POWVER

Technical Report 2019-14

Title: **Explainability as a Non-Functional Requirement**

Author: Maximilian A. Köhl, Dimitri Bohlender, Kevin Baum, Markus Langer, Daniel Oster, Timo Speith

Report Number: 2019-14

ERC Project: Power to the People. Verified.

ERC Project ID: 695614

Funded Under: H2020-EU.1.1. – EXCELLENT SCIENCE

Host Institution: Universität des Saarlandes, Dependable Systems and Software
Saarland Informatics Campus

Published In: IEEE 27th International Requirements Engineering Conference

Please cite this publication as follows:

Maximilian A. Köhl, Dimitri Bohlender, Kevin Baum, Markus Langer, Daniel Oster, Timo Speith.
Explainability as a Non-Functional Requirement.
27th IEEE International Requirements Engineering Conference, RE 2019, Jeju Island, Korea (South), September 23-27, 2019. IEEE 2019, ISBN 978-1-7281-3912-8: 363-368.

Explainability as a Non-Functional Requirement

Maximilian A. Köhl[*], Dimitri Bohlender[†], Kevin Baum[*], Markus Langer[*], Daniel Oster[*] and Timo Speith[*]
[*]Saarland University, Saarbrücken, Germany
Email: mkoehl@cs.uni-saarland.de, {kevin.baum, markus.langer, daniel.oster, timo.speith}@uni-saarland.de
[†]RWTH Aachen University, Aachen, Germany
Email: bohlender@embedded.rwth-aachen.de

***Abstract*—Recent research efforts strive to aid in designing *explainable systems*. Nevertheless, a *systematic* and *overarching* approach to ensure explainability by design is still missing. Often it is not even clear what precisely is meant when demanding explainability. To address this challenge, we investigate the elicitation, specification, and verification of *explainablity* as a *Non-Functional Requirement* (NFR) with the long-term vision of establishing a standardized certification process for the explainability of software-driven systems in tandem with appropriate development techniques.**

In this work, we carve out different notions of explainability and high-level requirements people have in mind when demanding explainability, and sketch how explainability concerns may be approached in a hypothetical hiring scenario. We provide a conceptual analysis which unifies the different notions of explainability and the corresponding explainability demands.

***Index Terms*—explainable systems, requirements specification, requirements elicitation, terminology, certified explainability**

I. Introduction

The desire to sufficiently understand the systems we interact with is natural. If a person acts in an unexpected way, for instance comes late to an important appointment, we might ask for an explanation and be content with hearing about a traffic jam. In contrast, software-driven systems are becoming more and more opaque due to their ever increasing complexity and autonomy. Sometimes even domain experts and system engineers struggle to understand certain aspects of a system [1]. Systems with machine-learning based components in particular become hard to understand [2]. This development results in an increasing interest in *explainable systems*.

Explanations enable understanding and thereby foster trust and trustworthiness, justify actions and decisions, improve usability, help in locating sources of error, and can minimize the chance for human error. Particularly in “human in the loop” scenarios, in which humans have to make a decision based on a system’s recommendation, humans cannot reach an informed decision without having access to the system’s reasons for its recommendation. The assessment of a system’s allegedly erroneous behavior via an adequate explanation could resolve questions of responsibility and liability, e.g., whether the design was faulty and the manufacturer is to blame, or whether someone else is responsible.

A lack of explainability, on the other hand, not only gives rise to various moral, social, and legal problems [3], [4]. It further fuels distrust [5], diminishes user acceptance and satisfaction [6], and inhibits the adoption of new technologies. These problems have also been identified by legislators. The European Union, for instance, debated about a general *Right to Explanation* [7] which is partly enshrined in certain regulations [8]. Furthermore, the EU High-Level Expert Group on AI proposed “Ethics Guidelines for Trustworthy AI,” in which they promote explainability as a crucial means for building trust in the decisions of software-driven systems [9].

Consequently, explainability needs to be taken into account during development in order to improve the quality of the target artifact and meet various regulatory requirements. Appropriate development techniques need to be established that guarantee a certain degree of explainability. Design decisions that impact the explainability of a system must not be taken by the developer implicitly, but explicitly specified as part of the design process. However, it is often unclear what precisely is meant when demanding explainability, and how it can be achieved by design. To the best of our knowledge there is no systematic and overarching approach to the explicit specification of explainability requirements, on how to take them into account during development, and how to evaluate whether an artifact indeed meets those requirements. This paper aims to be a starting point to address these issues.

A. Contribution

We begin this paper with a brief discussion of research in the field of explainable artificial intelligence (XAI) and explainable systems in general. In Section II, we carve out different notions of explainability and high-level requirements people have in mind when demanding explainability. Further, we sketch how explainability concerns may be approached in a hypothetical hiring scenario in Section III. Based on the insights gained, we provide a conceptual analysis in Section IV, unifying the different notions of explainability and the corresponding explainability demands. The resulting notion provides a starting point for a systematic and overarching approach to explainability requirements. In Section V, we conclude by sketching our long-term vision of the establishment of a standardized certification process for explainability in tandem with appropriate development techniques.

II. Charting the Field

The demand for explainable systems is obvious in recent and ongoing research. A prominent example is the DARPA-funded *Explainable Artificial Intelligence* research project [1]. It acknowledges the lack of insight and knowledge gain in the context of current machine-learning based systems, and aims to investigate (1) "how to produce more explainable models," (2) "how to design the explanation interface," and (3) "how to understand the psychological requirements for effective explanations." Within the field of machine learning, the terminology surrounding explainability is neither uniform nor consistent [10]. Terms like "interpretability" [11], "scrutability" [12], and "explainable artificial intelligence" [13] target roughly the same endeavor, i.a. making the inner workings of systems more accessible and the outputs such as predictions or recommendations assessable. However, it is not clear what precisely is meant by "inner workings" or by "making accessible." Even more pressing is that the same terms, e.g., "interpretability," in different papers may refer to distinct notions. For instance, Lipton [2] mentions three different kinds of system transparency which alternately constitute interpretability.

Existing approaches aim at making given and usually non-interpretable systems explainable [14], [15]. System engineers attempt to generate explanations via feature importance [16] or explanation vectors [17]. Regarding document classification, Martens and Provost [18] propose linguistic explanations with bag-of-words features to make system recommendations more assessable for domain, e.g., legal or personnel, experts. To increase the transparency with respect to end users, comprehensible local approximations [19], counterfactuals [7], and contrastive explanations [20] come into play.

Research on explainable systems is carried out in pursuit of various goals, reflected in the different meanings attributed to "explainability." There are roughly two categories of research: (A) research on how to adapt machine learning and other techniques to allow for a more thorough inspection and understanding for engineers who build such systems, and (B) research on how to enable users of such systems to understand them in relevant aspects. Here, deeper inspection and understanding of the system's behavior for engineers is a prerequisite for making those systems understandable to users. Conflating those categories, explainability is concerned with enabling human understanding of various aspects of software-driven systems. In line with this, it is not obvious what policymakers or other stakeholders actually mean when they demand explainability and enshrine it in laws or guidelines. What shall be explainable to whom and how should it be evaluated whether an artifact indeed meets those requirements? These observations suggest that different target groups need different, context-sensitive explanations to be able to understand the relevant aspects of a particular system.

Studying different strains of research we find that, while different techniques for implementing explainability emerge, the concept itself, and in which context which techniques are appropriate, remains under-specified. All in all, there are many accounts with varying and partially overlapping goals. Despite the demand for explainability, there is no overarching consensus about what "explainability" means. Hence, a unified notion of explainability is needed. To take explainability into account during development, it needs to be specified more precisely, and knowledge about which techniques to apply in which case needs to be systematically collected.

III. Case Study: Automated Hiring System

Let us now turn to a concrete example where explainability is required. Imagine the following scenario: a large organization tries to improve the efficiency of their hiring processes. In a meeting between the executive management, hiring managers, and employee representatives, they decide to task the IT department with developing and implementing a software system for trainee hiring [21]. In a first meeting with the IT department, the following functional requirements are identified: applicants shall apply through an online application system where they upload their CV. The system shall then automatically screen the applicants' CVs, and provide the hiring managers with a ranking of applicants based on their estimated fit for a given position. The hiring managers can use this ranking as an additional source of information to screen the most promising candidates and, afterwards, decide which applicants proceed to the next stage of the hiring process. Based on this decision, applicants either receive a rejection letter or an invitation to the next stage of the selection process.

Among other requirements it is demanded that the system's decisions, i.e., the ranking, "shall be explainable to the various stakeholders." These are at least: applicants, hiring managers, the executive management, employee representatives, the legal department, and the engineers of the hiring system themselves [22]. All of these groups possess different background knowledge about the system and the hiring process, as well as different motivations within the hiring process. For instance, applicants only know that they upload their CV and want a fair hiring process [23]. The executive management wants a lean and effective process [24]. Employee representatives and the legal department may want to know on which features the system bases its ranking, as they want to have an unbiased selection process in order to prevent lawsuits [25]. To reach an informed decision based on the system's ranking, the hiring managers demand reasons for why the systems ranks applicants as it does. The mere ranking is not informative enough and insufficient for them to come to a justified decision.

In a meeting of the IT department a need for the following design choices becomes apparent: Roughly, the system could be implemented based on machine learning using existing data, by explicitly programming various criteria into the system, or by a combination of both. A purely machine-learning based approach using existing data will most likely introduce unfair biases and make the system hard to explain and reason about. On the contrary, a system using explicitly specified criteria will probably not perform as well and cost more, especially because the criteria need to be developed, and characteristics of a "good" applicant remain undefined without a solid job analysis. However, its reasoning would be explicit and easily explainable. With regard to the explainability demand of the hiring managers and the legal department, it is clear that some reasons need to be provided for why a certain ranking was produced.

At this stage, more precise requirements need to be elicited. What needs to be explainable to whom and what qualifies as an explanation of what? Given precise and assessable explainability requirements, system engineers could explore the design space and determine appropriate development techniques in a more systematic and substantial way. For example, finding the right balance between machine-learning based components and explicit criteria such that the overall system becomes sufficiently explainable by design.

IV. Explainability Requirements

We start with a conceptual analysis of *explainable systems* as an important first step towards a systematic and overarching approach for the elicitation and specification of explainability requirements. Intuitively, what makes certain aspects of a system explainable to the relevant stakeholders is access of the stakeholders to some kind of *explanation* for their aspects of interest. However, what is an explanation?

A. What is an Explanation?

Looking into literature one finds a broad variety of different approaches on how to spell out the concept of explanation. On the one side, there are rather technical notions of explanation [26], [14], [19] which are usually linked to causes. On the other side, there are more pragmatic notions which regard explanations as answers to certain questions, in particular "Why questions" [27], [28], [29]. Both approaches, however, do not exclude each other. An answer to a question can be an explanation precisely because it has the structure and qualities identified by technical accounts.

The need for an explanation originates in a lack of understanding of some phenomenon X, called *explanandum* in the philosophy of science [30]—however, not as a whole but with respect to some *aspect* Y of interest. Intuitively, Y encodes a question one may sensibly ask about X. For example, when applicant A is ranked higher than applicant B, one may ask the question: which qualities of A make A a better fit for the job than B? Here, the explanandum X is the ranking produced by the system. When eliciting explainability requirements, it is crucial to precisely capture X and Y by interviewing the stakeholders. Questions like "Why does applicant A rank higher than applicant B?" are ill-posed in that they are highly ambiguous. This question, for instance, can be answered simply by pointing out that A ranks higher than B because, according to the system, A is a better fit for the job. However, such an explanation will not be of much help for hiring managers.

For our purposes, we need a notion of explanation that targets a certain *kind* of stakeholder—an explanation for an engineer may not explain anything to a user. That is, we need a notion that enables generalization and abstracts from concrete individuals. Of course, referring to groups introduces imprecision as it is rarely possible to specify precise characteristics and skills of a certain group [31]. Nevertheless, it enables generalization and it is in fact a common technique to assume that users with specific skills interact with a system [32]. For our purposes, we aim to be able to express that something needs to be explainable to a particular group, viz. the *target group* G of an explanation. A characterization of the concept of explanation which does not generalize and abstract from concrete individuals will not be very useful.

Still, a target group G may contain single agents who lack the required abilities or knowledge. To avoid such corner cases causing an explanation E to not qualify as such, even though it explains the aspect of interest to a significant part of G, we only require all *representatives* R of G to be content with the provided explanations. We presuppose that such representatives are equipped with the background knowledge and processing capabilities characteristic of the target group.

Furthermore, the *context* in which an explanation is provided matters. First, it does not only affect what needs to be explained. For example, before the hiring process, applicants might want to know which kind of information will be evaluated by the system and be interested in how to improve for their next selection process [33]. Second, different contexts may place constraints on the explanation generation process or the form of acceptable explanations. For example, to enable hiring managers to discuss the results in the context of a meeting, an explanation might have to fit on a single screen but still provide sufficient detail, or potentially be queryable at a reasonable latency to not hinder productivity.

Therefore, while some aspect Y may have an explanation that achieves the maximal depth of understanding, it might not be the best explanation in all contexts. In particular, if a context requires the explanation to be given in aural form, neither a detailed textual explanation nor a succinct visualization will suffice. Instead, for each context, an explanation must be found that maximizes the depth of understanding within the context's constraints.

The notion we propose in the following is both *target-aware* and *context-aware*. What counts as an explanation of what for whom depends on the intended target group G, i.e., a certain kind of stakeholder, and the explanatory *context* C. Following insights of Achinstein [34] and Van Fraassen [35], we propose a pragmatic notion of "explanation for" in terms of understanding:

Definition 1 (Explanation For): E is an *explanation* of explanandum X with respect to aspect Y for target group G, in context C, if and only if the processing of E in context C by any representative[1] R of G makes R understand X with respect to Y.

Analyzing *explanation* in terms of *understanding* may not seem illuminating at first—however, as we argue, it is illuminating as it enables leveraging results from psychology and the cognitive sciences to assess whether something is really an explanation and how people react to different kinds of explanations [36], [37]. In particular, tying explainability to understanding eventually enables verification through studies conducted with the relevant stakeholders.

Note that our analysis is not supposed to conflict with, or replace, technical notions of *explanation*. In particular, an explanation may render X understandable with respect

[1] Note that we assume that R does not understand X with respect to Y yet. If they already understand X then nothing would *make* them understand.

to Y precisely because the explanation carries the relevant information and structure as identified by technical accounts.

Overall, the idea is to enable examination of explainability by measuring understanding, e.g., in psychological studies of whether the processing of certain explanations makes stakeholders understand the explanandum with respect to the relevant aspect in relevant contexts. Explainability is not a technical concept but tightly coupled to human understanding. As such, it is also a matter of degree and probability. In the following investigation, we mostly omit this quantitative nature of explanations. Future work, however, should investigate this more rigorously.

B. Explainable Systems

What makes a system explainable with respect to a particular group in a certain context is the group member's access to explanations when in that context. To provide access to explanations, the latter need to be produced by something or someone. That which produces an explanation—what we here call the "means" of explanation—could be the system itself, another system, or even a human expert. The mere theoretical existence of some explanation is, however, not sufficient for a system to be explainable. We leverage the above characterization of explanation in order to specify what it takes for a system to be considered explainable:

Definition 2 (Explainable System): A system S is *explainable* by means M with respect to aspect Y of an explanandum[2] X, for target group G in context C, if and only if M is able to produce an E in context C such that E is an explanation of X with respect to Y, for G in C.

In general, a means M to produce an explanation of some aspect Y does not have to be part of the system S but may be provided by someone or something detached from S. Reconsidering the hiring example, explanations of the resulting ranking are dynamic and based on acquired data. It is natural to integrate the respective means directly into the system. However, in order to understand whether the system only considers applicant features that can legally be considered in hiring processes, applicants could also ask which criteria it considers. Such information about a system is static and already known at design time. As a result, the typical explanation of what a system's capabilities and features are is provided by human engineers in the form of documentation or a manual, which perfectly matches our characterization.

Just as the notion of explanation is not absolute but depends on an aspect of an explanandum and a target group in some context, a system is not just explainable *per se*, but only with respect to certain aspects, groups and contexts. Every unqualified use of the term "explainable" is under-specified.

C. Explainability Requirements

With Definitions 1 and 2 in place, we can now capture explainablity requirements. To meet the expectations of all stakeholders regarding explainability, we propose the following catalog of questions as a basis for elicitation of the requirements:

1) What are the relevant target groups G, e.g., engineers, end users, or lawyers, and which traits characterize each group's representatives R, e.g., specific background knowledge or cognitive capacities?
2) What are the explananda X, e.g., events or decisions?
3) Which aspects Y of the explananda X must be explained to which target group G, e.g., why is a decision justified, which causal chain of internal system events led up to it, why did some event e happen instead of event e'?
4) In which context C may an aspect Y need explanation, and what are the implied constraints? For example, explanations might have to be aural in a driving situation.

Based on the answers to those questions, explainability requirements are then formulated using the following schema:

Definition 3 (Explainability Requirement): A system S must be explainable for target group G in context C with respect to aspect Y of explanandum X.

Conceptually, requiring a system to be explainable does not entail a specific function that the system must be capable of performing, but rather constrains how it may be implemented. Choosing certain development techniques may impede the ability to provide explanations with the desired qualities, or they may conflict with requirements like privacy, e.g., explanations may leak personal information about the applicants. In general, a trade-off between the degree of explainability and other goals must be made. In line with this and the tight coupling of explainability with human understanding we suggest to understand explainability requirements as *Non-Functional Requirements* (NFRs) of a specific kind that must be *satisficed* rather than satisfied [38].

In the following, we will illustrate how our understanding of explainability requirements facilitates the elicitation of requirements, and enables their consideration during development, in tandem with other NFRs.

Softgoal Interdependency Graphs (SIGs) [39] represent and record the software design and reasoning process as well as the relationship among multiple requirements in the context of NFRs. Here, the main requirements constitute the acyclic graph's top nodes which are iteratively refined into sub-softgoals, forming the graph's middle layer, and eventually flow into the bottom layer which links concrete development techniques, coined *operationalizations*, to the fine-grained softgoals.

When considering the explainability of a specific system, a central question is what this demand actually boils down to, i.e., which explanandum X must be explained. Given specified explainability requirements, these X are already identified, e.g., explainable decisions, and can be modeled as top-level softgoals of an SIG.

Decomposition and elicitation of sub-softgoals lie at the heart of building SIGs. Naturally, the requirement to make a system explainable can be decomposed guided by our concept of explainability. Given explainability requirements, the refinement of X with respect to the relevant aspects Y

[2] Here X is not an arbitrary explanandum but an explanandum related to S.

is already given, and can be improved by considering the groups G and contexts C. Incrementally refining the softgoals in this way facilitates systematic elicitation and decomposition of the overarching explainability softgoal since the scope of subgoals is increasingly constrained. In fact, related NFRs, like transparency of the code base [40], contribute to the overarching explainability softgoal and will occur as subgoals in the SIG.

However, explainability requirements may conflict with other softgoals such as performance, development cost, precision, or security. A less explainable system may be cheaper to build or could offer a higher performance. The SIG notion acknowledges this and offers *priorities* and *interdependency links* between softgoals as the central concepts to support decision making.

In an SIG, the explainability (sub-)softgoals will be placed among the other softgoals, such that conflicting ones can be linked and associated with a positive or negative contribution. Likewise, when possible operationalizations to realize explainability softgoals affect others in different ways, their contribution is tracked in the links. For example, when considering different machine-learning based operationalizations for classification systems, *neural networks* might increase performance and reduce development costs. However, they may lack interpretability, while the simpler *decision trees* may be found to have significantly better interpretability without sacrificing the other criteria. Embedding explainability requirements into SIGs makes such trade-offs explicit and enables recording of design decisions through further notation offered by SIGs.

To aid in the refinement, operationalization and conflict resolution processes, the *NFR Framework* [39] proposes to build knowledge bases, coined *catalogs*, that accumulate possible refinements and interdependencies considered in previous projects. Having appropriate catalogs at hand may help to alleviate that need, and simplify the construction of SIGs—in particular when developers need some source of domain knowledge before moving towards the actual operationalizations and the target artifact. To start the refinement catalog off, our explainability terminology induces several patterns, e.g., decomposition of explainability softgoals by target-groups.

Finally, based on Definition 1, an explainability requirement is met if and only if an explanation E is provided such that the processing of E in context C by any representative R of G makes R understand X with respect to Y. Mapping this pattern onto refinements in SIGs enables the decomposition of broad and abstract explainablity softgoals, such as "the system must be explainable" or "decisions of the system must be explainable" down to fine-grained explainability requirements and softgoals. Explainability of the overall system is then satisficed by satisficing the resulting explainability sub-softgoals.

D. Assessing Understanding

In order to gain empirical confidence that a certain explanation is really understood by any representative, it might not be enough to provide a single representative with an explanation for a given explanandum and go through a checklist that assesses whether they understood the explanation. One of the problems with such approaches is that an individual representative might still have idiosyncratic understanding of an issue. In addition, assessing understanding through self-report questionnaires tends to suffer from cognitive biases, e.g., when people overestimate their understanding. A more promising approach would be to choose a variety of representatives of a target group with different backgrounds, e.g., different age, gender, experience with a given problem, and provide them with explanations. The feedback from all these representatives could then be used to gain insights into the target group's explanatory needs.

Furthermore, as the same explanation generally triggers different cognitive processes within people with different background and motivation [41], [42], it seems necessary to gain deeper insights into the representatives' processing of explanations. For instance, one could use the think-aloud technique [43] trying to understand how people perceive a given explanation. After processing the explanation, the representatives could try to use self-explanation [44] to answer their own questions based on the explanation. This would show whether the explanation helped them to understand the issue. The representatives could then also be asked to try to transfer their new knowledge to a related issue [44]. This would help to evaluate whether the explanation not only helps them to understand a specific issue, but also enables people to transfer their new knowledge to new situations. These steps allow us to examine understanding within a person and are examples of how to assess whether a given system is explainable. The fields of cognitive science and education provide further ideas for insights into processes that generate understanding [45].

By relying on the concept of understanding, our overarching characterization makes explainability measurable, using established techniques from psychology and cognitive sciences. In any case, revealing that representatives were not able to follow an explanation and that it did not enhance their knowledge should lead to iterative processes to improve the overall explainability of the system.

V. Conclusion

While explainability has become an important design desideratum it is under-specified what precisely is meant when demanding explainability. What shall be explainable to whom? How can an artifact be evaluated with respect to explainability requirements? How can explainability be achieved by design? In this paper, we briefly discussed various works in the area of explainable systems and presented a conceptual analysis which we used for the systematic specification and elicitation of explainability requirements.

Our long-term vision is to establish a standardized certification process in tandem with appropriate development techniques to achieve explainability by design. This paper is a starting point towards an overarching and systematic approach to explainability requirements. In future work, we intend to validate the proposed techniques in empirical studies, to develop explainability catalogs, and to identify potentially overlooked issues and improvements to our approach.

While we clarified what makes a system explainable and how explainability can be assessed empirically, further research is necessary on how to apply requirements and software engineering techniques to design explainable systems.

Acknowledgements

This work has received financial support by DFG grant 389792660 as part of TRR 248, by the ERC Advanced Investigators Grant 695614 (POWVER), and by the VolkswagenStiftung planning grant “Explainable Intelligent Systems.” It is based on discussions and work presented during the GI-Dagstuhl Seminar ES4CPS [46].

References

[1] “Broad agency announcement, explainable artificial intelligence (XAI), DARPA-BAA-16-53,” DARPA, Aug. 2016. [Online]. Available: https://www.darpa.mil/attachments/DARPA-BAA-16-53.pdf

[2] Z. C. Lipton, “The mythos of model interpretability,” *Queue*, 2018.

[3] B. D. Mittelstadt, P. Allo, M. Taddeo, S. Wachter, and L. Floridi, “The ethics of algorithms: Mapping the debate,” *Big Data & Society*, 2016.

[4] L. Floridi, J. Cowls, M. Beltrametti, R. Chatila, P. Chazerand, V. Dignum, C. Lütge, R. Madelin, U. Pagallo, F. Rossi, B. Schafer, P. Valcke, and E. Vayena, “Ai4people—an ethical framework for a good ai society: Opportunities, risks, principles, and recommendations,” *Minds and Machines*, 2018.

[5] M. Lahijanian and M. Kwiatkowska, “Social trust: A major challenge for the future of autonomous systems,” in *2016 AAAI Fall Symposia, Arlington, Virginia, USA, November 17-19, 2016*, 2016.

[6] L. R. Ye and P. E. Johnson, “The impact of explanation facilities on user acceptance of expert systems advice,” *MIS Quarterly*, 1995.

[7] S. Wachter, B. D. Mittelstadt, and C. Russell, “Counterfactual explanations without opening the black box: Automated decisions and the GDPR,” *arXiv preprint arXiv:1711.00399*, 2017.

[8] The European Parliament and the Council of the European Union. (2016, April) Commission Regulation (EU) 2016/679. [Online]. Available: http://data.europa.eu/eli/reg/2016/679/oj

[9] High-Level Expert Group on Artificial Intelligence. (2019) Ethics Guidelines for Trustworthy AI. [Online]. Available: https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

[10] T. Miller, “Explanation in artificial intelligence: Insights from the social sciences,” *Artificial Intelligence*, 2019.

[11] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” *arXiv preprint arXiv:1702.08608*, 2017.

[12] J. Masthoff, N. Oren, K. van Deemter, and W. W. Vasconcelos, “Towards scrutable autonomous systems,” in *Symposium: Influencing People with Information*, 2012.

[13] W. Samek, T. Wiegand, and K.-R. Müller, “Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models,” *arXiv preprint arXiv:1708.08296*, 2017.

[14] S. M. Lundberg and S. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017.

[15] O. Biran and C. Cotton, “Explanation and justification in machine learning: A survey,” in *IJCAI-17 workshop on explainable AI (XAI)*, 2017.

[16] I. Kononenko, E. Štrumbelj, Z. Bosnić, D. Pevec, M. Kukar, and M. Robnik-Šikonja, “Explanation and reliability of individual predictions,” *Informatica*, vol. 37, no. 1, 2013.

[17] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. MÃžller, “How to explain individual classification decisions,” *Journal of Machine Learning Research*, 2010.

[18] D. Martens and F. Provost, “Explaining data-driven document classifications,” 2013.

[19] M. T. Ribeiro, S. Singh, and C. Guestrin, “"why should I trust you?": Explaining the predictions of any classifier,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 2016.

[20] T. Miller, “Contrastive explanation: A structural-model approach,” *arXiv preprint arXiv:1811.03163*, 2018.

[21] A. Singh, C. Rose, K. Visweswariah, V. Chenthamarakshan, and N. Kambhatla, “Prospect: A system for screening candidates for recruitment,” in *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010.

[22] A. M. Ryan and N. T. Tippins, “Attracting and selecting: What psychological research tells us,” *Human Resource Management: Published in Cooperation with the School of Business Administration, The University of Michigan and in alliance with the Society of Human Resources Management*, 2004.

[23] S. W. Gilliland, “The perceived fairness of selection systems: An organizational justice perspective,” *Academy of Management Review*, 1993.

[24] D. E. Terpstra and E. J. Rozell, “The relationship of staffing practices to organizational level measures of performance,” *Personnel psychology*, 1993.

[25] D. M. Truxillo, D. D. Steiner, and S. W. Gilliland, “The importance of organizational justice in personnel selection: Defining when selection fairness really matters,” *International Journal of Selection and Assessment*, 2004.

[26] J. Y. Halpern, “Causes and explanations: A structural-model approach. part ii: Explanations,” *British Journal for the Philosophy of Science*, 2005.

[27] B. C. Van Fraassen, *The Scientific Image*. Oxford University Press, 1980.

[28] S. Bromberger, *On What We Know We Don’t Know: Explanation, Theory, Linguistics, and How Questions Shape Them*. University of Chicago Press, 1992.

[29] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, “Explaining explanations: An overview of interpretability of machine learning,” in *5th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2018, Turin, Italy, October 1-3, 2018*, 2018.

[30] J. Woodward, “Scientific explanation,” in *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, 2017.

[31] A. G. Sutcliffe, S. Thew, and P. Jarvis, “Experience with user-centred requirements engineering,” *Requir. Eng.*, 2011.

[32] A. Sutcliffe, *User-centred Requirements Engineering*. Springer Science & Business Media, 2012.

[33] D. M. Truxillo, T. E. Bodner, M. Bertolino, T. N. Bauer, and C. A. Yonce, “Effects of explanations on applicant reactions: A meta-analytic review,” *International Journal of Selection and Assessment*, 2009.

[34] P. Achinstein, *Evidence, Explanation, and Realism: Essays in Philosophy of Science*. Oxford University Press, 2010.

[35] B. C. Van Fraassen, “The pragmatics of explanation,” *American Philosophical Quarterly*, 1977.

[36] C. Bechlivanidis, D. A. Mullen, Lagnado, J. C. Zemla, and S. Sloman, “Concreteness and abstraction in everyday explanation,” *Psychonomical Bulletin & Review*, vol. 24, no. 5, pp. 1451 – 1464, 2017.

[37] M. Langer, C. J. König, and A. Fitli, “Information as a double-edged sword: The role of computer experience and information on applicant reactions towards novel technologies for personnel selection,” *Computers in Human Behavior*, vol. 81, pp. 19–30, 2018.

[38] H. A. Simon, *The Sciences of the Artificial (3rd Ed.)*. MIT Press, 1996.

[39] L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos, *Non-Functional Requirements in Software Engineering*. Springer, 2000.

[40] L. M. Cysneiros, M. Raffi, and J. C. S. do Prado Leite, “Software transparency as a key requirement for self-driving cars,” in *26th IEEE International Requirements Engineering Conference, RE 2018, Banff, AB, Canada, August 20-24, 2018*, 2018.

[41] T. Lombrozo, “The structure and function of explanations,” *Trends in cognitive sciences*, 2006.

[42] L. J. Skitka, E. Mullen, T. Griffin, S. Hutchinson, and B. Chamberlin, “Dispositions, scripts, or motivated correction? understanding ideological differences in explanations for social problems.” *Journal of personality and social psychology*, 2002.

[43] K. A. Ericsson and H. A. Simon, *Protocol Analysis: Verbal Reports as Data*. MIT Press, 1984.

[44] M. T. Chi, N. De Leeuw, M.-H. Chiu, and C. LaVancher, “Eliciting self-explanations improves understanding,” *Cognitive science*, 1994.

[45] R. White and R. Gunstone, *Probing Understanding*. Routledge, 2014.

[46] J. Greenyer, M. Lochau, and T. Vogel, “Explainable software for cyber-physical systems (ES4CPS): report from the GI dagstuhl seminar 19023, january 06-11 2019, schloss dagstuhl,” *CoRR*, vol. abs/1904.11851, 2019.