

We can see that now we have *morning* in the brackets because that's how we define our scenario. The after lunch scenario is implemented similarly, and we will jump to the 2 night scenarios, where we will have to set a preference between other preference rules.

For the first night scenario, we describe that at night the user does not drink coffee, thus following the same procedure as before we get the following rules:

```
rule(pr10(none),prefer(r4(none),r1(cappuccino)),[night]).
rule(pr11(none),prefer(r4(none),r2(espresso)),[night]).
rule(pr12(none),prefer(r4(none),r3(french)),[night]).
```

However, these rules are not enough to define our preference. Recall from the universal case, we have defined that we prefer all the other options over the *none* option. Now, with the rules pr10 - pr12, we are stating the opposite. We have conflicting arguments!

When we have conflicting arguments, as we did between our basic rules, we set preference among our current scenarios. We need to set a preference to not drink coffee when it's night, even though we universally stated that we drink coffee. The rules to describe this follow:

```
rule(c1(none),prefer(pr10(none),pr1(cappuccino)),[]).  
rule(c2(none),prefer(pr11(none),pr2(espresso)),[]).  
rule(c3(none),prefer(pr12(none),pr3(french)),[]).
```

If you look closely, now in the *prefer* part of the rule, we have other preference rules instead of our basic rules. With these higher level preference rules, we defend the pr1 to pr3 preference rules which attack the *none* option. And this concludes the formalization of our problem to decide which coffee we shall drink. It is noteworthy that the scenario options in the brackets, are called facts, and we show how you can define them in the full GORGIAS code which can be found in Appendix A.

Concluding, we demonstrated how using the GORGIAS argumentative framework we can define an argumentation arena, where the arguments are the rules and the preferences between the rules, are either the way to set a preference or a way to defence an argument. Argumentation doesn't apply only to political discussions, but we can apply it to various decision-making problems; our coffee policy could represent the system being run on a tailored coffee machine.

Chapter 4

Our Approach: Ethical Assessment of AI Systems Design

4.1 Methodology	17
4.1.1 Argumentative Input File	17
4.1.1.1 How-to Create Your Input File	18
4.1.2 Policy-maker Questions	23
4.1.3 Argumentative Compliance System	26
4.1.3.1 Types of Checks	26
4.1.3.2 Values and Requirements	27
4.1.3.3 Processing the Inputs	29
4.1.3.4 Output - Explanations	31

4.1 Methodology

In this chapter, we will go through the methodology that we developed our tool upon. From a top-down approach, in the first subsection, we state the format an input file of our tool should have, and we explain how one can create this file based on a policy in natural language. Along with the file, the policy-maker has to answer some questions to provide us with more needed information. Then, in the third subsection, we explain how our system processes the input file and how we use the post-processing results along with the answers from the policy-maker to create an argumentation file. This file is then used to check compliance and non-compliance. This check results in argumentative explanations of whether the given input is compliant or not. Finally, in the last subsection, we describe how we transform the argumentative explanations into natural language explanations.

4.1.1 Argumentative Input File

To be able to use our tool, the user has to formalize their policy into an argumentation based representation. This is the file that the system will use to make the necessary assessments. In the next section, based on an example policy, we demonstrate how the

user can formalize their policy. We advise the reader to read the Section 3.2.2 GORGIAS: Applying argumentation first before continuing, if you haven't already done so.

4.1.1.1 How-to Create Your Input File

For demonstration purposes, we assume that we are an insurance company that offers a car insurance plan at various price points, depending on the profile of the customer. Here follows the natural language format of our policy:

“The basic third-party car insurance plan costs 180 euros per year. If the driver is younger than 24 years old or older than 69 years old, then there is a 20% increase. Also, there is a 5% increase when the driver has been licensed for less than 2 years. Finally, if the customer lives in a high crime area, there is a 10% increase.”

Now that we have the natural language format of our policy, we will begin the formalization process. By the end of the formalization, we will end up with an argumentative GORGIAS PROLOG file. We will break this process into 3 steps.

Step 1: Identify the option predicate

Each policy has an outcome. In our case, the outcome is the price to offer an insurance plan. This outcome has several possible values as we have various price points. To represent this relation, we will create an option predicate called *premium_cost* where this predicate can take the values low, medium, high. Thus, we have the following option predicates

premium_cost(low), premium_cost(medium), premium_cost(high).

We set that the low option holds when the customer is eligible for the basic plan, medium when the customer has up to 10% increase and high when the customer has more than 10% increase.

Step 2: Construct Scenario-Based Preferences (SBPs)

After defining our option predicate, we translate the conditions of our policy into Scenario-Based Preferences. Recall from Section 3.2.2, that a Scenario-Based Preference is a tuple of sets, S and O. The set S, is a set of states in the current environment and the set O, is a set of options that are enabled for the specified S. The states in S describe a scenario and the options in O are available in the given scenario.

Each condition of our policy, will be a scenario and the price point at the specific condition will be the option. Also, most policies have a default scenario, in our case is the basic third-party plan which costs 180 euros. Our first SBP, will be the default scenario.

$$SBP_0 = \langle S_0 = \{basic_third_party_plan\}, O_0 = \{premium_cost(low)\} \rangle$$

Now, the rest SBPs will be a refinement of the default SBP. We will refine the scenario set by adjusting it to the more specific conditions and changing the value of the option predicate accordingly. The first refinement we get from the policy, concerns the age of the driver.

“If the driver is younger than 24 years old or older than 69 years old, then there is a 20% increase.”

$$SBP_1 = \langle S_1 = \{basic_third_party_plan, driver_younger_than_24\}, \\ O_1 = \{premium_cost(high)\} \rangle$$

$$SBP_2 = \langle S_2 = \{basic_third_party_plan, driver_older_than_69\}, \\ O_2 = \{premium_cost(high)\} \rangle$$

The next condition is regarding the number of year that the driver has his/her driving licence.

“Also, there is a 5% increase when the driver has been licensed for less than 2 years.”

$$SBP_3 = \langle S_3 = \{basic_third_party_plan, licenced_less_than_2\}, \\ O_3 = \{premium_cost(medium)\} \rangle$$

Lastly, we have the condition regarding the residence are of the customer.

“Finally, if the customer lives in a high crime area, there is a 10% increase.”

$$SBP_4 = \langle S_4 = \{\text{basic_third_party_plan, high_crime_area}\}, \\ O_4 = \{\text{premium_cost(medium)}\} \rangle$$

These were the 5 conditions we could extract directly from the policy in the natural language format. However, we can see that the scenarios can be combined. For example, a customer might be 21 years old and living in a high crime area. The SBP of this combination would be:

$$SBP_5 = \langle S_5 = \{\text{basic_third_party_plan, high_crime_area, driver_younger_than_24}\}, \\ O_5 = \{\text{premium_cost(high)}\} \rangle$$

It's up to the user to define as many SBPs are required to describe their policy thoroughly.

Step 3: Construct argumentative file

Now that we have our option predicate and the SBPs, we will put together our argumentative file. Throughout this process, we will see how we prioritize our SBPs and set preferences among them.

Before we begin, we need to state the format of a basic rule and a preference rule and a priority rule. Our file will be consisting of some basic rules and some preference rules. The basic rules will describe our option predicates, and they will have the following format:

$$rule(ri(), option_predicate(value), []).$$

Where i is an integer denoting the number of the rule, `option_predicate` is the name of our `option_predicate` and `value` is a value of our `option_predicate`. We will create N such rules, where N is the number of option predicate values we have.

The preference rules will describe which basic rule we prefer over the other ones based on our scenario. They follow this format:

$$rule(pri(),prefer(rk(),rj()),[scenario_values]).$$

Where i is an integer denoting the number of the preference, k is an integer denoting the rule number k , j is an integer denoting the rule number j and in the last square brackets, we can put our scenario values if any (a value that describes a scenario). With this preference rule we

describe that we prefer the rule k over rule j in the specified scenario. We will create as many preference rules as needed to describe our policy.

Having covered the notations, we will proceed with creating the basic and preferences rules for our example policy. We will name our file “thirdPartyCarInsurancePolicy.pl”. The extension is `.pl`, as PROLOG is our programming language. We will show a part of the process here and the full PROLOG file can be found in Appendix B.

This is a 2-step process.

- Step I: Define basic rules

For our policy, the option predicate is `premium_cost`, and we have 3 values: low, medium, high. Thus, we will create 3 basic rules as such:

$$\begin{aligned} &rule(r1(),premium_cost(low),[]). \\ &rule(r2(),premium_cost(medium),[]). \\ &rule(r3(),premium_cost(high),[]). \end{aligned}$$

These rules will be the first rules in our file.

- Step II: Define preferences rules

After setting the basic rules, now we will set the preferences rules that will define which of the basic rules will hold at a specified scenario. The preferences rules will be extracted from the SBPs we created in the previous section.

The first SBP to transform will be our default scenario, SBP_0

$$SBP_0 = \langle S_0 = \{basic_third_party_plan\}, O_0 = \{premium_cost(low)\} \rangle$$

Out of all rules, we prefer the *premium_cost(low)* when the scenario is just requesting a basic third party plan (*basic_third_party_plan*).

To express the above into preferences rules, we have to set the following rules:

$$\begin{aligned} &rule(pr1(), prefer(r1(), r2()), [basic_third_party_plan]). \\ &rule(pr2(), prefer(r1(), r3()), [basic_third_party_plan]). \end{aligned}$$

We have now defined that we prefer rule 1 over rule 2 and that we prefer rule 1 over rule 2 when at the default scenario of our policy.

If your policy has N basic rules, then you need to create N-1 preference rules in which you will set that you prefer the rule K over the other rules, which rule K is the basic rule for the option predicate value that's the default option of your policy.

After transforming the default SBP_0 , we continue with the rest.

$$\begin{aligned} SBP_1 = \langle S_1 = \{basic_third_party_plan, driver_younger_than_24\}, \\ O_1 = \{premium_cost(high)\} \rangle \end{aligned}$$

We create a preference rule as such:

rule(pr3(),prefer(r3(),r1()),[basic_third_party_plan, driver_younger_than_24]).

And we have to add another preference rule, at a higher level, like that:

rule(c1(),prefer(pr3(),pr2()),[]).

If we have the scenario *[basic_third_party_plan, driver_younger_than_24]* then both *pr2()* and *pr3()* hold. The rule *pr2()* holds because the scenario value *basic_third_party_plan* is enabled and *pr3()* holds because both *basic_third_party_plan* and *driver_younger_than_24* scenario values are enabled. These 2 rules attack each other when they both hold, thus we need an extra preference to explicitly define which of the 2 should hold. Thus, we add the preference rule *c1()* to show that only *pr3()* will hold when the scenario is *[basic_third_party_plan, driver_younger_than_24]*.

Recall that in argumentation, when arguments attack each other, it means that argument A supports opposite views than argument B and argument B has opposite views than argument A. In most cases, we try to find another argument as to break this dead-end. In our case was adding another preference with the *c1()* preference rule. There are cases though that it might not be possible to break such dead-ends.

The rest SBPs are transformed similarly and make sure to add extra preference rules if you can eliminate attacks. We won't transform the rest of SBPs in this section, but you can find the fully transformed PROLOG file of our policy at the Appendix B.

4.1.2 Policy-maker Questions

Other than the argumentative representation of your policy, the tool requires you to answer to some Yes or No questions regarding the development of your policy. It is recommended that these questions are answered by the policymaker.

Here is the list of the questions that you will be asked to answer with an explanation for each one of them, to help you understand what it is requested:

- If needed, the decision-making path behind a condition of the policy can be accessed.

Explanation: Answering “Yes” to this question, it means that if one asks why a condition is in your policy, then you should be able to explain. If answered otherwise, it means that there is no clear explanation why a condition had to be in your policy. Based on our car insurance policy, if we answered “Yes” it means that if someone asked us why there is the condition where a young driver has to pay more (or any other condition) we would be able to explain. A possible explanation could be that the condition has to be in the policy because young drivers tend to have more accidents.

- If needed, the data used to create the conditions of the policy can be accessed.

Explanation: Answering “Yes” to this question, it means that if one asks you to show the data that lead a condition to be in your policy, then you should be able to provide such data. If answered otherwise, it means that you do not have any data to support your condition. Continuing with our example condition where a young driver has to pay more, if we answered “Yes” to this question, it means that we would be able to provide the information that support this condition. Possible supporting information could be statistical reports that show that young people have more accidents.

- Your policy does not have any conditions that violate the consumers' privacy.

Explanation: Answering “Yes” to this question, it means that any data you acquire from a customer, this information is never used against an individual and its integrity. If answered otherwise, it means that in some way you use the information to an advantage of a customer's integrity, to achieve something in favour of your company. For example, a company could be sharing data of customers with another third company, from which your company can benefit from if you provide them the data.

- You have established a data governance process regarding your policy.
Explanation: Answering “Yes” to this question, it means that any data you acquire from a customer, can be evidently shown that are safely stored and processed. If answered otherwise, it means that the data are not governed in any way. A proof that you have established a data governance process could be showing that you comply with a data governance regulation e.g. the General Data Protection Regulation (GDPR).
- You have considered any long-term or/and short-term impact that your policy might have on the society, and you have eliminated any negative effects.
Explanation: Answering “Yes” to this question, it means that when developing your policy, you have thoroughly considered all the effects that your policy could have upon the society, and you have eliminated any conditions that would impose negative effects. If answered otherwise, it means that you have either not thought about the impact your policy could have on the society, or you are not able to control the impact your policy has. Our example condition where young drivers have to pay more for a car insurance, it could result into young drivers not insuring their cars and thus being illegal (in Cyprus it is illegal to drive a car without a car insurance).
- You have considered any long-term or/and short-term impact that your policy might have on the environment, and you have eliminated any negative effects.
Explanation: Answering “Yes” to this question, it means that when developing your policy, you have thoroughly considered all the effects that your policy could have upon the environment, and you have eliminated any conditions that would impose negative effects. If answered otherwise, it means that you have either not thought about the impact your policy could have on the environment, or you are not able to control the impact your policy has. For example, another car insurance policy could favour the customers with non-electric cars and thus people would buy more non-electric cars . This path will contribute to the car emissions environmental problem and cause more environmental problems.

- You continuously ensure that your policy adhere to the above requirements and in case an issue arises from a policy of yours, it is clear who is accountable for the issue.

Explanation: Answering “Yes” to this question, it means that as a company you follow continuously make sure that your policy does not violate any of the above requirements and ensure that you are making the most out of your policy without compromising other qualities. Also, another important aspect of accountability is that when an issue arises regarding a policy of yours, it can be easily figured out who is accountable for the issue. If answered otherwise, it means that you did not establish processes and standards to enforce the above requirements. A company can be accountable if there are teams that are responsible for checking the cause and effects of your policy and tackle any problems that might arise. Also, in case of an issue e.g. a data breach, it can be easily tracked who is responsible and act accordingly.

Lots of the above questions, put in question the business models and intellectual property of a company. We are no lawyers and we, most definitely, do not require from companies to have accessible to the public their business models and intellectual property. Although, we assume that if needed, a certified person can have access to such data.

4.1.3 Argumentative Compliance System

In this section, we will describe how the system works, from a top-down approach, given the inputs how we result to the represented output.

4.1.3.1 Types of Checks

In the last 2 sections, we explained how the user will have to create the input file and defined the questions that the policy-maker will have to answer. The file will be used to do a “Parameters” check and the answer to the questions will be used to do a “Meta-level” check.

The “Parameters” check, concerns the actual policy and the values that its various parameters can take. For example, if a policy has the age of the customer as a parameter, we will check the outcome of a policy upon different age values. The “Meta-level” check, concerns the information of the policy that cannot be extracted directly from the

policy per se, such as if processes were followed to ensure the privacy of the consumers are not being violated.

Based on the results of the checks, some values and/or requirements might be violated. In the next subsection, we define when a value or a requirement is being violated.

4.1.3.2 Values and Requirements

As stated by the European High-Level Expert Group on AI, there are 7 values and requirements. Below, we give an interpretation for the ones that can be applied in the policy context, as some requirements are specifically for AI systems and thus cannot be applied to a policy, and we state at which of the 2 checks the value or requirement belongs to.

Out of the 7 requirements, we focus on these 5:

- **Privacy and Data Governance**

Interpretation: Adherence to these requirements holds when one, in the business context it should be the person responsible for customer privacy and data protection, can evidently show that, any collected personal data of customers are safely stored and processed. Such evidence could be, and not limited to, 1) evidence of compliance with the General Data Protection Regulation (GDPR), or a non-European equivalent and 2) in no circumstances this information is used against an individual and its integrity, which can be shown by explaining how the data are used and processed.

- **Transparency**

Interpretation: Transparency consists of traceability and explainability. When these 2 values are satisfied then transparency holds. We define that traceability holds when one can provide the data used to create the conditions and set the requirements of a policy. And we define that explainability holds when one can explain the decision-making path that led to a policy outcome. The one responsible to provide such details, if needed, is usually the team or person which made the policy.

- **Diversity, Non-discrimination and Fairness**

Interpretation: In our implementation the values of diversity and non-discrimination are used interchangeably. We define that the values hold when a policy does not have different outcomes when presented various personal characteristics e.g. age, political and ethnic views and other details of this nature. Whereas fairness holds when a policy does not have different outcomes when presented similar cases that they differ on parameters that the customer does not control e.g. occupation, income.

- **Societal and Environmental Well-being**

Interpretation: These two requirements hold when the person or team that made a policy are able to demonstrate that the policy does not negatively affect the broad society and/or the environment e.g. a policy might promote an action that has negative consequences on the environment. To show that a policy respects the requirements, the responsible person or team should present research executed on how the policy might affect the society and environment in both short and long term.

- **Accountability**

Interpretation: Accountability holds 1) when a company continuously ensures that its policies adhere to the above, and not limited to, requirements and values (this can be shown from a company by explaining the policymaking and sustenance procedures) and 2) offers a process for third-parties to audit their policies.

For all the values and requirements, but the Diversity, Non-discrimination and Fairness, we require details that usually are not mentioned on a contract. For example, regarding the Transparency requirement, one can not find in a contract the reasons why a condition/field is in the contract e.g. for a car quote request, there is no explicit explanation why the age is a required field. This is the “Meta-level” check, and it’s a series of yes or no questions.

For the Diversity, Non-discrimination and Fairness values, we can check if a contract violates them or not by testing the parameters of a contract e.g. age, gender, with some values and check the outcomes for each case. In the following section we explain this

procedure in detail. This is the “Parameters” check, and we can see if the policy has different outcomes or not based on which option is enabled given certain facts, using the argumentative input file.

4.1.3.3 Processing the Inputs

Now that we have stated what inputs we need, why we need them and what they mean, we will elaborate on how we use them to extract our output.

The core functionality of our tool, happens with an argumentative based system just like how the input policy representation does. The input represents the conditions of a policy whereas our argumentation system represents the scenarios when each of the values and requirements, as stated in the previous subsection, hold or not.

The representation has 13 facts that are enabled based on the data we get when processing the inputs. The facts that might be enabled when performing the “Parameters” check are:

- *does_not_depend_on_age* → Enabled if for various ages, the policy returns the same option for all.
- *does_not_depend_on_nationality* → Enabled if for various nationalities, the policy returns the same option for all.
- *does_not_depend_on_gender* → Enabled if for various genders, the policy returns the same option for all.
- *does_not_depend_on_income* → Enabled if for various incomes, the policy returns the same option for all.
- *does_not_depend_on_marital_status* → Enabled if for various marital statuses, the policy returns the same option for all.
- *does_not_depend_on_occupation* → Enabled if for various occupations, the policy returns the same option for all.

If all are enabled, then for the given policy, the values of Diversity, Non-discrimination and Fairness hold and thus the policy adheres to them. In case any of them is not enabled, the values do not hold and the system will explain via the output which parameters cause the noncompliance. For example, if all are enabled but the

does_not_depend_on_age fact, it will explain that the values do not hold because there are some age values that different options are being enabled. We will get to more detail on how the output is being constructed in the next subsection.

The facts that might be enabled when performing the “Meta-level” check are:

- *explanation_available* → Enabled if answered “Yes” at the first [policymaker question](#).
- *facts_available* → Enabled if answered “Yes” at the second [policymaker question](#).
- *respects_privacy* → Enabled if answered “Yes” at the third [policymaker question](#).
- *established_data_governance* → Enabled if answered “Yes” to the fourth [policymaker question](#).
- *thought_about_societal_well_being* → Enabled if answered “Yes” to the fifth [policymaker question](#).
- *thought_about_environmental_well_being* → Enabled if answered “Yes” to the sixth [policymaker question](#).
- *accountability* → Enabled if answered “Yes” to the seventh [policymaker question](#).

In case “No” is selected, the specific fact that would be enabled otherwise, it won’t be enabled and this will be interpreted as there is no compliance with the specified requirement. In such cases, the system will display various sources where one can read as to be able to adhere to the specified requirement.

When the *explanation_available* and *facts_available* are both enabled, the Transparency requirement holds. The former fact represents the requirement of Explainability and the latter fact represents the requirement of Traceability. When the *respects_privacy* and *established_data_governance* facts are both enabled, the Privacy and Data Governance requirement holds. Moreover, when the *thought_about_societal_well_being* and *thought_about_environmental_well_being* facts are enabled, the Societal and Environmental Well-being requirement holds respectively. Lastly, when the *accountability* fact is enabled, the Accountability requirement holds. The explanations are extracted in the same way as described for the “Parameters” check.

The full argumentation file is in Appendix C, and we advise to give it a look before continuing to the next subsection, as we will refer to its contents to explain how our systems constructs its explanations. Finally, we will get in more detail on how the checks happen and how the system works, from a technical view, in the next chapter.

4.1.3.4 Output - Explanations

In this section, we will focus on the argumentation level extraction of explanations but in the next chapter we explain how from argumentative explanations we construct the Natural Language (NL) explanations.

For demonstration purposes, we will focus on the cases that the Transparency requirement holds and does not hold. To start with, this is the code concerning the requirement:

```
complement(promotes(Value),demotes(Value)).
complement(demotes(Value),promotes(Value)).

abducible(facts_available, []).
abducible(neg(facts_available), []).
abducible(explanation_available, []).
abducible(neg(explanation_available), []).

rule(r1(Value),promotes(Value),[]).
rule(r2(Value),demotes(Value),[]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                               Default - All Demoted                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

rule(pr1(Value),prefer(r2(Value),r1(Value)),[]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                               Transparency                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Traceability
rule(pr2(traceability),prefer(r1(traceability),r2(traceability)
),[facts_available]).
```

```
rule(c1(traceability),prefer(pr2(traceability),pr1(traceability)),[]).
```

```
% Explainability
```

```
rule(pr3(explainability),prefer(r1(explainability),r2(explainability)),[explanation_available]).
```

```
rule(c2(explainability),prefer(pr3(explainability),pr1(explainability)),[]).
```

```
% Transparency
```

```
rule(pr4(transparency),prefer(r1(transparency),r2(transparency)),[facts_available,explanation_available]).
```

```
rule(c3(transparency),prefer(pr4(transparency),pr1(transparency)),[]).
```

Based on the above code, we can see that there are 2 options: to promote or to demote a value. The default scenario is to demote all values until proven otherwise. From rules pr2, pr3 and pr4, we know that the transparency value gets promoted when the *facts_available* fact is present, the explainability value gets promoted when the *explanation_available* fact is present and the *transparency* value gets promoted when both of the aforementioned facts are enabled.

In case both of the facts are enabled, if we query the system if transparency is promoted, we get the following explanation:

```
Exp1 = [nott(c3(transparency)), pr4(transparency), f2, f1,
        c3(transparency), r1(transparency)] .
```

To get the above explanation, we enabled both of the facts (they are represented by f2 and f1) and then used this query:

```
?- prove([promotes(transparency)],Exp1).
```

We ignore the first nott symbol of the explanation, and we see that the rule pr4 holds, because the f2 and f1 facts are enabled and thus the option promotes(transparency) is preferable over demoting the value. The preference is set by the rule c3.

When both facts are enabled, it's that straight forward to extract the explanation, but now we will check the case where 1 of the facts is disabled and point out how the explanations are different.

Let's assume that the fact *facts_available* is disabled then the explanation of the same query is:

```
Expl = [nott(c3(transparency)), pr4(transparency), f2,  
ass(facts_available), c3(transparency), r1(transparency)] .
```

We ignore the first *nott* symbol of the explanation, and we see that the rule *pr4* holds, because the *f2* is enabled, and *we assume* that *f1* is enabled as well and thus the option *promotes(transparency)* is preferable over demoting the value. The preference is set by the rule *c3*.

We notice that *f1* has been replaced with *ass(facts_available)*. This happens due to the *abducible* statements that we defined before the rules. We use the *abducible* statements to be able to see what facts are missing for a value to be promoted and furthermore *explain why* a value is not being promoted. This is a crucial feature because we want our system to be as explainable as possible, as to be able to assist users to the maximum when evaluating their policy.

When an *ass* statement is present in an explanation, it means that a fact is not enabled and thus the specified value cannot be promoted. Based on the above explanation, the NL explanation will state that the value of Transparency is demoted because the Traceability requirement does not hold, but it will also note that the Explainability requirement holds.

Chapter 5

System Development Cycle

5.1 System Requirements	34
5.2 System Architecture & Design	35
5.2.1 HTML/CSS - Front-end	36
5.2.2 JavaScript - Middleman	39
5.2.3 Back-end	39
5.2.3.1 PHP	40
5.2.3.2 Argumentative Compliance System	40
5.2.3.2.1 Java	41

5.1 System Requirements

In Chapter 4, we have described our methodology, and now we will analyse the system we developed upon it. To begin with, we will state what is required from a user to have to be able to use our system.

The system is a Web Application which encapsulates all the processes we have described in Section 4.1.3 of Chapter 4. The web app is accessible by any web browser, however some functionalities might differ from one web browser to others. The system was developed and tested with Google Chrome and Firefox Web Browser.

When a user accesses the web app with her/his web browser of preference, there are 2 functionalities: 1) policy assessment and 2) a questionnaire. As we will see later on in the guide, the user can navigate to any of the above functionalities via a navigation bar or other action buttons. The latter functionality, does not require from the user anything other than following instructions and choosing a preferred option in the questionnaire.

The former functionality, that is the process of assessing if a policy is ethical or not, requires an input file. This input file is the argumentation based file of the policy, which we explained in detail what it exactly is and how to create in Section 4.1.1 of Chapter 4. This information is also available in the web app, so the user can see the guide directly from the web app.

Other than a web browser to be able to access the web application, and an argumentative policy representation (if the user wants to take the Policy Check process), our system has no other requirements. Moving on, we will focus on the architecture and design of our system.

5.2 System Architecture & Design

Our web app, is a simple Client - Server model (as shown in Figure 5.1), where the server side consists of 3 major components. There is the Front-end, which is what the clients see, the Back-end, where all the functionalities of our system happen and the middle-layer which interconnects the 2 aforementioned components. The middle-layer takes requests from the front-end and sends them to the back-end for a functionality to happen, and if there is a result from the back-end, the middle-layer presents the results to the front-end. We will break down each component, mentioning the technologies used for each one and abstractly describe their work.

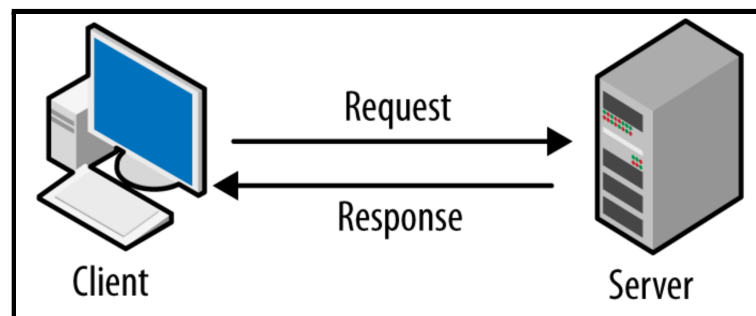


Figure 5.1: Client-Server Model

5.2.1 HTML/CSS - Front-end

This component is what the end user actually sees. What is being displayed, has been structured using the HTML markup language and the design was applied using the CSS style sheet language. To be more specific, we used the Bootstrap CSS framework to design a responsive web application with ease.

These are 4 main screens the end user sees:

- Landing page, where there is a small caption regarding the goal of our tools and why they are important and why we need to use them. From the two buttons in the bottom left and bottom right, the user can navigate to the specified tool. Also, on the top there is a navigation bar where the user can use to navigate to the tools and documentation as well.

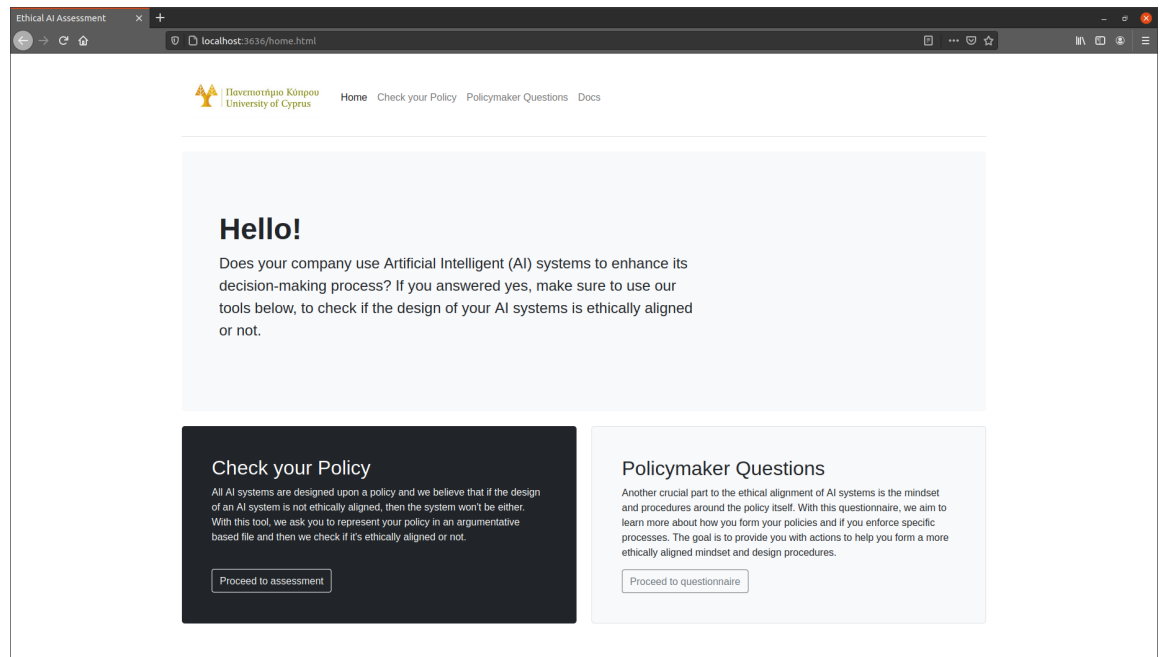


Figure 5.2: Landing Page

- Policy Assessment Tool, where there is a flow that the user follows as to assess its policy. In Figure 5.3, we can see the initial view of the Policy Assessment tool and by pressing the “Begin assessment” button, more elements will be added to the view where they will guide the user step by step on how to use the tool. We won’t show the rest elements in this section as we will go over them in the next chapter which is a guide for the web application as a whole.

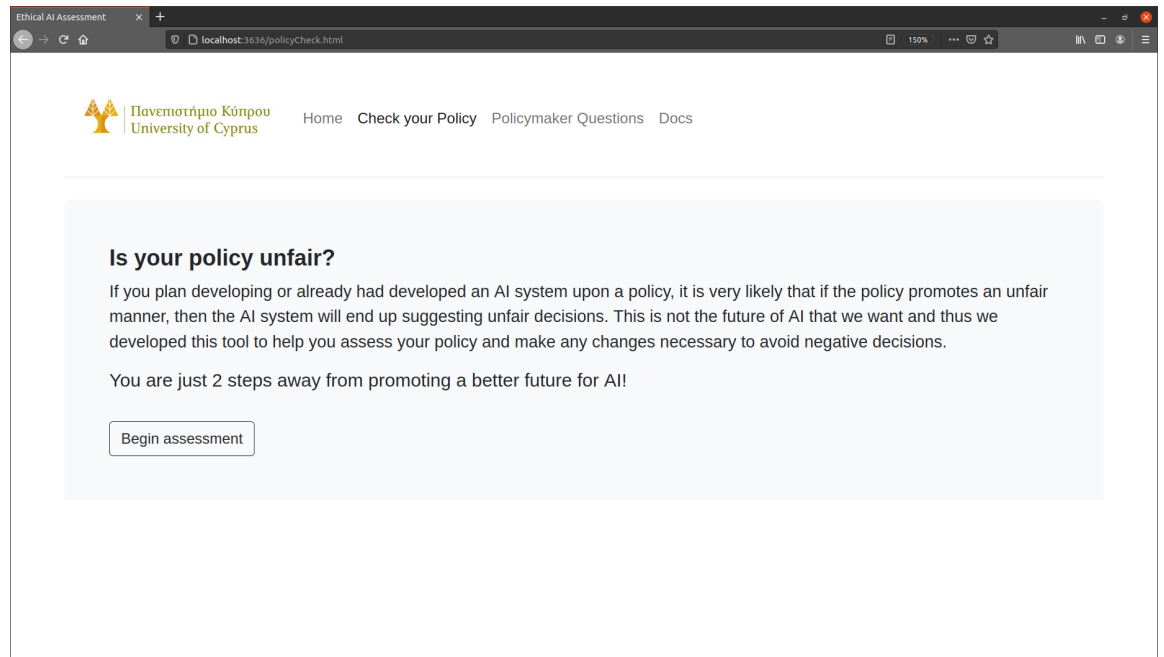


Figure 5.3: Initial Check your Policy Tool Page

- Policymaker Questions, where the questionnaire that concerns the policymakers is located at, with the initial view (see Figure 5.4) being an introduction for the questionnaire. By pressing the “Let’s find out” button, the questionnaire is shown, as we can see in Figure 5.5. Then by pressing submit, we get the results of the evaluation. We will see the results view in the next chapter as well.

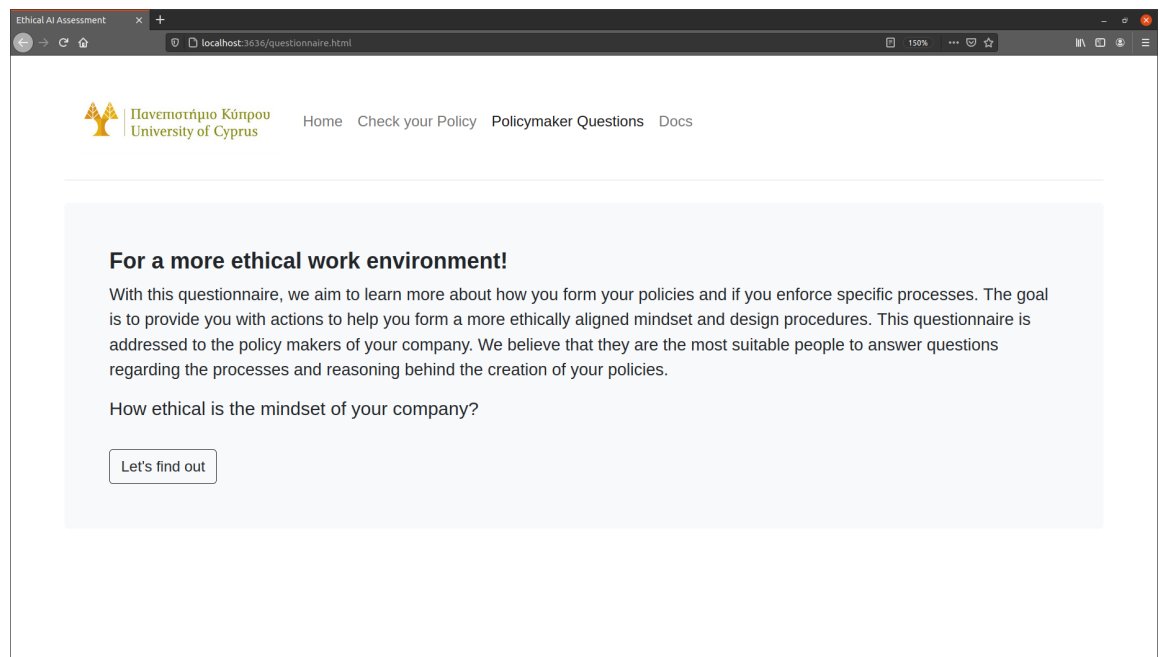


Figure 5.4: Initial Policymaker Questionnaire Page

Let's find out

The Questionnaire

Choose an option between "Yes" and "No" for each statement below, and in case you are not sure which one applies to you, choose "No". For all statements you chose "No", we will provide you with more details on how you can be compliant with each of those statements (thus be able to choose "Yes").

If needed, the reasoning behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
If needed, the supporting data behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
Your policy does not have any conditions that violate the consumers' privacy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have established a data governance process regarding your policy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the society and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the environment and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
It is clear and straight-forward who is accountable and for what someone is accountable when an issue arises from your policy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No

Submit

Figure 5.5: Polycymaker Questionnaire View

- Documentation, this is a long one-page documentation where we explain in detail how the user can create the required argumentative input file for the Policy Assessment Tool. In particular, it is a web-view of Section 4.1.1.

Πανεπιστήμιο Κύπρου
University of Cyprus

Home Check your Policy Polycymaker Questions Docs

How to Formalize Your Policy

In this documentation page, we will guide you on how you can create the argumentation based representation of your policy, which then you can use to proceed to the Policy Check. For demonstration purposes, we will define a policy of our own and build upon it.

Step 1: Your Policy in Natural Language

It is important that you have a Natural Language representation of your policy. Our demo policy, is a car insurance policy based on which an insurance company calculates the premium of the insurance.

Demo Car Insurance Policy - Natural Language

The basic third-party car insurance plan costs 180 euros per year. If the driver is younger than 24 years old or older than 69 years old, then there is a 20% increase. Also, there is a 5% increase when the driver has been licensed for less than 2 years. Finally, if the customer lives in a high crime area, there is a 10% increase.

Step 2: Identify the Goal of Your Policy and the Parameters that Contribute to it

Considering the Natural Language form of your policy, you have to identify what is the overall goal you try to achieve with your policy, and what parameters affect the achievement of that goal. For our demo policy we can see that:

Demo Car Insurance Policy - Goal and Parameter Space

- Goal: Decide premium cost
- Parameters that affect the decision
 - Age
 - Years being a licensed driver

Figure 5.6: Part of Documentation Page

These were the main views from which the end user interacts with our system. The behaviour of the application is defined by how the middle-layer interacts with the front-end and back-end. In the next section we explain how this communication happens.

5.2.2 JavaScript — Middleman

Our web application is an interactive and dynamic web page; based on user interactions, the web page is modified accordingly. There are buttons with which the user clicks to navigate, option buttons where the user selects a preference/option and there is an option to upload a file as well. What defines what happens after every such action, is the JavaScript (JS) code of our web app. More specifically, we use the jQuery JS library as it provides shortcuts and easier methods to implement the functionalities we want.

The JS code is mainly responsible for

- Redirecting the user to the appropriate view when the user clicks on a navigation button;
- Saving the options that the user has selected e.g. when answering the questionnaire;
- Handling any uploaded files;
- Sending any information or files to the server as to execute a procedure e.g. evaluate a policy or a questionnaire; and lastly
- Show dynamically any results from the server.

Now that we have defined the responsibilities of the middleman, we will focus on the core functionality of our web app which is the Back-end component.

5.2.3 Back-end

The Back-end is where all the core functionality of our web application happens. This component is responsible for all the processes mentioned in Sections 4.1.3.3 and 4.1.3.4 of Chapter 4. When a user via the Front-end request to either assess her/his policy, the middle-layer provides the Back-end with all the necessary information, the back-end proceeds to executing the desired process, and then it replies to the middle-layer with the results and then the middle-layer presents the results to the end user. In this section we will explain how we implement the execution of the desired process. To do that, we worked with various programming languages like PHP, Java and PROLOG.

5.2.3.1 PHP

The first part of our back-end, is the PHP code. When the middle-layer sends a request to execute a procedure, the PHP code will take all the information and prepare them as to execute the request.

Depending on which tool the request comes from, policy assessment or policymaker questionnaire, the code prepares the information as to pass them to our Argumentative Compliance System. If the request is about a policy assessment, it expects an argumentative policy representation file and the parameter values of the policy. If it is for the policymaker questionnaire, then it expects “Yes” or “No” options.

Moving on, if all the information are present, it executes the request by passing the information to the Argumentative Compliance System. This is done by executing a command line command which runs the system with the specified information. When the execution is over the returned data, from the Argumentative Compliance System returned to the PHP code, are then parsed in a specific format (JSON) which the middle-layer later on interprets and displays them to the user.

This is the functionality of the PHP part of the back-end; it acts as a glue between the middle-layer and the core back-end functionality, which we will get in more detail in the next section.

5.2.3.2 Argumentative Compliance System

The Argumentative Compliance System is the heart of our system as it encapsulates our methodology. The methodology describes in Sections 4.1.3.3 and 4.1.3.4 of Chapter 4, are being implemented in this component. This is the component where we apply Argumentation and use the power of Argumentation to support or demote decisions.

This system consists of 2 components, a Java package and the GORGIAS framework. The Java component acts as a wrapper class of the GORGIAS framework, as there is a library (JPL) that connects PROLOG execution with Java. In the Java package, we prepare the argumentation information which we will use to run the PROLOG files, and then interpret the results and explanations.

5.2.3.2.1 Java

The PHP code, calls our Java Package which based on the given information, it prepares the Argumentation files as to be able to run PROLOG queries on them. As mentioned before, there are 2 processes that can be requested and here is what the package does for each one of them:

- Policy Assessment

When the request is for assessing a policy, the package does the following actions:

1. It queries the given policy the default scenario and stores the default outcome/option.
2. Then, for each parameter value, it queries the given policy and checks if the outcome/option is different from the default one.
3. If the outcome is different, it stores which parameter value caused the variation and which parameter is about e.g. if a parameter value of the “Age” parameter has different outcomes, then we will store which parameter value is and that is regarding the “Age” parameter.
4. After storing the parameter value or if the outcome is the same, repeat steps 2 and 3 till all parameter values are exhausted.
5. When all values are exhausted, we go over the collected information from step 3 and construct facts that describe the behaviour of the policy. To be specific, we can construct the following facts:
 - does_not_depend_on_age if there are no Age parameter values that cause a different outcome.
 - does_not_depend_on_nationality if there are no Nationality parameter values that cause a different outcome.
 - does_not_depend_on_gender if there are no Gender parameter values that cause a different outcome.
 - does_not_depend_on_income if there are no Income parameter values that cause a different outcome.
 - does_not_depend_on_marital_status if there are no Marital Status parameter values that cause a different outcome.
 - does_not_depend_on_occupation if there are no Occupation parameter values that cause a different outcome.

These facts, will be used to set the knowledge for our argumentative file, which will then act as knowledge for our GORGIAS file (see Appendix C).

6. After defining the knowledge from Step 5, we query our GORGIAS file, and when get the results, we process them to see if the input policy is unfair or not and if it is, we see what causes the unfair behaviour.
7. Based on the returned GORGIAS format information, we format them and reconstruct them into a more Natural Language form and return the results to the PHP code.

- Policymaker Questionnaire

When the request is evaluating the policymaker questionnaire selected options:

1. The PHP code sends the Java package the selected option for each statement and based on which option was selected, we might construct a fact. Specifically, the following facts get constructed if the specified options are selected:
 - explanation_available if “Yes” is the selected option to the first policymaker question.
 - facts_available if “Yes” is the selected option at the second policymaker question.
 - respects_privacy if “Yes” is the selected option at the third policymaker question.
 - established_data_governance if “Yes” is the selected option to the fourth policymaker question.
 - thought_about_societal_well_being if “Yes” is the selected option to the fifth policymaker question.
 - thought_about_environmental_well_being if “Yes” is the selected option to the sixth policymaker question.
 - accountability if “Yes” is the selected option to the seventh policymaker question.

As mentioned in the Methodology chapter, in case “No” is selected, the specific fact that would be enabled otherwise, it won’t be enabled and this

will be interpreted as there is no compliance with the specified requirement.

Similarly to the other procedure, these facts, will be used to set the knowledge for our argumentative file, which will then act as knowledge for our GORGIAS file (see Appendix C).

2. After defining the knowledge from Step 1, we query our GORGIAS file, and when get the results, we process them as to provide information to the user on how they can comply with a requirement that they currently don't.

Chapter 6

System User Guide

6.1 Policy Assessment Guide	44
6.2 Policymaker Questionnaire Guide	50

6.1 Policy Assessment Guide

We will demonstrate how to use the system we developed, based on the demo policy mentioned in Chapter 4.

First, we go to the website, and we see the landing page:

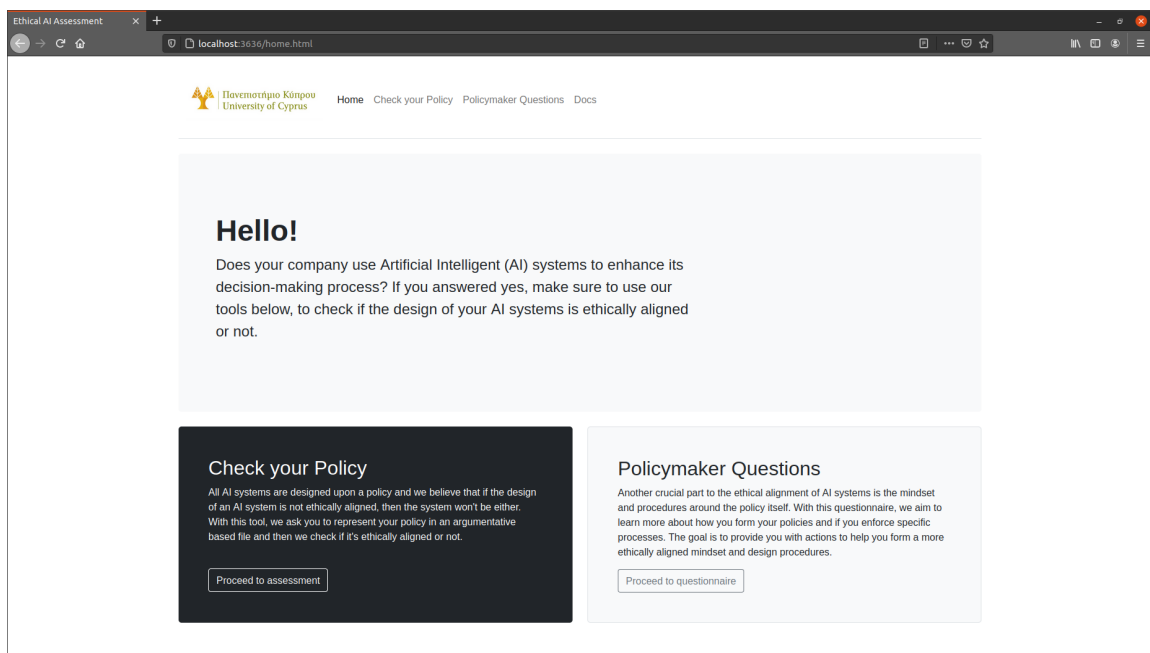


Figure 6.1: Landing Page

Then, because we want to assess our policy, we can either select the “Check your Policy” tab in the navigation bar (navbar) or click the “Proceed to assessment” in the bottom left box. After we press one of the mentioned options, we get to the main screen of the “Check your Policy” tool.

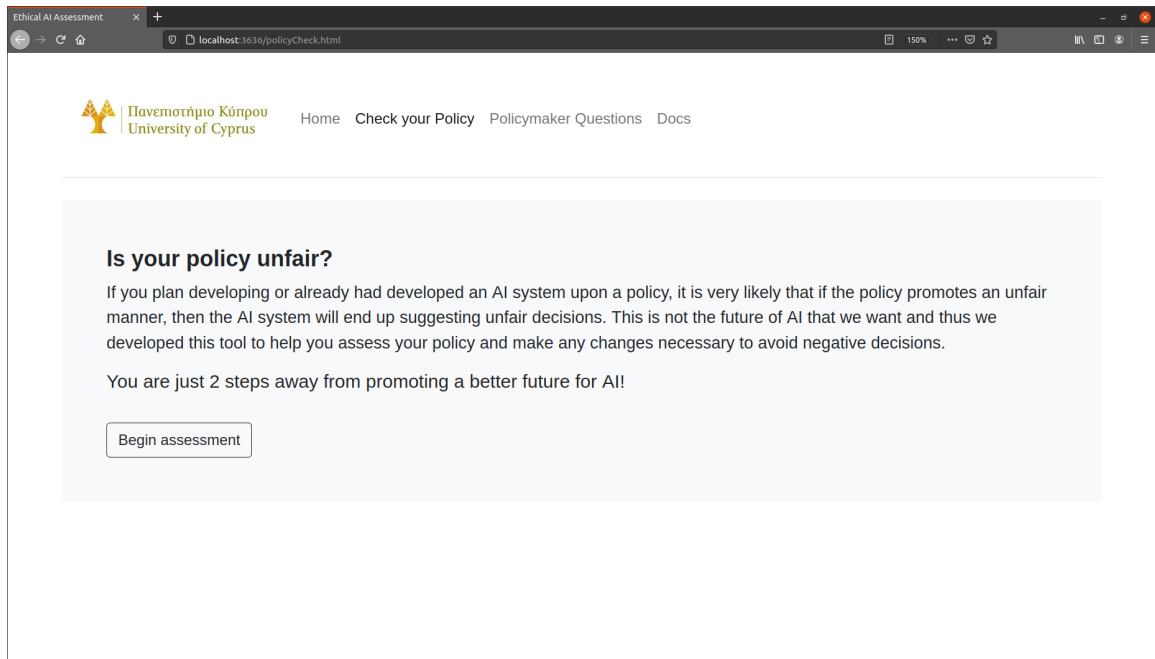


Figure 6.2: “Check your Policy” Initial page

We see a description and the goal of the tool. To begin the assessment procedure, we click the “Begin assessment” button, and we can see the 1st step of the procedure.

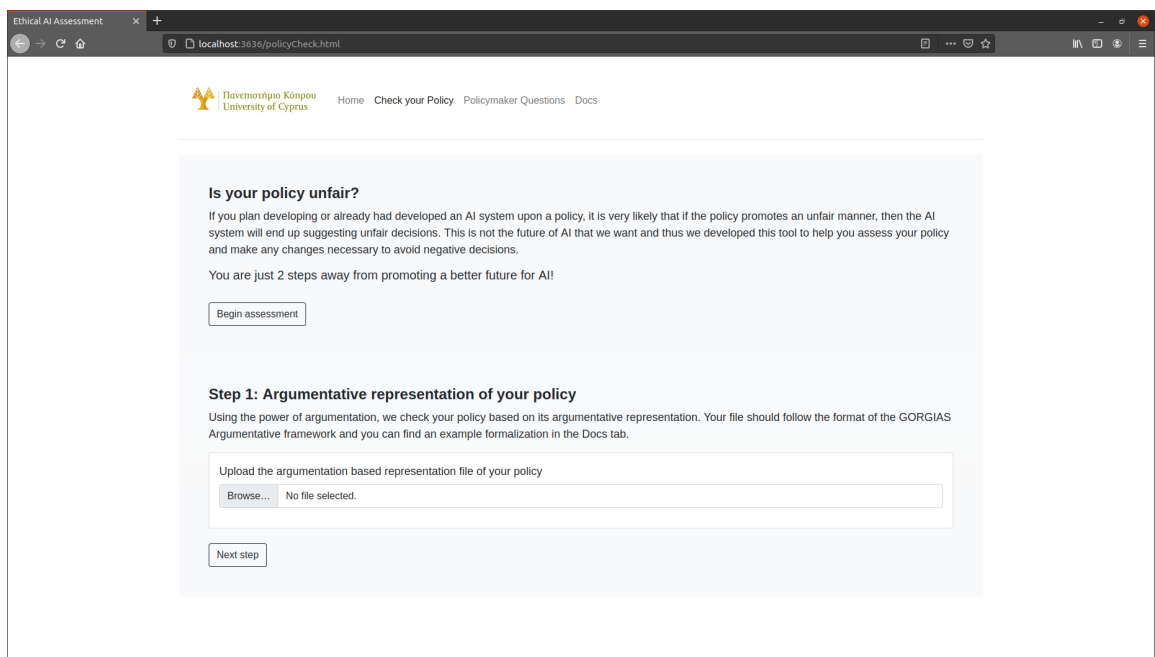


Figure 6.3: “Check your Policy” Step 1

The first step, as mentioned in previous chapters, it is the formalization of our policy into an argumentative file. The description of Step 1 states that and if the user does not know

how to do that, it informs the user that by navigating to the Docs tab, there is a guide on how to construct the file.

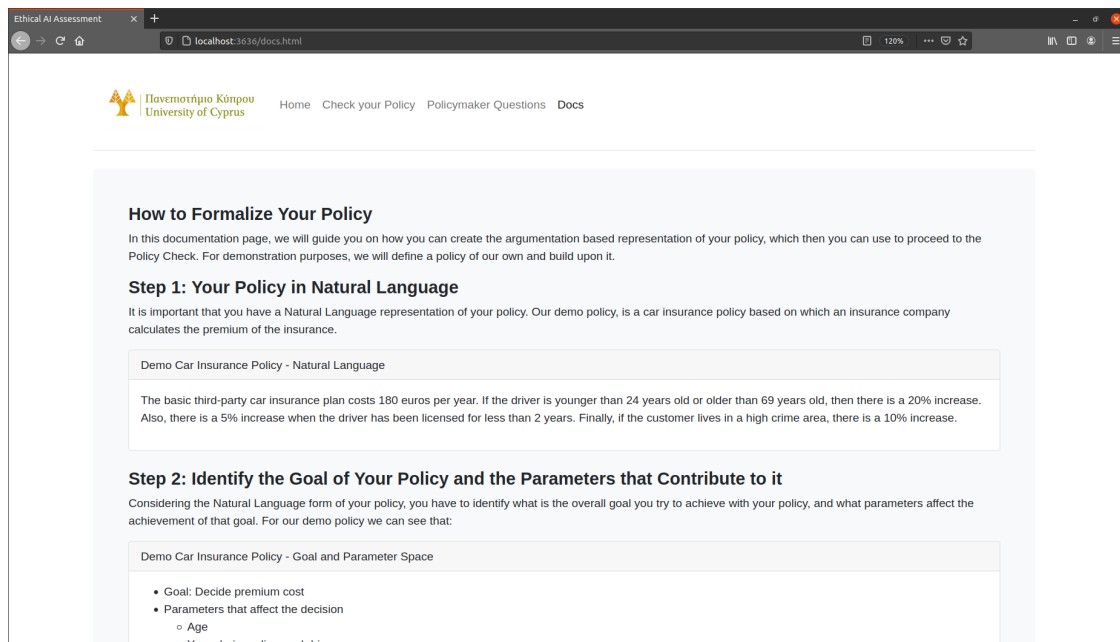


Figure 6.4: Part of Documentation Page

Following the guide, with our demo policy, we end up with the argumentative file as shown in Appendix B. After we upload the file, we click on “Next step” button and the 2nd step is shown. Here, we have two options “N/A” and “Custom”.

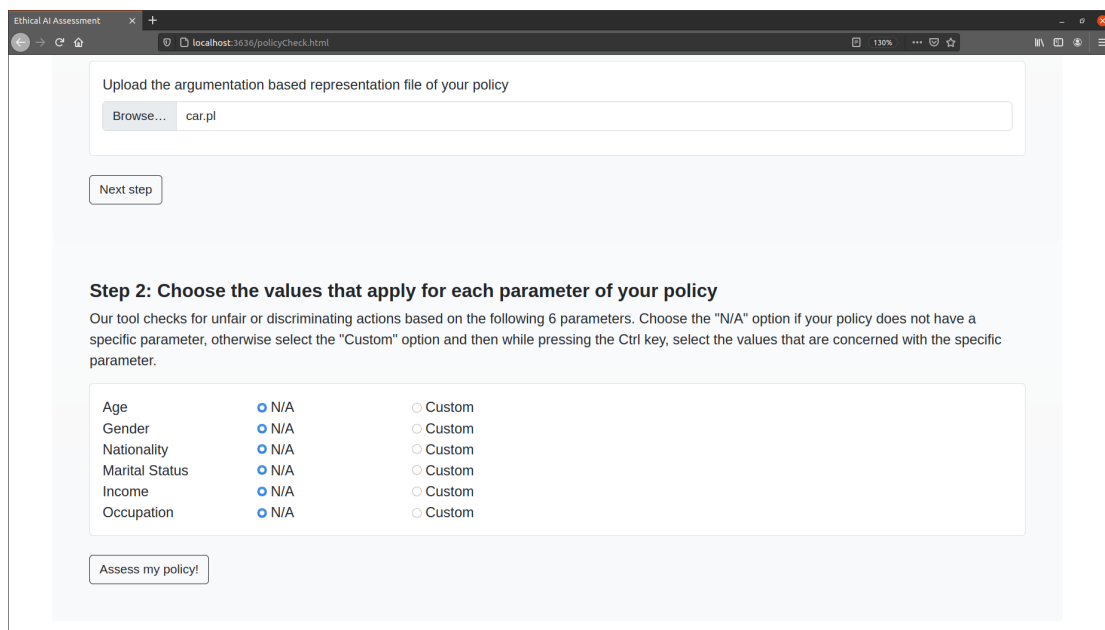
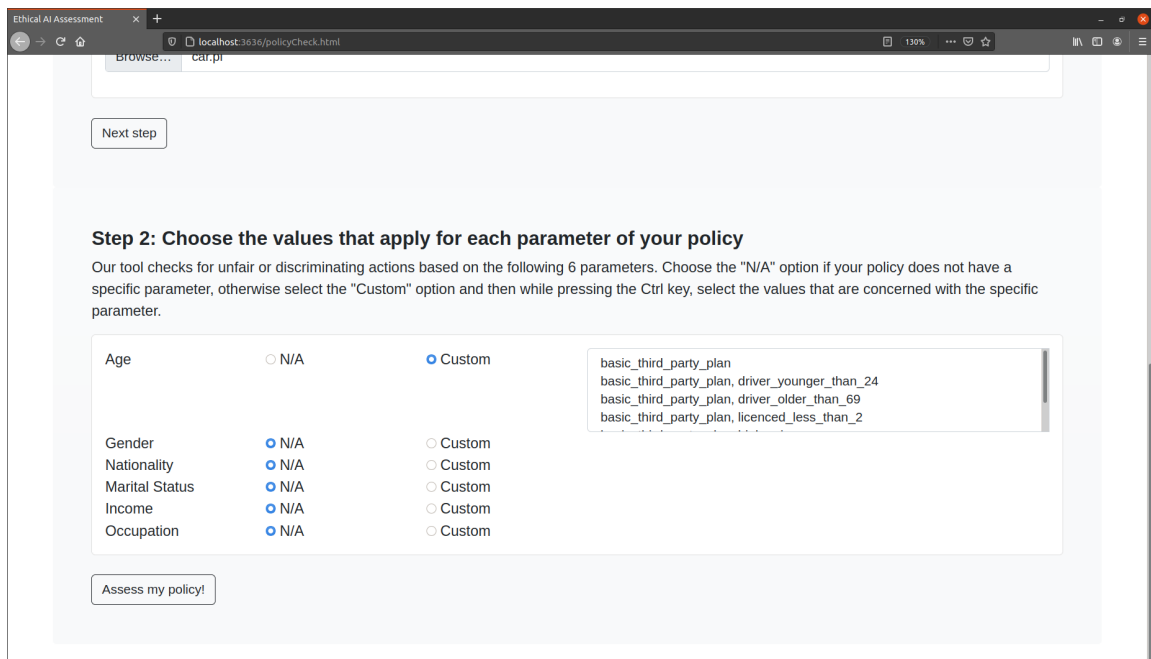


Figure 6.5: “Check your Policy” Step 2

We choose the “N/A” option if our policy does not have the specified parameter e.g. our policy doesn’t have a parameter regarding all the parameters but age, and we choose the “Custom” option as to select the parameter values for a specific parameter.

Note that our policy has other parameters, such as years being licensed or residence area, but the tool at the time being does not have a broad set of parameters to work with. We will specifically mention this in the Future Work.

Moving on, we select the “Custom” option for the Age parameter, and we get this list.



The screenshot shows a web browser window titled 'Ethical AI Assessment' with the URL 'localhost:3036/policyCheck.html'. The page is at 'Step 2: Choose the values that apply for each parameter of your policy'. It explains that the tool checks for unfair or discriminating actions based on 6 parameters. For 'Age', the 'Custom' option is selected, and a list of policy scenarios is shown. For the other parameters (Gender, Nationality, Marital Status, Income, Occupation), the 'N/A' option is selected. A 'Next step' button is at the top, and an 'Assess my policy!' button is at the bottom.

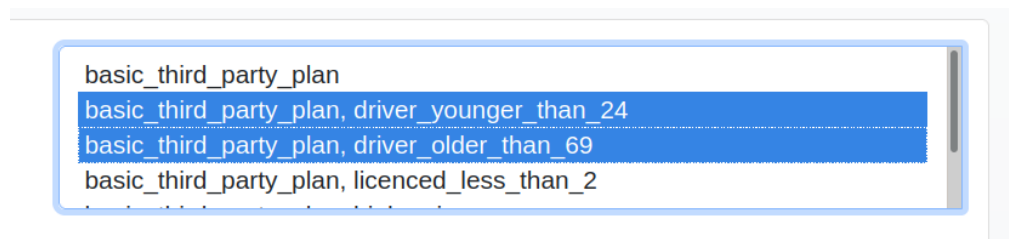
Parameter	N/A	Custom
Age	<input type="radio"/>	<input checked="" type="radio"/>
Gender	<input checked="" type="radio"/>	<input type="radio"/>
Nationality	<input checked="" type="radio"/>	<input type="radio"/>
Marital Status	<input checked="" type="radio"/>	<input type="radio"/>
Income	<input checked="" type="radio"/>	<input type="radio"/>
Occupation	<input checked="" type="radio"/>	<input type="radio"/>

Selected Age Values:

- basic_third_party_plan
- basic_third_party_plan, driver_younger_than_24
- basic_third_party_plan, driver_older_than_69
- basic_third_party_plan, licenced_less_than_2

Figure 6.6: Custom Options List

We can see that this list contains the scenario values of our policy. What we have to do is to select the scenarios that apply to the “Age” parameter. We can select one option by clicking on the option we want or, we can select multiple options by pressing the Ctrl key and clicking on the options we want. We will select the 2nd and 3rd option.



The screenshot shows a list of policy scenarios for the 'Age' parameter. The second and third options are highlighted with a blue background, indicating they have been selected.

- basic_third_party_plan
- basic_third_party_plan, driver_younger_than_24
- basic_third_party_plan, driver_older_than_69
- basic_third_party_plan, licenced_less_than_2

Figure 6.7: Selected Age Values

After we select the parameter values for each parameter, we click the “Assess my policy!” button. This will trigger the request to the Back-end to assess the given policy with the selected parameter values. While the request is being processed, we see the following waiting screen.

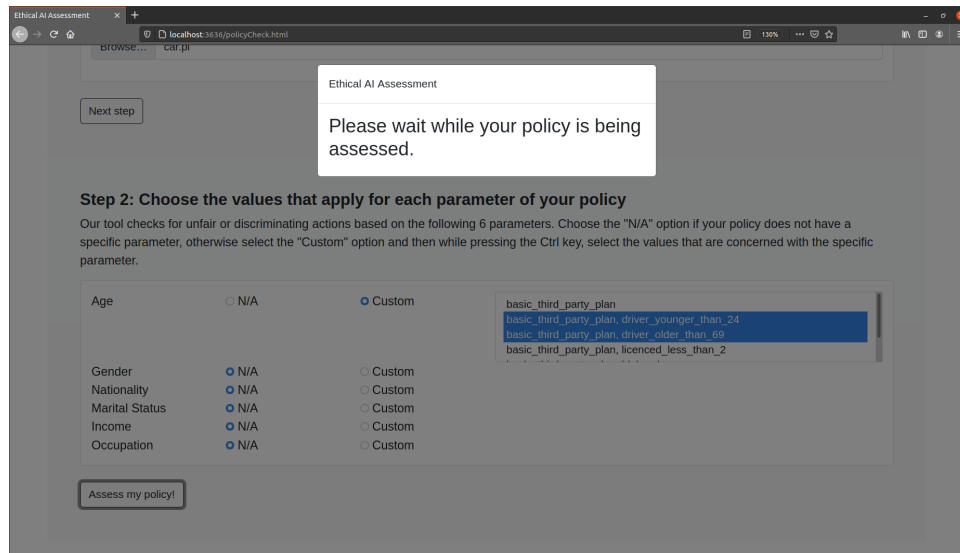


Figure 6.8: Waiting Screen

When the assessment is over, the overlay will disappear, and we can scroll down to see the results. In case there is an unfair act in our policy, we get a screen like the one in Figure 6.9.

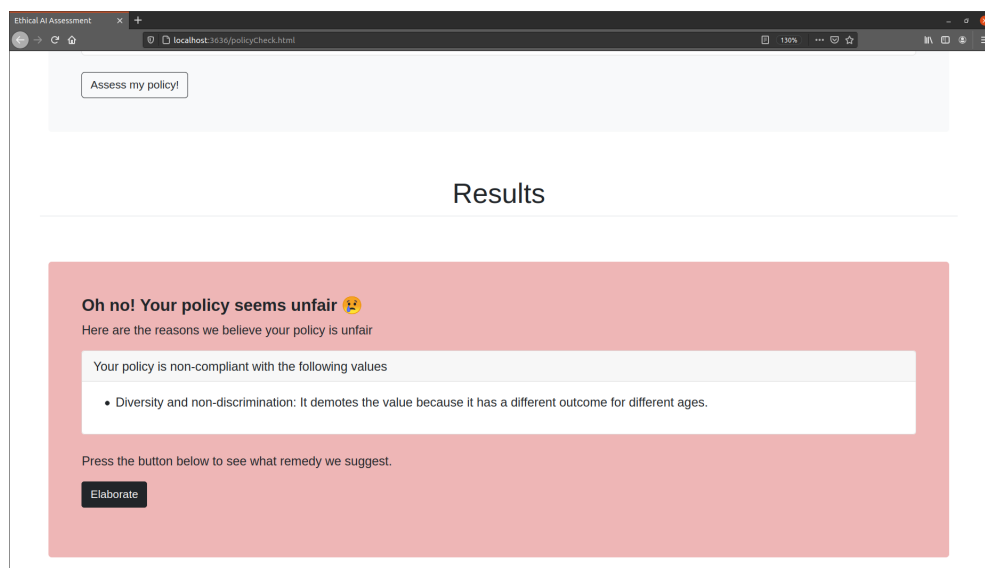


Figure 6.9: Unfair Policy Box

Paying a closer look at the red box in Figure 6.10, there are 2 main components: a white box showing which value is violated and why it is violated. In our case, we see that the reason is that different age values, have a different outcome than the default one, and another important component is the “Elaborate” button. By clicking the button, we get the extra box as shown in Figure 6.11.

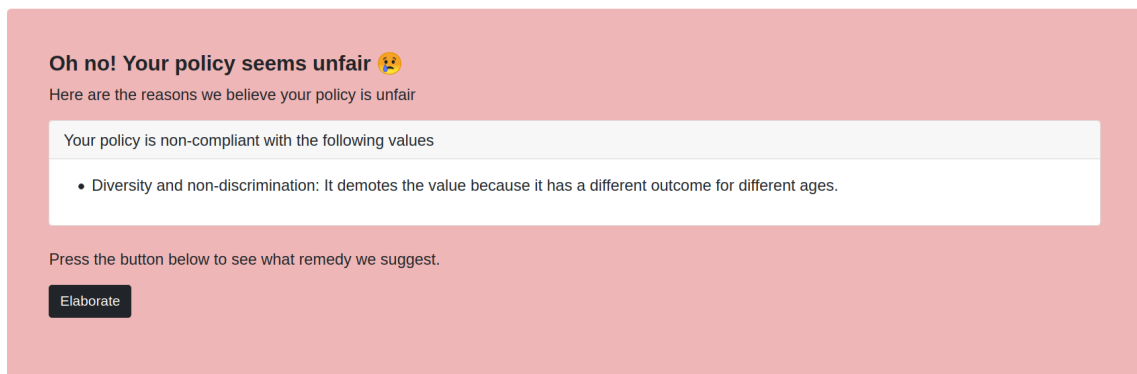


Figure 6.10: Unfair Policy Box - Close-up

An extra box is shown containing the Remedy we provide as to help you modify your policy to adhere to the values that it does not. For our policy, the proposed solution is to eliminate the different premiums for different ages, *if possible*. We understand that some policies have an unfair nature where nothing can be done to avert it.

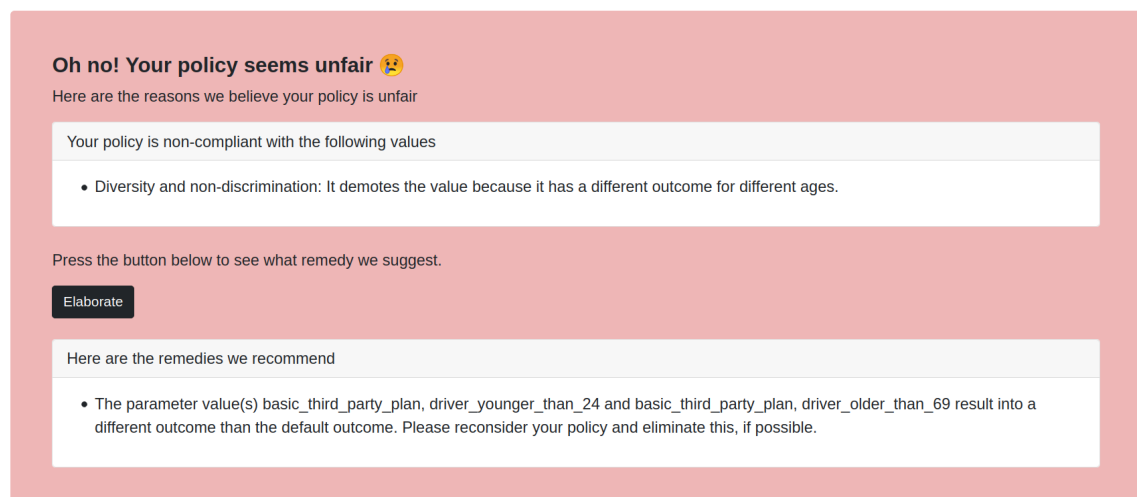


Figure 6.11: Remedy Box

Now, in case our policy does not show an unfair behaviour, instead of the red box shown in Figures 15 to 17, we get the following green box.



Figure 6.12: Fair Policy Response Box

This concludes the user guide about the “Check your Policy” tool. All figures in this section, present all the possible views of this tool.

6.2 Policymaker Questionnaire Guide

Similarly with the previous guide, when accessing the web application we land on the initial page as shown in Figure 6.1. Then, as to access the policymaker questionnaire, we can either click the “Proceed to questionnaire” button in the bottom right corner or click the “Policymaker Questions” at the navigation bar.

After selecting one of the possible ways to visit the questionnaire, the following figure is the initial page.

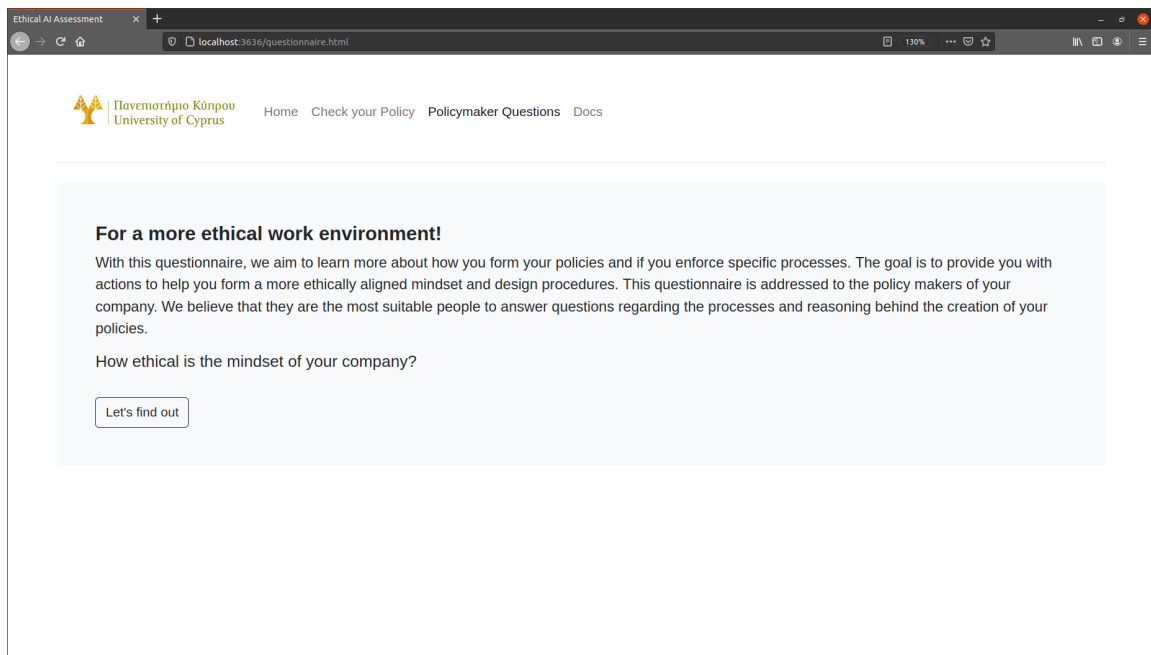


Figure 6.13: “Policymaker Questions” Initial page

We get some general information on what the goal of the questionnaire is and why one should use it. To continue, we click the “Let’s find out” button. After clicking it, the actual questionnaire will show up as we can see in Figure 6.14.

The Questionnaire

Choose an option between “Yes” and “No” for each statement below, and in case you are not sure which one applies to you, choose “No”. For all statements you chose “No”, we will provide you with more details on how you can be compliant with each of those statements (thus be able to choose “Yes”).

If needed, the reasoning behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
If needed, the supporting data behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
Your policy does not have any conditions that violate the consumers' privacy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have established a data governance process regarding your policy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the society and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the environment and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
It is clear and straight-forward who is accountable and for what someone is accountable when an issue arises from your policy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No

Figure 6.14: Policymaker Questionnaire

Then, we select the option that fits our policymaking process the best and click “Submit”. In Section 4.1.2 of Chapter 4 we described in detail the questions, why we chose the specific ones and what the goal of each question is. For demonstration purposes, we will select “Yes” for all questions except the 4th question that it’s about data governance. The figure below shows our selected options.

The Questionnaire

Choose an option between “Yes” and “No” for each statement below, and in case you are not sure which one applies to you, choose “No”. For all statements you chose “No”, we will provide you with more details on how you can be compliant with each of those statements (thus be able to choose “Yes”).

If needed, the reasoning behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
If needed, the supporting data behind a condition of the policy can be accessed.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
Your policy does not have any conditions that violate the consumers' privacy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have established a data governance process regarding your policy.	<input type="radio"/> Yes	<input checked="" type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the society and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
You have considered any long-term or/and short-term impact that your policy might have on the environment and you have eliminated any negative effects.	<input checked="" type="radio"/> Yes	<input type="radio"/> No
It is clear and straight-forward who is accountable and for what someone is accountable when an issue arises from your policy.	<input checked="" type="radio"/> Yes	<input type="radio"/> No

Figure 6.15: Selected Demo Options

Because we selected “No” at the data governance question, the system will reply that there is an issue with our procedures, and it will mention a guideline that we can read as to make any changes to establish data governance in our practice. We can see the mentioned results in Figure 6.16 in the red box.

The screenshot shows a web browser window titled 'Ethical AI Assessment' with the URL 'localhost:3636/questionnaire.html'. The questionnaire contains four questions, each with 'Yes' and 'No' radio button options. The 'No' option for the first question, 'You have established a data governance process regarding your policy.', is selected. Below the questions is a 'Submit' button. A red box highlights the 'Issues' section, which states: 'Below you can see what best-practices you can follow as to begin adopting a more ethical approach to the creation of your policies.' It then lists: 'Your policy is non-compliant with the following values' followed by a bullet point: 'Privacy and data governance: It promotes the value because you have not established data governance in your practice. You can check the European General Data Protection Regulation (GDPR) as a guideline on what you can establish as to incorporate data governance in your practice.'

Figure 6.16: Policy Making Procedure Issues

On the other hand, if we select “Yes” for all options then we get a green box saying that everything looks good.

The screenshot shows the same 'Ethical AI Assessment' web browser window. In this instance, the 'Yes' option is selected for all four questions. The 'Submit' button is visible. A green box at the bottom of the page displays the message: 'All Good! Thank you for owning an ethical work environment! You are an example for others to follow.'

Figure 6.17: Problem-free Policy Making Procedure

Chapter 7

Evaluation

7.1 Cognitive Evaluation	53
7.1.1 Process of Evaluation	53
7.1.2 The Questionnaire	54
7.1.3 Results	64

7.1 Cognitive Evaluation

The main objective of this thesis was to develop a *human-centric* system where a policy owner can check whether her/his policy can be safely implemented as an AI system. The human-centric value of our system is emphasized by the explanations we provide to the user whether the policy is safe or not.

Based on our goals, there are two things to evaluate:

1. We want to see if the broader community understands the importance of such tool and
2. we want to see if indeed our explanations satisfy the users and help them enough to make any changes to their policy, to adhere to more values.

We call this a Cognitive Evaluation as we evaluate the reasoning and beliefs of the users, before and after they see the results of our methodology. The evaluation is carried-out by a questionnaire, which we will explain in more detail in the next 2 subsections ([7.1.1](#) and [7.1.2](#)) and finally state our results in subsection [7.1.3](#).

7.1.1 Process of Evaluation

The evaluation has 2 parts: the registration part and the main questionnaire part. We first want to gather some anonymized information of the subject and then proceed to the main part.

Part 1: Registration

The subject has to enter the following information

- Age
- Level of Education
- Background (Science, Engineering, Mathematics, Humanities, Business, Economics, Arts)

We want the above information to assess if people with specific characteristics, have a better ability in identifying unfair acts than others.

Regarding age, we want to see if young people tend to identify unfair acts easily than older people, regarding the education level, we want to see if more educated individuals can identify unfair acts better than other less educated individuals and regarding the background, we want to see if people that work with policies in their practice (Business and Economics) can identify unfair acts easier than people with other backgrounds.

Part 2: Questionnaire

You will be asked to read a small policy (of a few sentences) and then answer in sequence up to six questions based on the policy. Your answer will be in the form of choosing one of Agree, Disagree, Neither Agree nor Disagree together with an associated level of confidence in your chosen answer.

You will then see the answer given based on our methodology, and you will be asked to reconsider your answer: you can change the level of confidence in your previous answer or change your answer altogether.

This process will be repeated in the same way with four policies in total, each in a different policy context.

7.1.2 The Questionnaire

The questionnaire was carried out through Google Forms, which was open for submissions, for a week. In this section, we will state the policies that were in the questionnaire and the questions the participants were requested to answer.

For the registration part, these were the options for each personal information, where participants had to choose 1 option from:

- Age
 - 18 - 24
 - 25 - 34
 - 35 - 44
 - 45 - 54
 - 55 and over
 - I prefer not to say

- Level of Education
 - No schooling completed
 - High school graduate, diploma or the equivalent
 - Bachelor's degree
 - Master's degree
 - Doctorate degree
 - I prefer not to say

- Academic Background
 - Science
 - Engineering
 - Mathematics
 - Humanities
 - Business
 - Economics
 - Arts
 - I prefer not to say

After completing the above sections, the participants were then redirected to read the policies and answer to questions.

Here are the policies that were used in the questionnaire, along with the questions for each one of them. The participants had to select 1 of the options for each question.

- Policy 1: Car Insurance Policy

“The basic third-party car insurance plan costs 180 euros per year. If the driver is younger than 24 years old or older than 69 years old, then there is a 20% increase. Also, there is a 5% increase when the driver has been licensed for less than 2 years. Finally, if the customer lives in a high crime area, there is a 10% increase.”

Questions:

- You believe the policy is fair.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree
- You can explain why the policy is fair or not.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

Then, the evaluation from our methodology was shown, and then they had to answer to a few more questions based on the given evaluation.

Evaluation:

“The policy is discriminatory because

1) the premium is different for people younger than 24 year old or older than 69 year old, than people not in these age groups and

2) people that happen to live in a specific area, they get a higher premium.”

Question:

- Given the explanation based on our methodology on why the policy is fair or not, is it easier for you to explain why the policy is fair or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Policy 2: Mortgage Policy

Definition: A mortgage is a loan that the borrower uses to purchase or maintain a home or other form of real estate and agrees to pay back over time, typically in a series of regular payments.

“A customer is given a loan if the current and future financial status of the customer allows us (the bank) to give a loan without risking a lot. A customer will be given a loan if the loan they request is less than 70% of the total value of the property they want, they have at least 20% of the loan value and their occupation can support them with enough income to pay back the loan. Lastly, in case of pregnancy plans, the request will be considered more extensively.”

Questions:

- You believe the policy is fair.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You believe the policy respects the privacy of the costumers.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You can explain why the policy is fair or not.
 - I can easily explain why
 - I can explain why with some difficulties
 - I can not explain why at all

- You can explain why the policy does or does not respect the privacy of the costumers.
 - I can easily explain why
 - I can explain why with some difficulties
 - I can not explain why at all

Evaluation:

“The policy is discriminatory and unfair because the final decision is based on (sensitive) personal information and life choices of the customer e.g. pregnancy plans. A bank is expected to require only financial data.”

Questions:

- Given the explanation based on our methodology on why the policy is fair or not, is it easier for you to explain why the policy is fair or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Given the explanation based on our methodology on why the policy respects the privacy of the customers or not, is it easier for you to explain why the policy respects the privacy of the customers or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Policy 3.1: Credit Score Policy 1

Definition: A credit score is a number based on a level analysis of a person's credit files, to represent the creditworthiness of an individual. A credit score is primarily based on a credit report, information typically sourced from credit bureaus.

This helps lenders decide how likely you are to repay your debts and plays a significant role when securing a mortgage. Scores range from 300 to 850 points and are based on: Your payment history and ability to repay your debts on time. Late payments will lower your credit score.

“In the 1930s, government surveyors graded neighbourhoods in 239 cities, colour-coding them green for “best,” blue for “still desirable,” yellow for “definitely declining” and red for “hazardous.” The “redlined” areas were the ones local lenders discounted as credit risks, in large part because of the residents’ racial and ethnic demographics.”

This policy was banned 50 years ago.

Questions:

- You believe this banned policy still has an effect today on credit scores.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You can explain why the policy was fair or not.
 - I can easily explain why
 - I can explain why with some difficulties
 - I can not explain why at all

Evaluation:

“The policy was unfair because it was discriminating based on the racial and ethnic background of people. This policy still has a mark on the society because it defined, to an extent, the evolution of the redlined areas. People in the redlined areas were less economically supported and thus led to a not-as-good life, as other areas.”

Question:

- Given the explanation based on our methodology on why the policy is still unfair or not, is it easier for you to explain why the policy is still unfair or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Policy 3.2: Credit Score Policy 2

You are the CEO of a bank, and you requested from the Software Engineers to create a Credit Score Calculation system. This system is designed to decide which customers are eligible to get a loan through the website of your bank. The customer enters information such as: age, gender, marital status, occupation and income. Then, the system based on this information, it calculates the score, and if the score is over 500, the customer is eligible otherwise it is not.

Scenario

You are a customer, and you fill the form and your credit score, based on the system, is 494, and thus you are not allowed to get a loan. You contact the bank and ask if they can explain to you why you can't get a loan, and they tell you that they can't know how the system calculated the score, and they can't give you a loan if the system's output is less than 500.

Questions:

- You believe the policy is fair.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree
- You believe the policy respects the privacy of the costumers.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You believe that it is OK for non Software Engineer employees to not know how a system of their company works.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You believe that it is OK for a decision to be taken solely by a system.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- You can explain why the policy is fair or not.
 - I can easily explain why
 - I can explain why with some difficulties
 - I can not explain why at all

- You can explain why the policy does or does not respect the privacy of the customers.
 - I can easily explain why
 - I can explain why with some difficulties
 - I can not explain why at all

Evaluation:

“The policy is unfair, it violates the privacy of the customers, and it violates critical AI systems requirements. It is unfair because it might have different results for customers based on factors such as their gender, and it violates their privacy because it is required to provide sensitive information.

The fact that it might have different results for customers based on e.g. their age, the problem is that it MIGHT. Based on this policy, no-one can understand what led to a result and this points the need for Explainable AI. It should be possible for humans to track the decision-making path that an AI system followed to conclude somewhere.

Lastly, a system should support decisions and not take decisions. The system should not be defining who gets a loan or not, but it should be used by employees to get a rough estimate and if needed, get in more detail before deciding. In our case, the customer had a credit score of 494. If it was a human assessing our info, our credit score could have been enough to get a loan.”

Questions:

- Given the explanation based on our methodology on why the policy is fair or not, is it easier for you to explain why the policy is fair or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Given the explanation based on our methodology on why the policy respects the privacy of the customers or not, is it easier for you to explain why the policy respects the privacy of the customers or not?
 - Much Easier
 - Easier
 - No Effect
 - Not Easier
 - Not Easier at All

- Given the explanation based on our methodology on why the system is problematic, you can now understand why it is not OK for stakeholders (users, employees etc.) of a system to not know how it works.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

- Given the explanation based on our methodology on why the system is problematic, you can now understand why it is not OK for a decision to be solely depended on a system.
 - Strongly Agree
 - Agree
 - Neither Agree nor Disagree
 - Disagree
 - Strongly Disagree

This was the complete questionnaire that we designed to evaluate our methodology. In the next section, we will present our results and what we extracted from the responses.

7.1.3 Results

After the questionnaire was public for a week, we got 12 responses in total. This is not the response we were anticipating, and the results might not be representative, but we will represent the outcome based on the responses we got.

From the demographic questions of the questionnaire, the majority of respondents have a Science background with a Bachelor's or a Master's degree. The age groups are split in half between 18-24 and 25-34 years old. We don't mind that there was no diversity regarding the age and level of education as much as we would prefer a wide spectrum of academic backgrounds.

The goal of the questionnaire was to see if the respondents could identify unfair or privacy violating policies, more easily as they were shown the answers based on our evaluation method. Regarding the questions if given an explanation based on our methodology had an effect or not on the respondent's perspective, we expect most of the answers to be positive (“Easier” and “Much Easier”).

Due to the minimal responses, along with some structural errors of the questionnaire, we can not claim that we got the ideal results or not. Thus, we won’t be able to explicitly support our statement mentioned in the previous paragraph. However, there is one noteworthy result.

In most cases, the explanations based on our methodology made it easier for the participants to explain a certain situation. They were asked if they can explain a situation before being shown our explanation and then asked again if they can explain the same situation easier or not. In the figure below, the percentages cannot be seen but based on the colouring we can see that in most cases the red colour holds the highest percentage, which represents the option “Easier”; for the participant it was easier to either explain a situation after being shown the explanation based on our methodology.

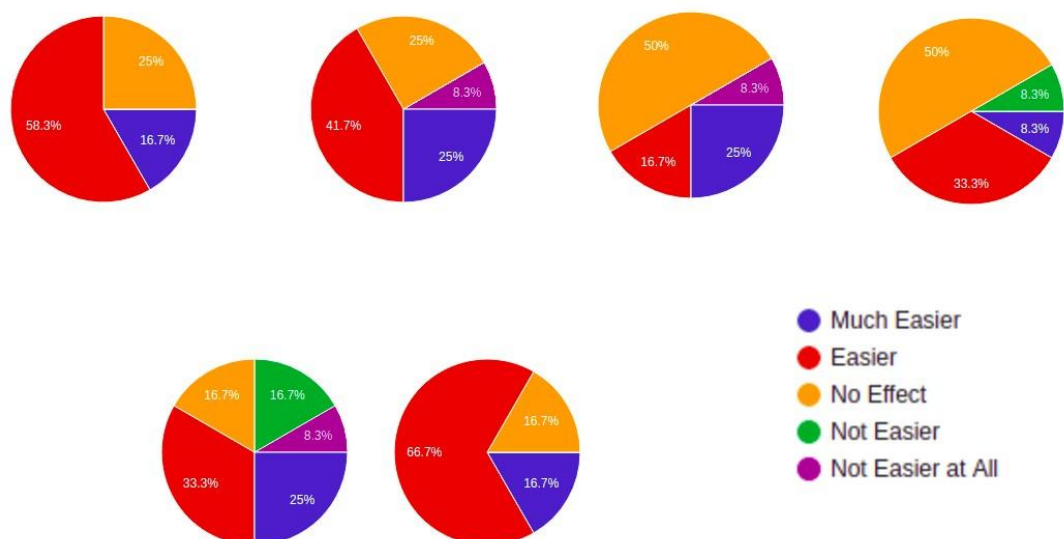


Figure 7.1: Easier to explain after shown an explanation based on our methodology

Chapter 8

Conclusions

8.1 Summary	66
8.2 Future Work	67

8.1 Summary

This thesis aimed to provide a framework that helps the ethical assessment of policies which AI systems are/will be designed upon. By proposing and developing such framework, we state the importance of ethics in AI and how crucial the design phase of systems is. Furthermore, we showed how we can use Argumentation to make assessments on ethical matters and abstract values in general.

Based on the requirements of the EU Trustworthy AI guidelines [8], we developed a framework that assesses policies at an agnostic level (i.e. not focusing on a specific type of policies) taking in mind 5 parameters: age, gender, nationality, income and occupation. We assess policies that have these parameters and test them against various values, and we check if there is an unfair behaviour. This process is done by the “Check Policy” tool of our web application. This tool covers the Diversity, non-discrimination and fairness requirement of the guidelines.

Additionally, for the requirements about Privacy and Data Governance, Transparency, Societal and Environmental Well-being and Accountability, we created a meta-level questionnaire for the policymakers, with which we pose a statement and depending on if the policymaker selects “Yes” or “No”, that interprets to if the processes of a company when forming a policy are for or against the statement. In case there is an option that it is against a statement, the tool will provide information as to what can be changed or followed to be able to adhere to the specified requirement proposed by the statement. This process is being done by the “Policymaker Questionnaire” tool of our web application.

Furthermore, it is important to mention that another important aspect of this thesis is the explainable nature of our tools. Using Argumentation, we achieved explainable tools and evidently from the evaluation, our explanations are effective and help users understand a problem more easily after they were shown an explanation based on our methodology.

Lastly, our approach can be applied to any AI problem, let that be autonomous systems or gaming bots. Depending on the world of the problem we can re-evaluate the preference between our values as to reflect the most important values in the given scenario e.g. in one scenario fairness might be more important than transparency.

8.2 Future Work

Given the current done work, there are a few objectives that can be done as to add more value to this research. We can add more parameters to check a policy upon (i.e. not perform checks at an agnostic level), add more resources regarding the policymaker questionnaire, perform a more comprehensive evaluation and develop a similar framework to be adapted on actual AI systems.

Work with experts from various backgrounds can be done as to add tailored parameters to the policy assessment tool to be able to check in depth more specific policies. For example, experts from the banking sector could assist the development by providing us more information of various banking policies and add more parameters regarding such policies. By adding these extra parameters, our tool can be used in a broader aspect and check if there is an unfair behaviour for more specific policies, in this case banking policies.

Currently, when a policymaker selects an option against a statement in the “Policymaker Questionnaire” tool, our response includes limited sources which the user can visit and learn more about what changes s/he can be made to be able to adhere to a specific statement. With furthermore research, more sources can be added as to provide various options to the user.

Our evaluation felt short, and we were not able to support all of our expectations. We were able to support that our explanations helped the users, but we were not able to

evidently support that we had an effect on the participant's perspective between the policies. With a more comprehensive questionnaire, that has a better structure between the objectives, we can evaluate the missing expectation.

At the time being, our framework currently supports the assessment of policies only. However, the same idea can be adapted as to, on-line, assess AI systems. This was proposed by Aler Tubella, A. et al. , in Governance by Glass-Box [12], which we can implement using Argumentation, and more specific via the GORGIAS argumentative framework [14].

Lastly, our methodology, as mentioned before, can be expanded and create a tool that assesses, on-line, AI systems. This can be used as an additional component to Machine Learning black-boxes as to provide Explainability and furthermore make sure that the decisions taken do not violate any values and requirements. This is crucial as with our plug-in system, we could detect in early-stage, before reaching the user, any discriminatory, unfair and in general unsafe decisions.

Bibliography

- [1] Stanford Encyclopedia of Philosophy. (2018, June 15). Aristotle's Ethics (Stanford Encyclopedia of Philosophy).
<https://plato.stanford.edu/entries/aristotle-ethics/#HumaGoodFuncArgu>
- [2] IBM. (n.d.). AI Ethics. <https://www.ibm.com/artificial-intelligence/ethics>
- [3] Microsoft. (n.d.). Responsible AI principles.
<https://www.microsoft.com/en-us/ai/responsible-ai>
- [4] Singapore Infocomm Media Development Authority. (2018, August 30). Composition of the Advisory Council on the Ethical Use of Artificial Intelligence ("AI") and Data. Infocomm Media Development Authority.
<https://www.imda.gov.sg/news-and-events/Media-Room/Media-Releases/2018/composition-of-the-advisory-council-on-the-ethical-use-of-ai-and-data>
- [5] European Commission. (n.d.-b). Denmark AI Strategy Report | Knowledge for policy.
https://knowledge4policy.ec.europa.eu/ai-watch/denmark-ai-strategy-report_en
- [6] Lapach, Y. (2021, April 21). Denmark introduces mandatory legislation for AI and Data Ethics. 2021.AI.
<https://2021.ai/denmark-introduces-mandatory-legislation-ai-data-ethics/>
- [7] IEEE Standards Association (IEEE SA). (2020, October 16). IEEE 7000™ Projects | IEEE Ethics In Action in A/IS - IEEE SA. Ethics In Action | Ethically Aligned Design.
<https://ethicsinaction.ieee.org/p7000/>
- [8] European Commission. (n.d.-a). Assessment List for Trustworthy Artificial Intelligence (ALTAI) for self-assessment | Shaping Europe's digital future.
<https://digital-strategy.ec.europa.eu/en/library/assessment-list-trustworthy-artificial-intelligence-altai-self-assessment>
- [9] A. (2018, November 1). Watchdog slams Lufthansa over 'algorithm' price hikes. Aviation – Gulf News.
<https://gulfnews.com/business/aviation/watchdog-slams-lufthansa-over-algorithm-price-hikes-1.2148603>
- [10] European Commission. (n.d.-c). *Proposal for a Regulation laying down harmonised rules on artificial intelligence* | Shaping Europe's digital future.
<https://digital-strategy.ec.europa.eu/en/library/proposal-regulation-laying-down-harmonised-rules-artificial-intelligence>

- [11] Theodorou, A. (2020). Why Artificial Intelligence is a Matter of Design. *Artificial Intelligence*, 105–131. https://doi.org/10.30965/9783957437488_009
- [12] Aler Tubella, A., Theodorou, A., Dignum, V., & Dignum, F. (2019). Governance by glass-box: Implementing transparent moral bounds for AI behaviour. *arXiv preprint arXiv:1905.04994*.
- [13] Wikipedia contributors. (2021, March 4). Argumentation theory. Wikipedia. https://en.wikipedia.org/wiki/Argumentation_theory
- [14] Kakas, A. C., Moraitis, P., & Spanoudakis, N. I. (2018). GORGIAS: Applying argumentation. *Argument & Computation*, 10(1), 55–81. <https://doi.org/10.3233/aac-181006>

Appendix A

This Appendix contains the fully formalized version of the coffee policy we have defined in section 3.2.2 GORGIAS: Applying argumentation.

```
% Policy
% "In general, I like to drink coffee.
% In the morning, I like to drink a cup
% of cappuccino. After I have my lunch,
% I like to have a shot of espresso
% and at night I do not drink coffee."

% Path to GORGIAS
:- compile('gorgias/lib/gorgias').
:- compile('gorgias/ext/lpwnf').

% Options: drink(cappuccino), drink(espresso), drink(french),
drink(none)
% Need to complement all possible drink options.

complement(drink(cappuccino),drink(espresso)).
complement(drink(cappuccino),drink(french)).
complement(drink(cappuccino),drink(none)).

complement(drink(espresso),drink(cappuccino)).
complement(drink(espresso),drink(french)).
complement(drink(espresso),drink(none)).

complement(drink(french),drink(cappuccino)).
complement(drink(french),drink(espresso)).
complement(drink(french),drink(none)).

complement(drink(none),drink(cappuccino)).
complement(drink(none),drink(espresso)).
complement(drink(none),drink(french)).

% Basic rules
rule(r1(cappuccino),drink(cappuccino),[]).
rule(r2(espresso),drink(espresso),[]).
rule(r3(french),drink(french),[]).
rule(r4(none),drink(none),[]).

% Universal case: Drink coffee
rule(pr1(cappuccino),prefer(r1(cappuccino),r4(none)),[]).
rule(pr2(espresso),prefer(r2(espresso),r4(none)),[]).
```

```

rule(pr3(french),prefer(r3(french),r4(none)),[]).

% Morning cappuccino
rule(pr4(cappuccino),prefer(r1(cappuccino),r2(espresso)),[morning]).
rule(pr5(cappuccino),prefer(r1(cappuccino),r3(french)),[morning]).
rule(pr6(cappuccino),prefer(r1(cappuccino),r4(none)),[morning]).

% After lunch espresso
rule(pr7(espresso),prefer(r2(espresso),r1(cappuccino)),[after_lunch])
.
rule(pr8(espresso),prefer(r2(espresso),r3(french)),[after_lunch]).
rule(pr9(espresso),prefer(r2(espresso),r4(none)),[after_lunch]).

% Night none
rule(pr10(none),prefer(r4(none),r1(cappuccino)),[night]).
rule(pr11(none),prefer(r4(none),r2(espresso)),[night]).
rule(pr12(none),prefer(r4(none),r3(french)),[night]).

rule(c1(none),prefer(pr10(none),pr1(cappuccino)),[]).
rule(c2(none),prefer(pr11(none),pr2(espresso)),[]).
rule(c3(none),prefer(pr12(none),pr3(french)),[]).

% Knowledge
% rule(f1,morning,[]).
% rule(f2,after_lunch,[]).
% rule(f3,night,[]).

```

Appendix B

This Appendix contains the fully formalized version of the policy we have defined in section 4.1.1.1 How-to Create Your Input File.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% POLICY %%%%%%%%%%
% "The basic third-party car insurance plan costs 180 euros per
% year.
% If the driver is younger than 24 years old or older than 69
% years old, then there is a 20% increase.
% Also, there is a 5% increase when the driver has been
% licensed for less than 2 years.
% Finally, if the customer lives in a high crime area, there is
% a 10% increase."
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
:- compile('src/gorgias/lib/gorgias').
:- compile('src/gorgias/ext/lpwnf').
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Complements %%%%%%%%%%
% We need to add the following lines to show that at each
% decision, only one of the predicate options can hold.
```

```
complement(premium_cost(low),premium_cost(high)).
complement(premium_cost(low),premium_cost(medium)).
complement(premium_cost(medium),premium_cost(high)).
complement(premium_cost(medium),premium_cost(low)).
complement(premium_cost(high),premium_cost(medium)).
complement(premium_cost(high),premium_cost(low)).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Basic Rules %%%%%%%%%%
```

```
rule(r1(),premium_cost(low),[]).
rule(r2(),premium_cost(medium),[]).
rule(r3(),premium_cost(high),[]).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Default Scenario %%%%%%%%%%
```

```
% SBP0 = <S0 = {basic_third_party_plan},
% O0 = {premium_cost(low)}>
rule(pr1(),prefer(r1(),r2()),[basic_third_party_plan]).
```

```

rule(pr2(),prefer(r1(),r3()),[basic_third_party_plan]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Rest Scenarios by SBP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% SBP1 = <S1 = {basic_third_party_plan,
% driver_younger_than_24}, 01 = {premium_cost(high)}>

rule(pr3(),prefer(r3(),r1()),[basic_third_party_plan,
driver_younger_than_24]).
rule(c1(),prefer(pr3(),pr2()),[]).

% SBP2 = <S2 = {basic_third_party_plan, driver_older_than_69},
% 02 = {premium_cost(high)}>

rule(pr4(),prefer(r3(),r1()),[basic_third_party_plan,
driver_older_than_69]).
rule(c2(),prefer(pr4(),pr2()),[]).

% SBP3 = <S3 = {basic_third_party_plan, licenced_less_than_2},
% 03 = {premium_cost(medium)}>

rule(pr5(),prefer(r2(),r1()),[basic_third_party_plan,
licenced_less_than_2]).
rule(c3(),prefer(pr5(),pr1()),[]).

% SBP4 = <S4 = {basic_third_party_plan, high_crime_area},
% 04 = {premium_cost(medium)}>

rule(pr6(),prefer(r2(),r1()),[basic_third_party_plan,
high_crime_area]).
rule(c4(),prefer(pr6(),pr1()),[]).

% SBP5 = <S5 = {basic_third_party_plan, high_crime_area,
% driver_younger_than_24}, 05 = {premium_cost(high)}>

rule(pr7(),prefer(r3(),r1()),[basic_third_party_plan,
high_crime_area, driver_younger_than_24]).
rule(c5(),prefer(pr7(),pr6()),[]).
rule(c6(),prefer(pr7(),pr2()),[]).
rule(d1(),prefer(c5(),c4()),[]).

```

```
%%%%%%%%%%%% Knowledge %%%%%%%%%%%%%%  
% You can have this knowledge section  
% if you want to try your policy but it  
% is not required by the tool.  
% rule(f1,basic_third_party_plan,[]).  
% rule(f2,driver_younger_than_24,[]).  
% rule(f3,driver_older_than_69,[]).  
% rule(f4,licenced_less_than_2,[]).  
% rule(f5,high_crime_area,[]).
```

Appendix C

This Appendix contains the whole argumentation PROLOG file of our system that was mentioned in section 4.1.3.3 Processing the Inputs.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
:- compile('src/gorgias/lib/gorgias').
:- compile('src/gorgias/ext/lpwnf').
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

complement(promotes(Value),demotes(Value)).
complement(demotes(Value),promotes(Value)).

abducible(facts_available, []).
abducible(neg(facts_available), []).
abducible(explanation_available, []).
abducible(neg(explanation_available), []).

abducible(does_not_depend_on_age, []).
abducible(neg(does_not_depend_on_age), []).
abducible(does_not_depend_on_nationality, []).
abducible(neg(does_not_depend_on_nationality), []).
abducible(does_not_depend_on_gender, []).
abducible(neg(does_not_depend_on_gender), []).

abducible(does_not_depend_on_income, []).
abducible(neg(does_not_depend_on_income), []).
abducible(does_not_depend_on_occupation, []).
abducible(neg(does_not_depend_on_occupation), []).
abducible(does_not_depend_on_marital_status, []).
abducible(neg(does_not_depend_on_marital_status), []).

abducible(respects_privacy, []).
abducible(neg(respects_privacy), []).
abducible(established_data_governance, []).
abducible(neg(established_data_governance), []).
abducible(thought_about_societal_well_being, []).
abducible(neg(thought_about_societal_well_being), []).
abducible(thought_about_environmental_well_being, []).
abducible(neg(thought_about_environmental_well_being), []).
abducible(accountability, []).
```

```
abducible(neg(accountability), []).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
rule(r1(Value),promotes(Value),[]).
```

```
rule(r2(Value),demotes(Value),[]).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%                               %
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
rule(pr1(Value),prefer(r2(Value),r1(Value)),[]).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%                               %
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Traceability
```

```
rule(pr2(traceability),prefer(r1(traceability),r2(traceability)
),[facts_available]).
```

```
rule(c1(traceability),prefer(pr2(traceability),pr1(traceability
)),[]).
```

```
% Explainability
```

```
rule(pr3(explainability),prefer(r1(explainability),r2(explainab
ility)),[explanation_available]).
```

```
rule(c2(explainability),prefer(pr3(explainability),pr1(explaina
bility)),[]).
```

```
% Transparency
```

```
rule(pr4(transparency),prefer(r1(transparency),r2(transparency)
),[facts_available,explanation_available]).
```

```
rule(c3(transparency),prefer(pr4(transparency),pr1(transparency
)),[]).
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%
```

```
%                               Diversity, Non-discrimination & Fairness
```

```
%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```

% Diversity & Non-discrimination
rule(pr5( diversity_non_discrimination_age),prefer(r1( diversity_
non_discrimination_age),r2( diversity_non_discrimination_age)),[
does_not_depend_on_age]).
rule(c4( diversity_non_discrimination_age),prefer(pr5( diversity_
non_discrimination_age),pr1( diversity_non_discrimination_age)),
[]).

rule(pr6( diversity_non_discrimination_nationality),prefer(r1(di
versity_non_discrimination_nationality),r2( diversity_non_discri
mination_nationality)),[does_not_depend_on_nationality]).
rule(c5( diversity_non_discrimination_nationality),prefer(pr6(di
versity_non_discrimination_nationality),pr1( diversity_non_discr
imation_nationality)),[]).

rule(pr7( diversity_non_discrimination_gender),prefer(r1( diversi
ty_non_discrimination_gender),r2( diversity_non_discrimination_g
ender)),[does_not_depend_on_gender]).
rule(c6( diversity_non_discrimination_gender),prefer(pr7( diversi
ty_non_discrimination_gender),pr1( diversity_non_discrimination_
gender)),[]).

rule(pr8( diversity_non_discrimination),prefer(r1( diversity_non_
discrimination),r2( diversity_non_discrimination)),[does_not_dep
end_on_gender,does_not_depend_on_nationality,does_not_depend_on
_age]).
rule(c7( diversity_non_discrimination),prefer(pr8( diversity_non_
discrimination),pr1( diversity_non_discrimination)),[]).

% Fairness
rule(pr9( fairness_income),prefer(r1( fairness_income),r2( fairnes
s_income)),[does_not_depend_on_income]).
rule(c8( fairness_income),prefer(pr9( fairness_income),pr1( fairne
ss_income)),[]).

rule(pr10( fairness_occupation),prefer(r1( fairness_occupation),r
2( fairness_occupation)),[does_not_depend_on_occupation]).
rule(c9( fairness_occupation),prefer(pr10( fairness_occupation),p
r1( fairness_occupation)),[]).

rule(pr19( fairness_marital_status),prefer(r1( fairness_marital_s

```

```
tatus),r2(fairness_marital_status)),[does_not_depend_on_marital_status]).
```

```
rule(c18(fairness_marital_status),prefer(pr19(fairness_marital_status),pr1(fairness_marital_status)),[]).
```

```
rule(pr11(fairness),prefer(r1(fairness),r2(fairness)),[does_not_depend_on_income,does_not_depend_on_occupation,does_not_depend_on_marital_status]).
```

```
rule(c10(fairness),prefer(pr11(fairness),pr1(fairness)),[]).
```

```
% Privacy and Data Governance
```

```
rule(pr12(privacy),prefer(r1(privacy),r2(privacy)),[respects_privacy]).
```

```
rule(c11(privacy),prefer(pr12(privacy),pr1(privacy)),[]).
```

```
rule(pr13(data_governance),prefer(r1(data_governance),r2(data_governance)),[established_data_governance]).
```

```
rule(c12(data_governance),prefer(pr13(data_governance),pr1(data_governance)),[]).
```

```
rule(pr14(privacy_data_governance),prefer(r1(privacy_data_governance),r2(privacy_data_governance)),[respects_privacy,established_data_governance]).
```

```
rule(c13(privacy_data_governance),prefer(pr14(privacy_data_governance),pr1(privacy_data_governance)),[]).
```

```
% Societal and Environmental well being
```

```
rule(pr15(societal_well_being),prefer(r1(societal_well_being),r2(societal_well_being)),[thought_about_societal_well_being]).
```

```
rule(c14(societal_well_being),prefer(pr15(societal_well_being),pr1(societal_well_being)),[]).
```

```
rule(pr16(environmental_well_being),prefer(r1(environmental_well_being),r2(environmental_well_being)),[thought_about_environmental_well_being]).
```

```
rule(c15(environmental_well_being),prefer(pr16(environmental_well_being),pr1(environmental_well_being)),[]).
```

```
rule(pr17(environmental_societal_well_being),prefer(r1(environmental_societal_well_being),r2(environmental_societal_well_being)),[thought_about_environmental_well_being,thought_about_societal_well_being]).
```

```

al_well_being])).
rule(c16(environmental_societal_well_being),prefer(pr17(environmental_societal_well_being),pr1(environmental_societal_well_being)),[]).

```

```

% Accountability

```

```

rule(pr18(accountability),prefer(r1(accountability),r2(accountability)),[accountability]).
rule(c17(accountability),prefer(pr18(accountability),pr1(accountability)),[]).

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

% Knowledge %

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

% rule(f1,facts_available,[]).
% rule(f2,explanation_available,[]).
% rule(f3,does_not_depend_on_age,[]).
% rule(f4,does_not_depend_on_nationality,[]).
% rule(f5,does_not_depend_on_gender,[]).
% rule(f6,does_not_depend_on_income,[]).
% rule(f7,does_not_depend_on_marital_status,[]).
% rule(f8,does_not_depend_on_occupation,[]).
% rule(f9,respects_privacy,[]).
% rule(f10,established_data_governance,[]).
% rule(f11,thought_about_societal_well_being,[]).
% rule(f12,thought_about_environmental_well_being,[]).
% rule(f13,accountability,[]).

```