Gottfried Wilhelm
Leibniz Universität Hannover
Fakultät für Elektrotechnik und Informatik
Institut für Praktische Informatik
Fachgebiet Software Engineering

Criteria and Metrics for the Explainability of Software

Masterarbeit

im Studiengang Informatik

von

Hannah Deters

Prüfer: Prof. Dr. rer. nat. Kurt Schneider
Zweitprüfer: Dr. rer. nat. Jil Klünder
Betreuer: Larissa Chazette, M. Sc.

Hannover, 28.09.2022

Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig und ohne fremde Hilfe verfasst und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel verwendet habe. Die Arbeit hat in gleicher oder ähnlicher Form noch keinem anderen Prüfungsamt vorgelegen.

Hannover, den 28.09.2022

Hannah Deters

Abstract

In this master thesis, a concept for the evaluation of explainability in software systems was developed. For this purpose, a comprehensive literature review was conducted in which 86 relevant papers were obtained from an initial set of 1025 papers. These papers contributed to the conceptualization of the evaluation method. During this conceptualization, it was found that the characteristics of explainability are strongly linked to the objective that the explanations are supposed to achieve. It became clear that it is not possible to achieve a satisfactory result if the evaluation of explainability does not take these objectives into account. What has also been noticed is that the literature already provides methods for evaluating single aspects of explainability, but these consist almost exclusively of user studies. Since conducting multiple user studies would be unrealistically expensive in non-research settings, heuristics were developed to provide a first estimate of explainability. Overall, an overarching concept was developed that links the definition of objectives, the initial assessment with heuristics, and the more reliable evaluation with user studies.

In the second part of the master's thesis, a user study was conducted to evaluate whether the developed heuristics produce reliable results. For this purpose, the interrater agreement was examined to see whether the heuristics allow uniform ratings. It was found that a group of evaluators together can produce a uniform result. Significance tests were then used to determine whether the heuristics are capable of identifying significant differences in the explainability of two systems. It was found that significant differences were revealed within the different aspects of explainability.

Zusammenfassung

In dieser Masterarbeit wurde ein Konzept zur Bewertung von Erklärbarkeit in Softwaresystemen entwickelt. Dazu wurde eine umfassende Literaturrecherche durchgeführt, bei der aus einer anfänglichen Menge von 1025 Papers 86 relevante Papers ermittelt wurden. Diese Paper trugen zur Konzeptualisierung der Evaluationsmethode bei. Während dieser Konzeptualisierung wurde festgestellt, dass die Merkmale der Erklärbarkeit eng mit dem Ziel verbunden sind, das mit den Erklärungen erreicht werden soll. Es wurde deutlich, dass es nicht möglich ist, ein zufriedenstellendes Ergebnis zu erhalten, wenn bei der Bewertung der Erklärbarkeit diese Ziele nicht berücksichtigt werden. Es wurde zudem festgestellt, dass es in der Literatur zwar bereits Methoden zur Bewertung einzelner Aspekte der Erklärbarkeit gibt, diese aber fast ausschließlich aus Nutzerstudien bestehen. Da die Durchführung mehrerer Nutzerstudien in einem nicht-wissenschaftlichen Rahmen unrealistisch kostenintensiv wäre, wurden Heuristiken entwickelt, um eine erste Einschätzung der Erklärbarkeit zu erhalten. Insgesamt wurde ein übergreifendes Konzept entwickelt, das die Zieldefinition, die erste Einschätzung mit Heuristiken und die zuverlässigere Bewertung mit Nutzerstudien verbindet.

Im zweiten Teil der Masterarbeit wurde eine Nutzerstudie durchgeführt, um zu evaluieren, ob die entwickelten Heuristiken zuverlässige Ergebnisse liefern. Zu diesem Zweck wurde die Interrater-Übereinstimmung untersucht, um festzustellen, ob die Heuristiken eine einheitliche Bewertunge ermöglichen. Es wurde festgestellt, dass eine Gruppe von Bewertern zusammen ein einheitliches Ergebnis erzielen kann. Anschließend wurde anhand von Signifikanztests geprüft, ob die Heuristiken in der Lage sind, signifikante Unterschiede in der Erklärbarkeit zweier Systemen zu erkennen. Es wurde nachgewiesen, dass innerhalb verschiedener Aspekte der Erklärbarkeit signifikante Unterschiede festgestellt werden konnten.

Contents

Chapter 1

Introduction

Explainability of software describes the capability of a system to explain itself and its own behavior. This is needed for any system that contains elements that are not clear to a user. A typical example of systems that require explainability are AI systems. AI Systems are black boxes that produce results that would be beyond the user's understanding without any explanation. A system that is so well established that any target user already knows how it works, such as an old simple telephone, would not require explainability. Explainability is therefore a requirement that is not needed for every system in every case, but in certain situations in certain contexts for certain users. However, in situations where explanations are needed, explainability is an important demand that can significantly increase the quality of a system.

1.1 Motivation

The demand for explainability is growing rapidly, especially in the areas of AI, recommender systems and deep learning. In recent years, lots of research has been done on how explainability can be generated in such systems, how explanations can be displayed and implemented. An important consequence of this high demand is that there is an urgent need for an approach to measure explainability. The literature already offers methods to measure single aspects of explainability. However, to the best of our knowledge, there is no procedure to measure the overall explainability of a system, especially not for every type of system. Not only systems from the above-mentioned areas should be measurable, but every possible system. For this purpose, a concept is developed and prototypically implemented as part of this thesis to help with the evaluation and make it as explicit as possible.

1.2 Solution Approach

In order to develop an approach for the evaluation of explainability, the existing literature was first examined. For this purpose, a secondary literature review was executed. The information obtained from the literature was then processed into a concept that combines individual approaches to enable the evaluation of explainability of software in general. Afterwards, own ideas are presented supporting the previously created concept. As part of this, heuristics were developed based on the results of the literature, which intend to allow a primary estimation of the explainability. These heuristics were then evaluated to determine whether they provide consistent results. Furthermore, a prototype was developed that realizes the entire concept. This prototype was implemented for illustration purposes as an exemplary java application. In addition, the study to test the heuristics was carried out on this application. This procedure ensures that the final prototype is based on a scientifically well-founded concept.

1.3 Structure of the Thesis

In order to establish a uniform consensus regarding terminology and to guarantee the necessary state of knowledge, chapter 2 first clarifies the foundations regarding explainability and also metrics in the field of software engineering. Chapter 3 then describes how exactly the literature review was conducted and what literature was found. Afterwards, the most comprehensive part of the master thesis is presented in chapter 4. The concept that was created from the literature is described extensively, and a conclusion is drawn as to what influence this has on the prototype developed in chapter 5. In this chapter, the heuristics that were created are first introduced and reasons are given how these heuristics were created. Then the structure of the prototype is explained and presented with pictures. Subsequently, a user study was conducted to evaluate one part of the prototype – namely the heuristics. The procedure and results of this study are discussed in chapters 6 and 7. Chapter 8 then discusses the overall result by answering the research questions and, in turn, identifies limitations that must be taken into account. Finally, a conclusion is drawn in chapter 9 and an orientation is given as to what future research this thesis leads to.

Chapter 2

Background and Related Work

This chapter presents the basics of explainability and metrics within software engineering. Explainability is discussed in particular in the context of a non-functional requirement, but also the role of explainability in the context of AI and why it is becoming increasingly important in other areas as well. The section on metrics serves primarily to clarify terminology and to ensure that there is a basic understanding of this topic.

2.1 Explainability

Explainability of a software system refers to its ability to explain itself. Chazette et al. [12] have established a formal definition of explainability that captures important aspects.

> A system S is explainable with respect to an aspect X of S relative to an addressee A in context C if and only if there is an entity E (the explainer) who, by giving a corpus of information I (the explanation of X), enables A to understand X of S in C.

Definition 2.1: Explainability defined by Chazette et al. [12]

According to Chazette et al. [12] explanations are transmitted by an entity E which is called an explainer. This refers to the part of the system that transmits the information, which in a simple example could be a dialog that presents a textual explanation. One important point of the definition is that the property *explainable* refers to an aspect X of the system. That means, that explainability is considered here with regard to a certain aspect which can be, for example, the parameters and data structures of the system. However, the aspect X can also refer to the system in general. Furthermore, the addressee A is important. An explanation is not equally informative for every person. Prior knowledge, for example, plays a major role in the understanding of an explanation, as well as the type of presentation, which can be perceived differently depending on the user's preferences. The same applies to the context in which the explanation is given. According to Chazette et al. [12], a plausible factor here would be time pressure or the type of system. Lastly, the information I that is transmitted is important. That this information is appropriate is a necessary condition for a system/aspect to be considered explainable. If useless or wrong information is conveyed, even a good presentation or adaptation to the user and context will be of no use.

Another aspect that is mentioned in the context of explainability is the goal to be achieved by the explanations. Tintarev and Masthoff [84] emphasize that the benefits that the person wants to obtain from the explanation must be considered. Kass and Finin [39] describe an explanation as valuable if it contributes to the accomplishment of user goals, and Hoffman et al. [33] refer to this aspect as *goal-relevance*. Thus, it can be seen that the goal to be achieved by the explanation also plays a relevant role in explainability.

2.1.1 Explanations

Köhl et al. [48] point out that explanations can be both technical descriptions and pragmatic answers to users' questions. Brunotte et al. [8], in turn, consider explainability from a privacy awareness context. Explanations can therefore also be unsolicited responses to the processing and storage of data. Arya et al. [3], meanwhile, consider explainability from the perspective of AI systems and see explanations as a means of gaining insight into the system and understanding the decision-making process. What becomes clear is that explanations in software systems have many facets and are needed in many different areas. So it is important to first define what is meant when talking about explanations. Considering the goal of this work to develop a concept for measuring explainability in general, it is important to include all possible types of explanations. Thus, when talking about explanations here and in the following chapters, any interface element of a software system in which the system explains itself is meant.

Explanations can also have many manifestations. They can be, for example, presented in textual form. Written sentences as text is probably the explanation that comes to mind at first, but also a single word can be an explanation, if it contains appropriate information about the system. Explanations can also be presented visually using figures – for example, Simonyan et al. [76] used saliency maps to illustrate an image classification model by displaying which areas were essential for the classification. An explanation can also be a simple color highlighting of elements, using red or green to indicate whether a result from image recognition has good or bad predictive power. There are many more examples of forms of explanation that will not be discussed further here. What becomes clear is that explanations can be given in many different ways. The important point here is that the system conveys information to the addressee with which the system explains a certain aspect of itself.

2.1.2 Focus of Current Research on Explainability

Explainability has already been widely introduced in areas of artificial intelligence. [29, 52, 69, 74] This could be due to the fact that artificial intelligence is an alienating topic for human beings, for which explanations are particularly important in order to be able to trust these systems. The urge of companies to mitigate negative feelings of users is stronger than the urge to initiate developments for positive changes. This may be a reason explainability has been explored mostly in AI domains so far. However, this does not mean that explainability cannot be very powerful in other software domains as well. Since being recently considered as a non-functional requirement, it offers a wide range of opportunities to improve software systems and can immensely improve the user experience. Chazette et al. [12] show how many quality aspects are positively affected by explainability. It is therefore important to not only relate explainability and, accordingly, its measurement to AI-based systems, but to take every software type into account.

2.2 Software Metrics

There are many definitions for metrics in the field of software engineering, which, however, all focus on the same aspects. Metrics provide some kind of measurement for the software and the production process, so that quantitative values are generated which can be used for evaluation. Thus, software data is taken as input, resulting in a numerical value as output, from which the degree to which a certain attribute is fulfilled can be recognized. [36, 34, 56]

To further specify software metrics, Honglei et al. [34] divided metrics into three areas that are frequently referred to in the literature: procedure metrics, project metrics and product metrics. Procedure metrics help to evaluate the procedure of software development, focusing on aspects such as the duration of certain phases or the efficiency of certain methods. Project metrics, on the other hand, help to understand the project situation and status, taking into account aspects such as risks and costs of the project. Product metrics, which will be the focus of this work, are used to understand and control the quality of the software. [34]

According to the IEEE Standard for a Software Quality Methodology [36], software quality is the "degree to which software possesses a desired combination of attributes". This indicates that before suitable metrics can be obtained, it is necessary to determine what the desired combination of attributes is that is to be achieved. Therefore, the first step is to establish criteria and sub-criteria for *good* explainability that can subsequently be measured.

2.2.1 Goals of Metrics

One thing that should always be kept in mind when creating, selecting, or applying metrics are the goals that the metrics are intended to achieve. Metrics should be designed in a way, so that they can provide some kind of benefit. This includes that it should be possible to derive relevant information from the output of the metric that is useful to the product or project. Possible benefits that can be drawn from metrics are as follows:

- Support in the establishment of quality requirements [36]
- Analysis of the deviation between the quality of the real software and the established requirements. [36, 34]
- Improvement of the quality of the product by pinpointing the places where defects could occur to increase customer satisfaction. [34, 56]
- Comparison of two systems with regard to a specific aspect.
- Observation of changes in quality when the system is modified. [36]

If none of these benefits are met, then it should be considered whether the metric has any benefit, and therefore what it can be used for.

2.3 Related Work

As mentioned above, research in the area of explainability of AI systems is already well advanced. Thus, there are already some SLRs on the topic of explainability, and even in the area of criteria and partly also metrics. Tintarev and Masthoff [84] published a paper in year 2011 involving the evaluation of explanations in software systems. In the year after, they further published a paper in which they specialized this evaluation exclusively on the effectiveness of the explanations. [85] However, a major limitation of these publications is that the research explicitly refers only to recommender systems. There are also a few other publications that attempt to explore what aspects make explanations good and thus capture criteria for good explainability. [10, 11, 51, 52, 88] However, all of these publications focus explicitly on certain types of systems – namely AI-related systems. Moreover, methods for measuring these criteria were not sufficiently addressed in these publications. Mohseni et al. [57] published an extensive paper in 2021 that presented both criteria and metrics in a clear and detailed manner. Nevertheless, they also explicitly referred to explainable AI systems. Overall, it can be seen that the evaluation of explainability has been considered, but it is explicitly limited to AI systems. As argued in section 2.1.2, explainability is not only relevant in these areas, but should be considered for all types of software. This problem is addressed in this master thesis by developing an accessible concept of evaluation that is applicable to all types of software systems.

Chapter 3

Literature Review

This chapter explains how the literature, on which the concept described in chapter 4 is based, was collected. Initially, the approach was to conduct a systematic literature review, but after some deliberation which will be explained later, it was decided to conduct a secondary literature review. The process and reasons are explained in detail below.

3.1 Research Questions

In order to obtain relevant information from the literature review, it is important to first determine which research questions are to be answered here. For this purpose, the following research questions were defined:

RQ 1 What criteria have already been established in the literature that define good explainability?

RQ 2 What metrics for measuring explainability are frequently used or recommended in the literature?

RQ 3 Is it possible to measure the explainability of a software system, regardless of the type of system?

The research questions RQ1 and RQ2 were raised to simultaneously identify the state of the literature and support the development of an overarching conceptual framework for evaluating explainability. The fact that the research questions are very broad and that it will not be possible to answer them clearly in a single paragraph at the end is due to the goal of the thesis. In a next step, the concept that emerges from these questions can be tested with more explicit research questions, but until then, the concept must first emerge. The third research question attempts to determine whether there is a universal way to measure the explainability of software systems at all. Since the criteria and metrics are currently designed for very specific systems, it is not clear whether explainability can be measured in a universal valid way. This research question will not be answered directly by the literature, but in the conceptual phase afterwards.

3.2 Procedure

The basic approach was to form a starting set containing basic conceptualizing papers on the topic of criteria and metrics for the explainability of software systems. The generation of this stating set is explained in section 3.2.2. From this set, the most relevant papers were then selected that were thematically as diversified as possible. These papers were subsequently used to initiate a Forward Snowballing phase and a Backward Snowballing phase, resulting in the final set of literature. Based on the inclusion and exclusion criteria mentioned below, papers were filtered using the following procedure:

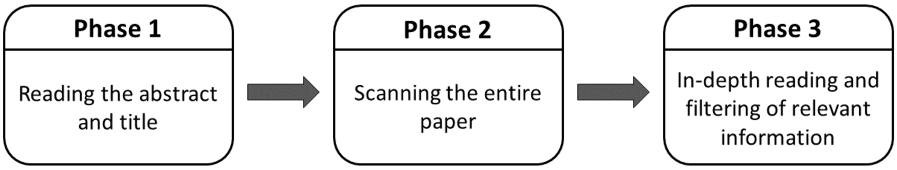

Phase 1
Reading the abstract and title
Phase 2
Scanning the entire paper
Phase 3
In-depth reading and filtering of relevant information

Figure 3.1: Phases during literature review

3.2.1 Inclusion and Exclusion Criteria

Inclusion and exclusion criteria were established to decide whether a paper is relevant for this work. In the following phases, these criteria were strictly applied in order to ensure traceability and counteract subjectivity. While exclusion criteria take precedence over inclusion criteria, the paper was included if it passes any of the inclusion criteria (I1 - I2) but was excluded if it meets any of the exclusion criteria (E1 - E3).

Inclusion criteria:

I1 The paper specifies what constitutes good explainability. (RQ1)

I2 The paper specifies how explainability can be evaluated. (RQ2)

I3 The paper explicitly evaluates the explainability of a given software system. (RQ2)

Exclusion criteria:

E1 The paper is not peer reviewed.

E2 The paper is not freely available for reading.

E3 The paper is not written in English or German language.

3.2.2 Definition of the Startset

The first approach to generate a start set was a database search using a search query, which corresponds to the standardized approach of a systematic literature review. The following search query was created for this purpose.

> (explainability OR interpretability OR “explainable systems”) AND
> (metric OR criteria OR measure OR evaluation OR maturity) AND
> (software OR “requirements engineering” OR hci OR “human
> computer interaction”)

Google Scholar returned *19,600 results*, which showed that the query did not sort out irrelevant papers. The restriction that criteria or some kind of evaluation must occur, was not sufficient, because many papers use one of these words, but, for example, evaluate completely different things than explainability. The attempt to sort out irrelevant papers was not successful, even after a few more attempts. Without running the risk of excluding relevant papers as well, the number of results could not be reduced in a reasonable way.

As a solution, the literature set from the paper *Exploring Explainability: A Definition, a Model, and a Knowledge Catalogue* by Chazette et al. [13] was taken as the start set for my literature search. This paper is a current SLR on the development of explainable systems. Thus, this start set provides a good basis for the secondary literature review for this thesis.
This start set contained 229 papers. After completing phase 1 (reading abstract and title), there were 33 papers left that matched the inclusion but not exclusion criteria. In phase 2 (scanning the whole paper) another 18 papers were rejected, so that altogether 15 papers were left.

Start	Phase 1	Phase 2
229 paper	33 paper	15 paper

From these 15 papers, the ones that analyze and not just apply the criteria and metrics for explainability were selected to reduce the number of papers irrelevant to my topic in the snowballing phase. From these eight papers, the four papers that differed well from each other were selected in order to achieve a large variety in the snowballing set. [33, 63, 77, 84] In this regard, attention was paid to the diversity of the authors and the subject areas. On the other hand, care was also taken to ensure that the topics fit in with the goal of the thesis as much as possible, in order to again avoid irrelevant papers in the snowballing set. These four papers resulted in a snowballing set with a total of *1,025 papers.*

3.2.3 Forward Snowballing

For the forward snowballing set, all papers that cite one of the four papers from the start set and are published before 04/25/2022 and listed on Google Scholar were included. In total, all four papers were cited *769 times.* After sorting out duplicates among themselves and duplicates in the start set, *665 papers* were left. In the first phase (reading the title and abstract) 583 papers were rejected, leaving *82 papers.* In phase 2 (scanning of the entire paper), a further 36 papers were rejected, resulting in a total of *46 relevant papers* from the forward snowballing phase.

Start	Preprocessing	Phase 1	Phase 2
769 paper	665 paper	82 paper	46 paper

3.2.4 Backward Snowballing

In the backward snowballing set, all papers that were cited by the four papers were included – a total of 404 papers. During preprocessing, all duplicates were again removed. In addition, for the paper by Nunes and Jannach [63], it was taken into account that they had pre-sorted all cited papers so that all papers with a connection to criteria and metrics could be selected. After pre-processing, *131 papers* remained, of which 74 papers were rejected during phase 1. After scanning these papers (phase 2), 24 papers with relevant information were retrieved.

Start	Preprocessing	Phase 1	Phase 2
404 paper	131 paper	47 paper	24 paper

3.3 Results

In total, **86 papers** were found in this literature review that establish criteria and metrics for explainability of software systems, analyze them, or use them directly. These papers allowed to identify eleven main criteria, which will be presented in detail in chapter 4. Table 3.1 shows for each of these criteria which paper discussed the criterion itself, a sub-criterion or a method for measuring it.

Criterion	Paper
Understandability	[4, 11, 15, 19, 24, 37, 39, 55, 58, 60, 64, 71, 74, 78, 80, 87, 88, 89, 90, 94]
Transparency	[2, 6, 10, 11, 15, 16, 18, 19, 21, 22, 23, 25, 27, 29, 30, 32, 33, 35, 37, 38, 40, 41, 42, 45, 47, 49, 50, 52, 51, 53, 57, 59, 60, 61, 63, 64, 66, 67, 68, 69, 72, 77, 80, 83, 84, 85, 87, 88, 89, 90, 94]
Effectiveness	[1, 2, 4, 5, 6, 10, 14, 15, 17, 20, 21, 22, 27, 30, 32, 38, 52, 54, 57, 58, 59, 61, 62, 63, 64, 67, 68, 70, 72, 73, 77, 78, 81, 82, 83, 84, 85, 87, 88, 90, 93]
Efficiency	[5, 10, 11, 15, 18, 19, 25, 27, 30, 33, 39, 47, 49, 50, 52, 61, 63, 80, 82, 84, 85, 87, 88]
Satisfaction	[2, 4, 6, 10, 11, 14, 15, 19, 22, 25, 27, 29, 30, 32, 41, 44, 45, 47, 49, 52, 51, 53, 54, 55, 57, 61, 62, 63, 64, 71, 73, 77, 78, 79, 80, 83, 84, 86, 87, 88, 89, 92, 95]
Correctness	[7, 11, 18, 21, 22, 50, 51, 55, 57, 66, 72, 77, 88, 89, 93]
Suitability	[11, 14, 22, 24, 33, 39, 44, 45, 50, 51, 54, 55, 61, 62, 66, 71, 72, 77, 82, 85, 87, 88, 89, 90]
Trustability	[2, 4, 5, 6, 10, 11, 14, 15, 18, 20, 19, 21, 22, 25, 29, 30, 32, 33, 35, 38, 45, 47, 52, 55, 57, 59, 60, 61, 62, 63, 64, 65, 66, 69, 70, 74, 77, 81, 84, 85, 87, 88, 89, 90, 91, 92, 93]
Persuasiveness	[2, 5, 6, 10, 15, 21, 25, 27, 30, 32, 35, 38, 39, 41, 44, 45, 59, 62, 63, 68, 78, 82, 84, 85, 86, 87, 88, 89, 90, 93, 95, 96]
Scrutability	[10, 38, 62, 63, 68, 84, 85, 88]
Debugability	[10, 35, 42, 47, 52, 63]

Table 3.1: Main criteria and corresponding literature

Chapter 4

Concept of Criteria and Metrics

In this chapter, the concept of evaluating the explainability of software systems is introduced. This concept is based on criteria and metrics found in the literature during the literature review. A total of eleven main criteria were identified, which were further refined with up to eight sub-criteria. For almost every sub-criterion, at least one metric is given to measure that criterion. For criteria for which no metrics could be found in the literature, own thoughts on measurements are presented. At the end of this chapter, a conclusion is drawn as to what particular impact this concept has on the prototype approach.

4.1 Main Criteria and Objectives

During the literature review, eleven main criteria were identified. These criteria are briefly presented in table 4.1. However, these criteria are not equally relevant for each type of system. For clarification, the following section identifies an objective to be achieved by explainability for each criterion.

Understandability aims to ensure that each target user can understand the explanation as easily as possible, which means that they need as little cognitive effort as possible or that they do not need to make any additional effort to understand the explanation. Therefore, the corresponding objective to be achieved with particularly understandable explanations is to provide the best possible knowledge with the least cognitive effort. Understandability is a criterion that will nearly always be required in terms of explainability and would only be considered negligible in very specialized, domain-specific areas where other objectives are more in focus. Transparency, on the other hand, has the objective of giving an accurate impression of how the system

ID	Criterion	Description
C1	Understandability	The explanations are easily understandable by the addressee.
C2	Transparency	The explanations provide sufficient insight into how the system works.
C3	Effectiveness	The explanations help the addressee to use the system better – Make better decisions, use functions that fit the best.
C4	Efficiency	The explanations help the addressee to use the system faster – Make decisions faster, understand the system faster, execute actions faster.
C5	Satisfaction	The explanations increase the comfort of use and the enjoyment.
C6	Correctness	The explanations are truthful.
C7	Suitability	The explanations are suited to the context, the user and the goal of the use.
C8	Trustability	The explanations help the addressee to have confidence in the system.
C9	Persuasiveness	The explanations convince the user to use / try / buy some system related item.
C10	Scrutability	The explanations help to correct the system if necessary.
C11	Debugability	The explanations allow users / software engineers to identify and localize defects in the system.

Table 4.1: Definition of main criteria

works. How easy it is to understand is not its main concern. However, it is of course possible to combine the objectives of understandability and transparency so that the system can be viewed as accurately as possible, while ensuring that it remains as understandable as possible.

The objective of explanations that meet the effectiveness criterion is the better use of the system. The meaning of *better* use of a system is very diverse. In the case of recommender systems, for example, more effective use would improve the user's final decisions. In the case of a simple ticketing system, an effective explanation would advise the user if there is a cheaper ticket available for the user's purpose. Somewhat opposed to effectiveness is efficiency. Efficient explanations help the user to use the system faster. The associated objective is therefore, the faster use of the system. In general, it is best to make a choice between the objectives of efficiency and effectiveness, since particularly effective use implies that all possibilities are taken into account, which in turn would take too much time for an efficient use. That means that a combination of both objectives will result in certain trade-offs on both sides.

Satisfaction is a criterion that aims the comfort of use and the enjoyment of the explanations. The objective of satisfaction therefore requests a higher overall comfort of use of the system and seeks the pleasure of the users.

Correctness focuses on the explanation being truthful. This can have different dimensions (for example completeness and soundness [77]) which is further explained in section 4.2.6. Correctness should always be considered, regardless of the goal of the system, however, there are varying degrees of relevance for different systems. In AI systems that automatically generate explanations, correctness is more important. Although correctness could be neglected depending on the company's goal, for example, to persuade customers to buy products with explanations that are not entirely honest, this would not be acceptable and is therefore not considered further below.

Another objective is the adaptability to people and situations. Explanations can help make a system suitable for different groups of users or situations. A necessary criterion for this objective is suitability, which ensures that the explanations themselves are adapted to these situations and people. For example, an explanation should have different characteristics if it has to be captured while driving a car than in a quiet situation in the office.

The criteria of trustability and persuasiveness are closely linked. Especially since their objectives target similar user behavior. The first objective is to increase the user's trust in the system and thus enable them to comply with the system. This supports the objective of convincing the user to use/try/purchase an item associated with the system. When there is a high level of trust, the objective of persuasion is easier to achieve.

Finally, there are two more objectives that can be achieved with the help of explainability. These two objectives focus on errors in the system. They differ mainly in the perspective from which they are viewed. Scrutability aims to allow a user to validate the system and, if necessary, tell it that it is wrong. These errors are not necessarily based on faulty implementation, but on mismatches between the user and the model. Debugability, on the other hand, targets to make it easier for a developer to detect and fix actual bugs in the implementation.

4.2 Criteria and Metrics

In this section, it is specified for each main criterion, which sub-criteria are required and how these sub-criteria can be measured. Therefore, metrics are presented to evaluate the extent to which the sub-criteria are met in a software system. First, aspects are listed that can be measured to evaluate a criterion. After that, methods are described how exactly these aspects can be measured. In addition, a small example of a system where the associated objective would be realistic is given for each criterion.

Almost every upper criterion can be measured in some way via the metric *M0 Subjective perception.* In this metric, users answer questions to state what their perception is. This metric can be measured in 3 ways: Answering post-study questionnaires, choosing the best implementation regarding a specific concern, and comparing the ratings of the system before and after seeing the explanation. In addition to the metrics found in literature, chapter 5 presents some heuristics with which the explainability can be estimated. These heuristics are integrated into the diagrams, but are not further explained in this chapter. For the denotation of the heuristics, the letter R was chosen[1] because the letter H is used later in the thesis to denote the null hypotheses.

4.2.1 Understandability

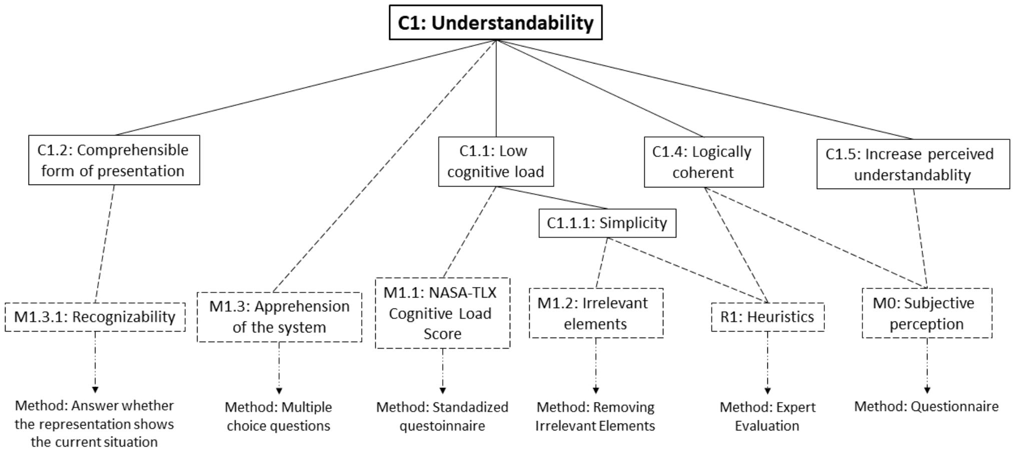

C1: Understandability
C1.2: Comprehensible form of presentation
C1.1: Low cognitive load
C1.4: Logically coherent
C1.5: Increase perceived understandablity
C1.1.1: Simplicity
M1.3.1: Recognizability
M1.3: Apprehension of the system
M1.1: NASA-TLX Cognitive Load Score
M1.2: Irrelevant elements
R1: Heuristics
M0: Subjective perception
Method: Answer whether the representation shows the current situation
Method: Multiple choice questions
Method: Standadized questoinnaire
Method: Removing Irrelevant Elements
Method: Expert Evaluation
Method: Questionnaire

Figure 4.1: C1 – Understandability criteria and metrics

To ensure the explainability of a software system, the first important aspect is to make the embedded explanations easily understandable for the user. This

[1] Heuristics are often colloquially referred to as **R**ules of thumb.

criterion is important for almost all systems, which is why no example is given at this point. For understandability, the explanation must fulfill the following points: It must reduce the cognitive load and thus be as simple as possible. The explanation must have a comprehensible form of presentation, which is adapted to the type of information to be conveyed. It must enable the user to grasp the information and be logically coherent to help the user link information. Lastly, it must increase the user's perceived comprehensibility.

Sub-criteria

C1.1 Reduce cognitive load: Since explainability is only a means to an end, it should not require more effort from the user than absolutely necessary. Thus, explanations should be presented in such a way that they convey the information with as little mental effort as possible for the user. [19, 39, 87, 90] To make explanations easily understandable, they should be kept simple. This includes, for example, that an explanation contains only as many elements as necessary and that natural language is used (*C1.1.1 Simplicity*). [4, 39, 55, 80, 89, 94]
C1.2 Comprehensible Form of Presentation: The form of presentation should be chosen so that the information is provided in an understandable way. In some cases, for example, it is possible to present the information visually with icons, while in other situations it is necessary to use more complicated illustrations such as heat maps or to provide a textual explanation. [4, 11, 60, 78]
C1.4 Logically Coherent: Furthermore, the parts of the explanation (e.g. sentences in a text) should be logically coherent. This enables the user to link information and understand it more easily. Explanations that are not coherent in themselves confuse the user and cause exactly the opposite of the intended purpose. [89]
C1.5 Increase perceived understandability: Finally, users should get the feeling that they can easily understand the explanation, since they are the ones who depend on it. [15]

Metrics

To measure the above criteria, various methods were found through the literature review. Figure 4.1 shows which metric can be used for which criterion. To measure the cognitive Load (*M1.1 NASA-TLX Cognitive Load Score*) Users are asked to answer six questions from a standardized questionnaire. This procedure was applied in three papers. [19, 87, 90] Wiegand et al. [94] used a method to determine whether the explanations are as simple as possible (*M1.2 Irrelevant Elements*). They asked users to remove elements from the explanation that were not necessary for their understanding. If most users remove certain elements, these elements might need to be removed from the explanation. To examine whether the users have processed the information correctly, Vilone and Longo [88] suggest asking users to answer a set of multiple choice questions regarding the information the explanation tried to explain (*M1.3 Apprehension of the system*). If the participants can not answer correctly, the understandability of the explanation is insufficient. Iyer et al. [37] checked the comprehensibility of a saliency map by asking users to judge whether the representation shown fits the situation or not (*M1.3.1 Recognizability*). For this purpose, they are alternately shown an explanation that fits the situation and an explanation that does not fit the situation. The perceived understandability is assessed using questionnaires in several papers. [4, 24, 58, 80, 89] Table A.1 in the appendix contains a compilation of these questions.

4.2.2 Transparency

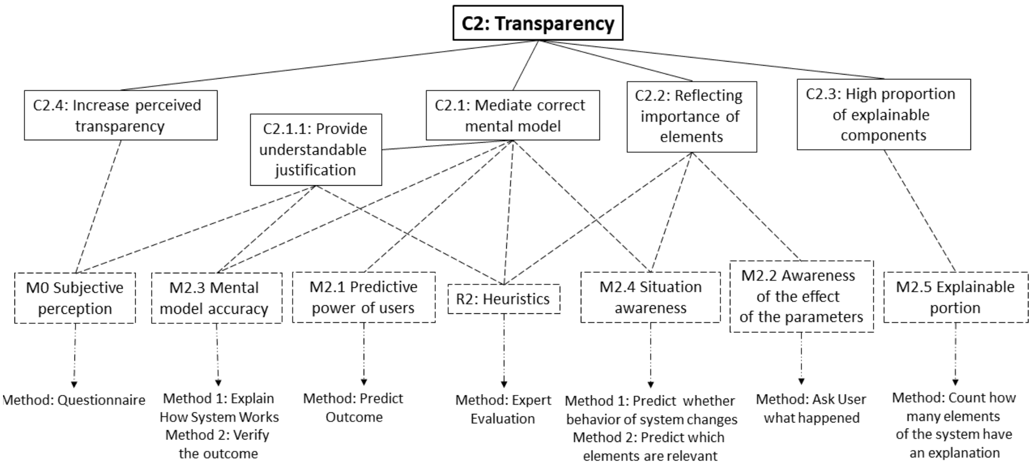

C2: Transparency
C2.4: Increase perceived transparency
C2.1.1: Provide understandable justification
C2.1: Mediate correct mental model
C2.2: Reflecting importance of elements
C2.3: High proportion of explainable components
M0 Subjective perception
M2.3 Mental model accuracy
M2.1 Predictive power of users
R2: Heuristics
M2.4 Situation awareness
M2.2 Awareness of the effect of the parameters
M2.5 Explainable portion
Method: Questionnaire
Method 1: Explain How System Works
Method 2: Verify the outcome
Method: Predict Outcome
Method: Expert Evaluation
Method 1: Predict whether behavior of system changes
Method 2: Predict which elements are relevant
Method: Ask User what happened
Method: Count how many elements of the system have an explanation

Figure 4.2: C2 – Transparency criteria and metrics

The second criterion to achieve good explainability is transparency. Systems that should satisfy this criterion are especially those that appear like a black box to the user – often systems with artificial intelligence. A real-world example are the algorithms that generate personalized advertising. As a user, it would be desirable to be able to find out what data these algorithms are based on, for instance, what data has already been collected about them and why certain advertisements are classified as appropriate.

Transparent explanations must provide enough insight into how the system works. It should be ensured that the correct mental model is conveyed and to achieve this, understandable justifications must be provided. It should also be ensured that the relevance of elements is reflected and that as many components of the system as possible can be understood.

Sub-criteria

C2.1 Increase Mental Model Accuracy: An explanation should help the user to better understand how the system works. Thus, the explanation should allow the user to create a correct mental model of the system. [22, 23, 33, 37, 42, 47, 51, 72, 77, 89] An important aspect of a correct mental model is to understand why the system performs certain actions. Understandable justifications are therefore particularly important in explanations (*C2.1.1 Providing understandable justification*). [10, 35, 51, 61, 84, 87]
C2.2 Reflecting Importance of Elements: The transparency of a system includes the ability to understand which elements influence the result and to what extent. The explanation should therefore reflect this importance or weighting of the parameters so that the user can understand what effects certain inputs have. [11]
C2.3 High Proportion of Explainable Components: Especially with automatically generated explanations, it is desirable that the system manages to generate these for as many components to be explained as possible. [16] In systems in which explanations are set statically by the developers, this criterion can be ignored, since in this case an explanation will appear at all places desired.
C2.4 Increase Perceived Transparency: Finally, the explanations should give the user the feeling that they have understood the system sufficiently. [15, 18, 25, 32, 38, 47, 83]

Metrics

To verify that the explanations convey a correct mental model, many papers suggest having participants predict what will happen next (*M2.1 Predictive power of users*). This can be either outcomes, or functions or actions of the system. [2, 22, 23, 33, 37, 40, 57, 84, 88] In order to evaluate whether the user understands the effect of all parameters, the metric M2.2 (*Awareness of the effect of parameters*) can be used. The first method would be asking participants to predict whether the behavior of the system would change when a certain parameter changes a certain way [49]. Another possibility would be to ask the participants to state which elements are particularly relevant for the algorithm [80, 88]. Mental model accuracy and whether understandable justifications were provided can be measured with metric M2.3 (*Mental Model Accuracy*). Participants can for example be asked to explain how the system works to the study investigator [33, 57, 77, 89]. Another method is to present participants with either a typical or an atypical system behavior and ask them to judge whether or not the behavior would occur that way [29, 33, 49, 57, 88]. To assess whether the explanation allows the users to be aware of the situation, the participants are shown a scenario and asked to explain what happened (*M2.4 Situation Awareness*) [88, 90, 94]. If the system generates the explanations automatically, it might be useful to count how many elements of the system – where it seems reasonable – have an explanation (*M2.5 Explainable Portion*) [16, 89]. The perceived transparency can be examined using a questionnaire (see table A.2 in the appendix).

4.2.3 Effectiveness

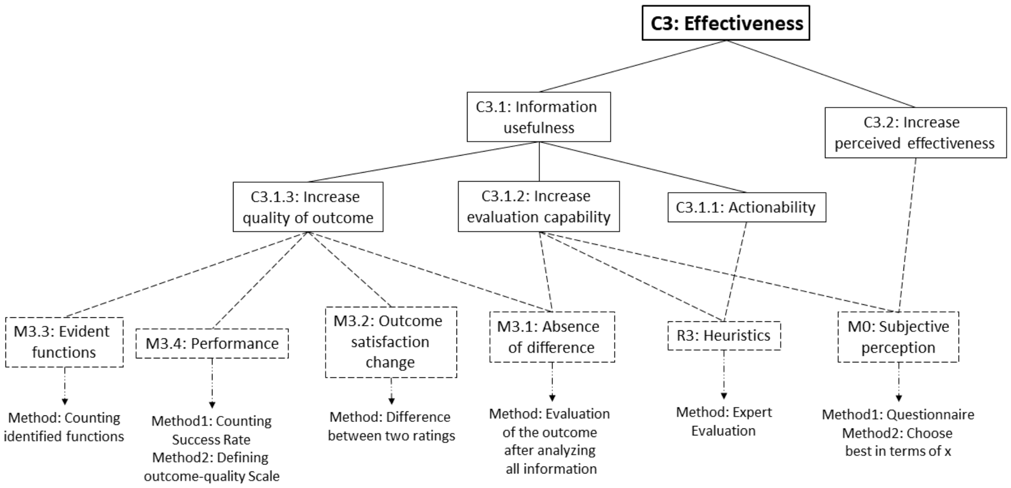

C3: Effectiveness
C3.1: Information usefulness
C3.2: Increase perceived effectiveness
C3.1.3: Increase quality of outcome
C3.1.2: Increase evaluation capability
C3.1.1: Actionability
M3.3: Evident functions
M3.4: Performance
M3.2: Outcome satisfaction change
M3.1: Absence of difference
R3: Heuristics
M0: Subjective perception
Method: Counting identified functions
Method1: Counting Success Rate
Method2: Defining outcome-quality Scale
Method: Difference between two ratings
Method: Evaluation of the outcome after analyzing all information
Method: Expert Evaluation
Method1: Questionnaire
Method2: Choose best in terms of x

Figure 4.3: C3 – Effectiveness criteria and metrics

Effectiveness can also be an important criterion for explainability. Especially in systems that can be classified as critical in some way, this criterion should be taken into account. An example would be systems in the health care sector. Here, a well-founded and considerate decision is absolutely necessary, since the health of people is to be protected in all case. It would not be as important for the explanations to be particularly easy to understand, as the user group is very educated in the domain.

An effective explanation should help the user to use the system better. Depending on the system, this could mean making better decisions, using features that meet their needs better, etc. To achieve this, the usefulness of the information is important. This includes that the information increases the quality of the outcome, improves the assessment of the situation and is actionable. Furthermore, according to Hernandez-Bocanegra and Ziegler [32], it is beneficial for the effectiveness if the explanations are interactive.

Sub-criteria

C3.1 Usefulness: An explanation should contain adequate information to allow the user to make a sound decision. [27, 32, 58, 59, 88, 93] One aspect of the usefulness of an explanation is that it is actionable (*C3.1.1 Actionability*). In other words, it should enable the user to react to it. An example to illustrate actionability would be a system for applying for a loan. The system tells the user that he is not creditworthy because he does not have a regular income and therefore needs a guarantor. The user can react to it and now look for a job or a guarantor. [22, 72, 77, 88] Another aspect of criterion C3.1 is the capability to evaluate the outcome or recommendations of the System (*C3.1.2 Increase evaluation capability*). This criterion is best applied to recommender systems. Here, users should be able to evaluate the recommendations and decide whether they are appropriate or inappropriate. [6, 21, 67] Finally, the usefulness of the information is also related to the quality of the outcome (*C3.1.3 Increase Quality of Outcome*). With an explanation that contains adequate information, the user is enabled to achieve the best possible results. [10, 27, 30, 61, 63, 67, 68, 73, 84, 85, 93]
C3.2 Increase Perceived Effectiveness: Lastly, the user should feel supported by the explanation to make the best possible decision. [15, 20, 38]

Metrics

To check whether the explanations improve the quality of outcome, four different metrics were found that can be selected depending on the system type. For example, if a system helps users to make decisions, metric M3.1 (*Absence of a difference*) can be used to check whether the user has made the best possible decision. The participants are first asked to make a decision with the help of the explanations, then all needed information is to be considered (for example, when deciding on a movie, the trailer would be watched). Finally the user evaluates again whether they would decide in favor of this decision. [1, 5, 6, 21, 27, 59, 64, 67, 84, 85] Another possibility is to examine the differences between two satisfaction ratings (*M3.2 Outcome Satisfaction Change*). The participants first use the system without and then with explanations and evaluate the outcome they have generated. If the ratings are better with explanations, these increase the quality. [15, 27, 84] The third metric M3.3 (*Evident functions*) can be used if the explanations are particularly intended to help the user to choose the right functions in order to increase the quality of use. The number of functions identified by the user are counted here. [54] Finally, the performance of the users can be measured (*M3.4 Performance*). For this purpose, either successful/unsuccessful outcomes must be defined and are then counted [78, 85, 90], or an outcome-quality scale must be developed with which the performance of the users can be evaluated [22, 57, 61, 73, 82]. Questions to evaluate the perceived effectiveness and the capability of evaluation are gathered in table A.3 in the appendix.

4.2.4 Efficiency

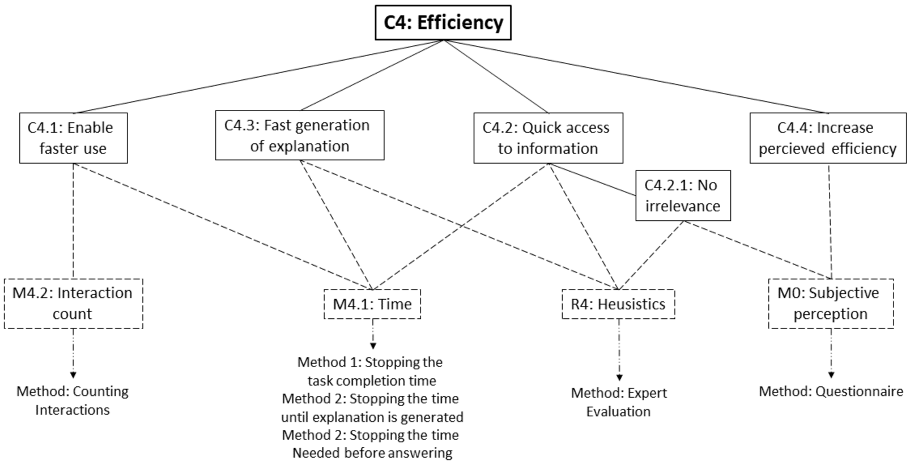

C4: Efficiency
C4.1: Enable faster use
C4.3: Fast generation of explanation
C4.2: Quick access to information
C4.4: Increase percieved efficiency
C4.2.1: No irrelevance
M4.2: Interaction count
M4.1: Time
R4: Heusistics
M0: Subjective perception
Method: Counting Interactions
Method 1: Stopping the task completion time
Method 2: Stopping the time until explanation is generated
Method 2: Stopping the time Needed before answering
Method: Expert Evaluation
Method: Questionnaire

Figure 4.4: C4 – Efficiency criteria and metrics

Efficiency is another objective which can be achieved through explainability. An explanation that targets efficiency should help the user to use the system faster. This could mean, for example, that the user can make decisions faster, or that he can perform actions in the system faster. An example scenario of the need for efficient explainability would be a tool for helping tax returns. This is a system that is used reluctantly and should be completed as quickly as possible. So the explanations of the system should support the user to get the tax return done as fast as possible.

An efficient explanation should therefore provide quick access to information (C4.2) and thus not contain irrelevant information (C4.2). In addition, the system should enable faster use (C4.1) and, as needed, increase perceived efficiency (C4.4). In order not to slow down the speed of use, it is also important for this objective that the generation of explanations is fast (C4.3).

Sub-criteria

C4.1 Enable faster Use: Explanations should contain information that enables the user to use the system more quickly. This can be, for example, explanations stating why some inputs are redundant or irrelevant in the specific situation, thus saving the user unnecessary inputs. Or explanations that help users to make decisions faster. [10, 19, 27, 30, 47, 52, 63, 82, 84, 85]
C4.2 Quick Access to Information: The explanations must also be designed in such a way that the user can quickly absorb the information. For example, long texts or graphics with many elements would not be suitable for this. [11] Part of this criterion is that the explanations do not contain irrelevant information that only cost the user time to consider (*C4.2.1 No irrelevance*). [39, 88]
C4.3 Fast Generation of Explanation: To ensure that the explanations do not have the opposite effect of efficiency by slowing down the system, it is necessary that the generation of the explanations does not take a noticeable amount of time. [11]
C4.4 Increase Perceived Efficiency: Lastly, it is important that the user also gets the impression that the explanations make the system faster to use. [80]

Metrics

Measuring time can be used to evaluate many efficiency criteria (*M4.1 Time*). The time it takes users to complete a specified task evaluates criterion C4.1, the time it takes to generate an explanation assesses criterion C4.3 and the time it takes a user to comprehend the explanation and then answer a question estimates criterion C4.2. [2, 5, 15, 18, 25, 27, 47, 49, 50, 61, 82, 84] To count the interactions with the system can also estimate whether the user is enabled to faster usage (*M4.2 Interaction Count*) [61, 84]. Perceived efficiency can be measured with questions from table A.4 in the appendix.

4.2.5 Satisfaction

The next criterion that is considered is satisfaction. An explanation should increase the comfort of use and in no way interfere or disturb it. A system where this criterion is particularly important are shopping websites. This system is in everyday use, and is used by all kinds of people. Most of the

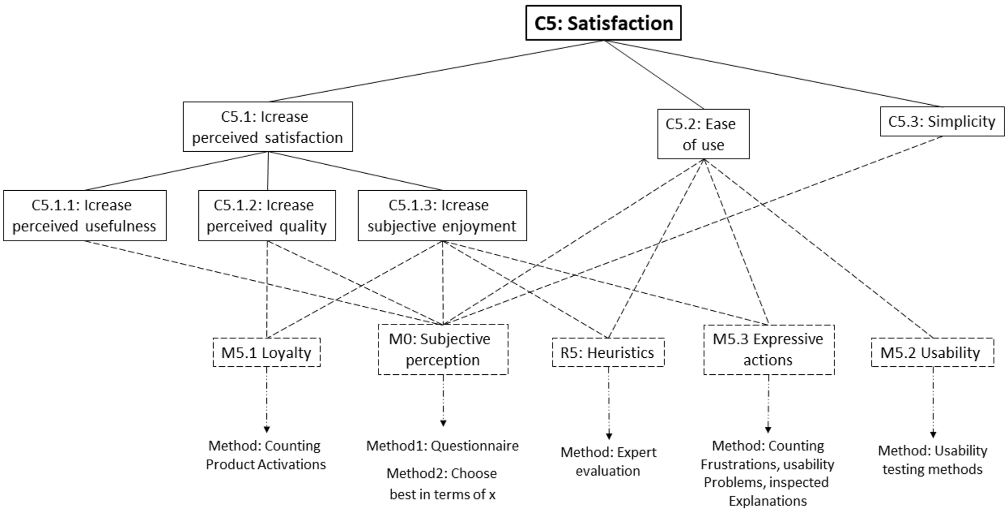

C5: Satisfaction
C5.1: Icrease perceived satisfaction
C5.2: Ease of use
C5.3: Simplicity
C5.1.1: Icrease perceived usefulness
C5.1.2: Icrease perceived quality
C5.1.3: Icrease subjective enjoyment
M5.1 Loyalty
M0: Subjective perception
R5: Heuristics
M5.3 Expressive actions
M5.2 Usability
Method: Counting Product Activations
Method1: Questionnaire
Method2: Choose best in terms of x
Method: Expert evaluation
Method: Counting Frustrations, usability Problems, inspected Explanations
Method: Usability testing methods

Figure 4.5: C5 – Satisfaction criteria and metrics

users would not want to spend time to get familiar with the system. However, if the system has a bad user experience, this would reflect negatively on the vendor and the customer might choose another website to do their shopping. Besides, if the system is particularly enjoyable to use, this can have a positive effect on the company's profits.

Satisfaction is mainly about the user's perception. This can involve the subjective usefulness, the subjective enjoyment, or the perceived quality of the system. Satisfactory explanations should also be easy to use and have a high level of simplicity. Furthermore, according to Sokol and Flach [77], interactive explanations are rated as more satisfactory by the user.

Sub-criteria

C5.1 Increase Perceived Satisfaction: The most important sub-criterion is the perceived satisfaction. [2, 14, 15, 30, 54, 73, 95] Since the concept of satisfaction is always subjective, the word "perceived" is almost redundant in this context. However, it has been included for consistency reasons. Three other criteria belong to this sub-criterion. Perceived usefulness requires that the explanation contains new and interesting information and is therefore perceived as useful (*C5.1.1 Increase perceived usefulness*). [15, 22, 30, 45, 52, 57, 64, 77, 78, 84, 88, 89, 92, 95] In addition, an explanation should make the user feel that the quality of the system or functions is high. Thus, the explanations should be designed in such a way that the user's good impression of the system increases (*C5.1.2 Increase perceived quality*). [10, 15, 27] Finally, it is necessary that the explanations

increase the subjective enjoyment. It is particularly important that the common use of the system is not disturbed by explanations (*C5.1.3 Increase perceived enjoyment*). [6, 10, 32, 44, 47, 63, 92]
C5.2 Ease of Use: How easy an explanation is to use is closely related to the well-known concept of usability. In general, the higher the usability, the easier it is to use the system. Therefore, this aspect is also important regarding the design of the explanations. This means both a possible interaction with the explanation and the appropriate embedding of an explanation in the overall system. [30, 51, 52, 62, 63, 71, 80, 84]
C5.3 Simplicity: According to a study by Habers et al. [31], explanations with fewer elements are preferred by most users. Other authors also recommended simple explanations when possible. Especially the negative effects of too long or too complicated explanations should be avoided for a satisfying explanation. [11, 22, 77, 88]

Metrics

Loyalty can be used to measure how satisfied the user is with the system (*M5.1 Loyalty*). If the user perceives the system as either useless or of low-quality, or does not find it enjoyable to use, then it is very unlikely that the user will continue to use the system. Conversely, if many users continue to use the product, then these three criteria can not be very poorly met. [25, 84] As discussed above, usability also plays a role in explainability. For this purpose, standardized usability testing methods can be used for evaluation (*M5.2 Usability*). [57, 84] In addition, expressive actions can be counted, for example frustrations, usability problems or interactions with explanations (*M5.3 Expressive actions*). Depending on the particular actions chosen, this can be used to measure how easy the system is to use with the help of the explanations or how much the user enjoyment increases as a result of the explanations. [25, 84] All sub-criteria of satisfaction can be measured with the help of questionnaires. Many of these questions are compiled in table A.5 in the appendix.

4.2.6 Correctness

As mentioned at the beginning of this chapter, correctness is a criterion that should be maintained regardless of the goal of the system. This includes the accuracy of an explanation. For example, an explanation may be particularly accurate in terms of the generalizability of the explanation (complete) or particularly accurate in terms of its applicability to the underlying

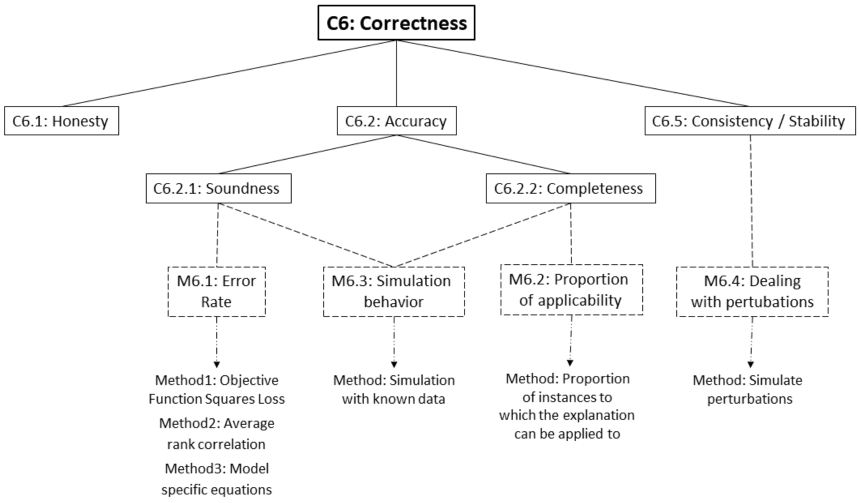

C6: Correctness
C6.1: Honesty
C6.2: Accuracy
C6.5: Consistency / Stability
C6.2.1: Soundness
C6.2.2: Completeness
M6.1: Error Rate
M6.3: Simulation behavior
M6.2: Proportion of applicability
M6.4: Dealing with pertubations
Method1: Objective Function Squares Loss
Method2: Average rank correlation
Method3: Model specific equations
Method: Simulation with known data
Method: Proportion of instances to which the explanation can be applied to
Method: Simulate perturbations

Figure 4.6: C6 – Correctness criteria and metrics

model (sound) or, at best, both. However, when a decision must be made between the two, it has been argued that soundness should be preferred. [77] Either way, it is important that the explanations are consistent with each other. If some information is omitted for simplicity, it should be done in a consistent and controlled manner – similar inputs should always lead to similar results – in this case explanations. Finally, an explanation should always be honest and not support bad intentions.

Sub-criteria

C6.1 Honesty: Honesty is a criterion that is very difficult to verify, and therefore the only criterion for which no metric has been found in the literature. It depends on the moral values of those who implement the explanations, and therefore can only be answered with certainty by the developers. The key point of honesty is that no false information is intentionally given, meaning that dark patterns, as Langer et al. [51] call them, are eliminated.
C6.2 Accuracy: Explanations should be accurate, meaning that the distance between the explanation and the model being explained should be as small as possible. [7, 11, 50, 57] There are two points of view from which the accuracy can be considered. On one hand, explanations should be truthful with respect to the model, meaning that the explanation should be correct for that exact situation (*C6.2.1 Soundness*). [55, 77, 88] On the other hand, an explanation should be generalizable. That means, an explanation should

not allow any false conclusions when applied to other situations, so that confusion is prevented (*C6.2.2 Completeness*). [22, 55, 77, 88]
C6.3 Consistency and Stability: The last sub-criterion concerns ensuring that explanations are consistent and stable, regardless of their accuracy. Similar situations should generate similar explanations. At best, even with different underlying models – i.e. across different systems – which however would require a definition of explainability standards. This point belongs to the criterion of correctness, since this is automatically given for accurate explanations, and for inaccurate explanations at least this sub-criterion must be fulfilled. [7, 50, 77, 88]

Metrics

The first metric that can be used to measure soundness is the error rate (*M6.1 Error Rate*). This metric includes model specific methods to measure how often an explanation is wrong. In the literature, this metric was used for AI systems. Even though there is no universal approach, this metric is an important starting point for real-world measurement of explainability in terms of correctness. [24, 50, 77, 93] The completeness can be measured using metric M6.2 (*Proportion of applicability*). This counts the number of elements / instances to which an explanation can be applied to. [1, 21, 77] Consistency can be estimated by simulating perturbations. For this, insignificant perturbations are added to the input and the extent to which the explanation (for example, a heatmap) changes is measured (*M6.4 Dealing with perturbations*) [88].

4.2.7 Suitability

Suitability is particularly important for systems that are used in a special context, for special user groups, or for special goals. The suitability criterion ensures that the system performs appropriately in these specific circumstances. An example where this criterion is particularly important is a tool to assemble custom computers. This tool will have very different user groups. On one hand, there is the user group of inexperienced novices who just want to quickly assemble a satisfactory computer. On the other hand, there are very experienced users who are well versed in computer components. In the group of inexperienced users, the explanations should be more goal-oriented and basic. That is, an explanation will rather explain what effects an SSD hard drive has compared to an HDD hard drive. Explanations for the experienced user group, in contrast, do not need this kind of explanation as they are more interested in the exact chip types or similar.

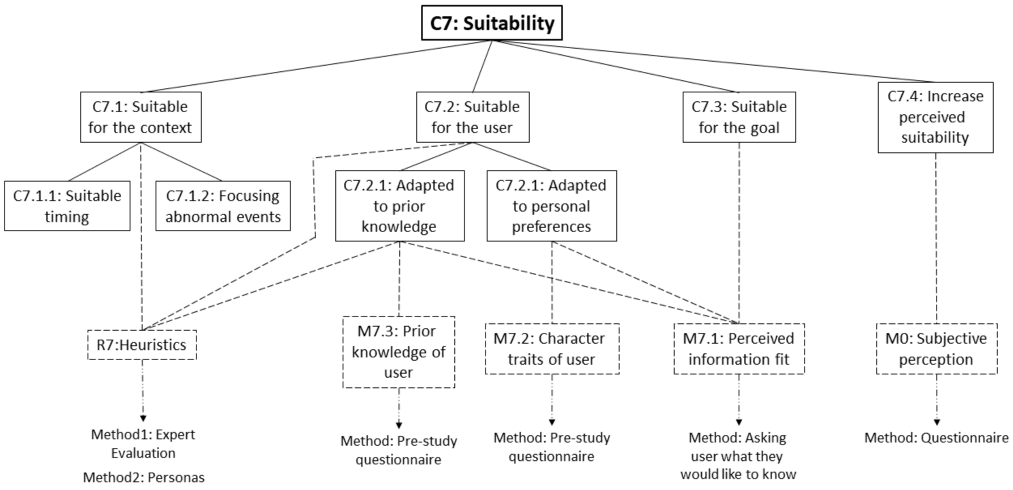

C7: Suitability
C7.1: Suitable for the context
C7.2: Suitable for the user
C7.3: Suitable for the goal
C7.4: Increase perceived suitability
C7.1.1: Suitable timing
C7.1.2: Focusing abnormal events
C7.2.1: Adapted to prior knowledge
C7.2.1: Adapted to personal preferences
R7:Heuristics
M7.3: Prior knowledge of user
M7.2: Character traits of user
M7.1: Perceived information fit
M0: Subjective perception
Method1: Expert Evaluation
Method2: Personas
Method: Pre-study questionnaire
Method: Pre-study questionnaire
Method: Asking user what they would like to know
Method: Questionnaire

Figure 4.7: C7 – Suitability criteria and metrics

Suitability can be briefly divided into three sub-areas. The explanations should be adapted to the context, which also includes that explanations appear at the right timing and focus on abnormal events. The second part is the adaptation to the respective user. Among other things, prior knowledge and personal preferences are significant here. The third part refers to the goal the system or user is aiming for. Finally, as with any main criterion, the perception of the user is relevant, which indicates whether they feel that the program is adapted to them and the circumstances.

Sub-criteria

C7.1 Suitable for the context: Systems are not always used in the ideal context of an office on a desktop screen with a mouse and keyboard. Some systems have special circumstances and may even change their context of use. Therefore, in some systems, it might make sense to require that the explanations are tailored to the context of use. [11, 24, 51, 77, 89] This includes that in certain contexts other timing would be appropriate at which an explanation is triggered (*C7.1.1 Suitable timing*). [33, 54] Furthermore, it is important to focus on events that are abnormal for this context, since these are usually the situations in which the user is confused and therefore needs explanations (*C7.1.2 Focus on abnormal events*). [11]
7.2 Suitable for the User: Software systems often have different user groups. Generally, it is important that software systems are adapted to these user groups, which also applies to explanations. [11, 39, 44, 61, 66, 72, 77, 85, 89] The differences between the user groups can concern all kinds of aspects. Two

things are frequently addressed in the literature. The first one is the prior knowledge of the users, which can refer to the system itself, i.e. how often such a system has been interacted with, as well as the prior knowledge in the domain in general. An explanation should be adapted to the prior knowledge of the users, so that it is neither too complicated nor repeats already known information (*C7.2.1 Adapted to prior knowledge*). [14, 39, 55, 77, 88, 89] The second aspect is the user's preferences. These depend on many factors and can therefore usually not be determined in advance by the developer. The preferences must consequently be determined by the user himself while usage, and a program that is supposed to contain particularly suitable explanations should allow the user to make such an adjustment. This includes for example what kind of explanations a user prefers – visual / textual / etc. (*C7.2.2 Adapted to personal preferences*). [45, 62]
C7.3 Suitable for the Goal: Finally, explanations should be adapted to the user's goal. This can be achieved, on one hand, by the customer specifying a goal in advance and then supporting this goal. On the other hand, this goal could also be individually defined by the user, so that explanations are adapted according to the specified goal of the user. [14, 22, 33, 39, 55, 89]
C7.4 Perceived Suitability: Perceived suitability, in this case, requires that the explanations are more likely to lead a user to believe that the system is adapted to him, his goal, and the circumstances. [87]

Metrics

In order to check whether the explanations are adapted to the personal preferences of the user, two metrics can be combined. The first step is to conduct a pre-study questionnaire that captures certain traits of the participants (*M7.3 Character traits of users*). [44, 45, 62, 82, 90] Then, the user is asked directly what information they would personally like to know (*M7.1 Perceived information fit*). In this way, it is possible to identify which information is missing for which user groups. By examining the information requested by the user, it is also possible to determine whether the information provided is appropriate to the user's goal (C7.3). [54] Similar to M7.2, prior knowledge can also be assessed by a pre-study questionnaire (*M7.3 Prior knowledge of user*). [71, 82, 91] It is also necessary to combine this metric with another metric (e.g. M7.1) in order to link information and to identify which user groups – in terms of prior knowledge – need which kind of information. A few questions on the suitability criterion of explainability can be found in the appendix in table A.6.

4.2.8 Trustability

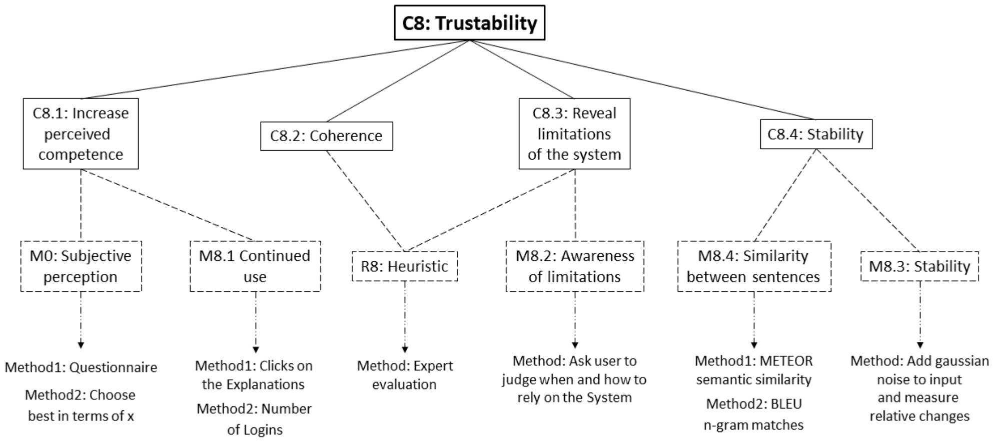

C8: Trustability
C8.1: Increase perceived competence
C8.2: Coherence
C8.3: Reveal limitations of the system
C8.4: Stability
M0: Subjective perception
M8.1 Continued use
R8: Heuristic
M8.2: Awareness of limitations
M8.4: Similarity between sentences
M8.3: Stability
Method1: Questionnaire
Method2: Choose best in terms of x
Method1: Clicks on the Explanations
Method2: Number of Logins
Method: Expert evaluation
Method: Ask user to judge when and how to rely on the System
Method1: METEOR semantic similarity
Method2: BLEU n-gram matches
Method: Add gaussian noise to input and measure relative changes

Figure 4.8: C8 – Trustability criteria and metrics

Trust is especially important in applications where the user does not have the ability to understand and verify all operations, and therefore must rely on trusting the system that it has everything under control. An example of this would be a simple navigation system. When driving a car, it is not possible for the user to accurately check the route and understand how correctly the route has been calculated. Therefore, especially in the case of unexpected outcomes (e.g., if a roadwork site is on the usual route), it should briefly but trustworthily explain to the user why a different route than usual has been chosen. This allows the user to trust the system's route and drive the best possible way.

Trustworthy explanations should increase the perceived competence of the system and at the same time clarify the limits of the system to enable the user to evaluate when the system can be trusted and when it cannot. This leads to even more trust in the system in the right situations. In addition, it is important that the explanations are coherent and stable so that a user does not become confused and thus distrustful.

Sub-criteria

C8.1 Increase perceived competence: The user's perceived competence has a significant influence on the level of trust he can have in the system. If a user thinks that a system works poorly, he cannot trust it. Therefore, an explanation that helps the user to trust the system must increase the user's perceived competence. [6, 10, 11, 18, 25, 35, 45, 47, 62, 63, 65, 69, 84, 89, 91]

C8.2 Coherence: An explanation and thus the entire system can be trusted more if the statement of the explanation does not contradict the user's prior knowledge. If an explanation does not match the user's prior belief, he or she is more likely to distrust the system than to question his or her knowledge. [11, 55, 77]
C8.3 Stability: Similar to coherence, it is also important that expectations learned during the use are fulfilled. Similar situations (e.g. similar input) should produce similar explanations. Otherwise, users may doubt the explanations and thus not trust them as much. [11, 22, 88]
C8.4 Reveal limitations of the system: If the system's explanations reveal where the system's limitations are, and the user can therefore assess under which circumstances these limitations have not been exceeded, he can trust the system more in these circumstances. The user gets the feeling that he can estimate when the system might be wrong and trusts the results more in all other situations. [33, 57]

Metrics

Perceived competence can be assessed by observing how many users continue to use the system or the explanations. If most users are loyal to the system or use the explanations frequently, the perceived competence cannot be bad, otherwise they would use another better system or would not look at the explanations (*M8.1 Continued Use*). [81, 84] To assess whether the system is revealing its limitations, users can be asked to estimate when and how to rely on the system. If these values coincide with the actual limitations, the sub-criterion is well satisfied (*M8.2 Awareness of limitations*). [33] To check the stability of the explanations, Vilone and Longo [88] name the possibility of adding Gaussian noise to the input and checking whether explanations remain the same (*M8.3 Stability*). In addition, Vilone and Longo [88] also introduced similarity between sentences to measure how similar explanations are (*M8.4 Similarity between sentences*). Trust is a criterion that can be captured very well through questionnaires, as it is inherently subjective. Suggested questions can be found in the appendix in table A.7.

4.2.9 Persuasiveness

Closely linked to trust is the criterion persuasiveness. If trust in the system is high, it is easier to convince the user of an intention. However, it is important to ensure that no dark patterns are used, meaning that the user is not persuaded to decide for something he does not want after all.

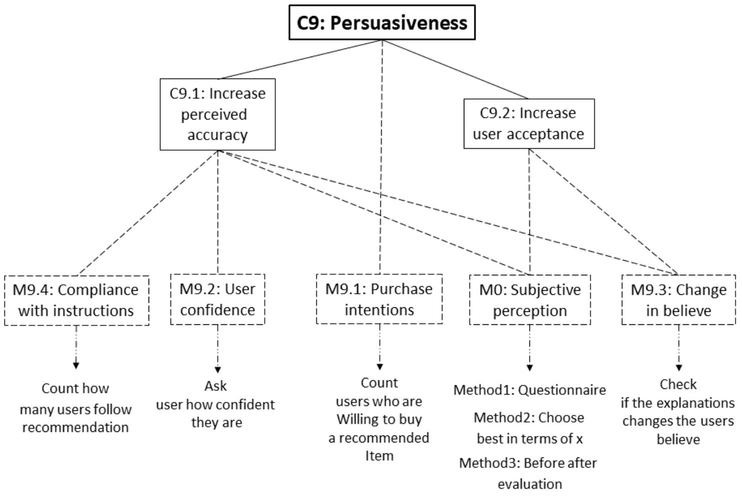

C9: Persuasiveness
C9.1: Increase perceived accuracy
C9.2: Increase user acceptance
M9.4: Compliance with instructions
M9.2: User confidence
M9.1: Purchase intentions
M0: Subjective perception
M9.3: Change in believe
Count how many users follow recommendation
Ask user how confident they are
Count users who are Willing to buy a recommended Item
Method1: Questionnaire
Method2: Choose best in terms of x
Method3: Before after evaluation
Check if the explanations changes the users believe

Figure 4.9: C9 – Persuasiveness criteria and metrics

Persuasion is especially important in systems where products are suggested. For example, an online e-commerce like Amazon wants to show the user with explanations why this product was suggested and in the best case convince him that the product is the right one for him.

Convincing explanations should increase the usage and purchase intention. To this end, user acceptance should be increased, since users can only be convinced if they also accept the explanation. Furthermore, the perceived accuracy is an important factor, which also shows how closely the criteria trust and persuasiveness are connected, as perceived accuracy is similar to perceived competence (C8.1).

Sub-criteria

C9.1 Increase perceived accuracy: The explanations in a system that is supposed to be particularly persuasive must be designed in such a way that the user is convinced that the system has a high accuracy. In this way, the user is more likely to rely on the system's recommendation and be persuaded. As the perceived accuracy of the system increases, so does the user's confidence in the respective matter. [32, 44, 45, 62, 82]
C9.2 Increase user acceptance: Similar to the first sub-criterion, this point aims at the user accepting the outcome or the explanation. [10, 21, 27, 30, 35, 38, 39, 62, 90, 95] This is more likely to happen if the user believes that the accuracy of the system is high (C9.1). The difference to the first point is that it is not absolutely necessary for the system to perform well, since it can also gain the user's acceptance with the help of emotional argumentation.

Metrics

If the user is to be convinced by the system to buy elements, the purchase intentions are a good way to measure the persuasiveness of explanations. If more users are willing to buy after an explanation than without it, the persuasiveness of the explanation is fairly high (*M9.1 Purchase intention*). [2] To check how accurately users perceive the system through the explanations, users can be asked how confident they are about certain matters. If the persuasiveness of the explanations is high, the user is convinced and more confident about the respective matters after he received the explanation (*M9.2 User confidence*). [78, 95] Another way is to count how often users follow the instruction given in the explanation. For example, if products are recommended, how often they click on these products to take a closer look at them, etc. (*M9.3 Compliance with instructions*). [41, 84, 90] To assess whether the explanation increases the user's acceptance, it can be tested whether it changes the user's belief. If the user changes his mind after the explanation, it is persuasive (M9.3 Change in belief). [95] Questions on the evaluation of persuasiveness are given in the appendix in table A.8.

4.2.10 Scrutability

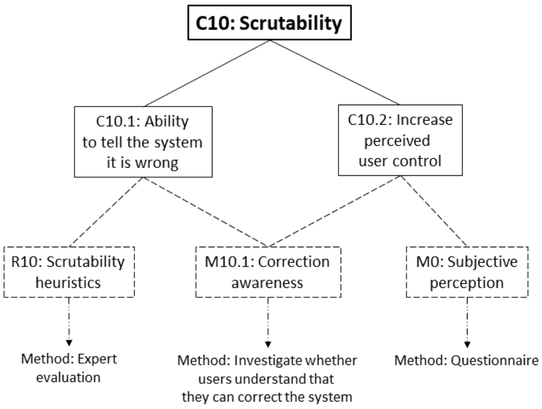

C10: Scrutability
C10.1: Ability to tell the system it is wrong
C10.2: Increase perceived user control
R10: Scrutability heuristics
M10.1: Correction awareness
M0: Subjective perception
Method: Expert evaluation
Method: Investigate whether users understand that they can correct the system
Method: Questionnaire

Figure 4.10: C10 – Scrutability criteria and metrics

Systems that cannot always deliver the right outcomes, or that are based on user preferences in particular, should give the user the opportunity to tell the system that it is wrong. In those systems, explanations can help the user to recognize that the system is wrong and, at best, directly integrate a way to report the error. An example where this has not yet been sufficiently implemented but would be very helpful are streaming services. They suggest movies to the user based on movies they have already watched and explain what this suggestion is based on. Sometimes, however, it happens that a user receives suggestions based on a movie that he did not like at all or similar. In this case, a mechanism could be integrated directly into the explanation to allow the user to correct this misunderstanding.

Sub-criteria

C10.1 Ability to tell the system is wrong: The first sub-criterion includes two conditions. First, the content of the explanation must enable the user to recognize that the system is wrong, and second, the system must offer a way to report this based on this explanation. Overall, the user then has the ability to tell the system that it is wrong. [10, 63, 68, 84, 85, 88]
C10.2 Increase perceived user control: An explanation should also increase the perceived user control, indicating that the user is aware that he can control and correct certain aspects. [38, 62]

Metrics

To check whether the explanation enables the user to tell the system that it is wrong, it can be investigated whether the user is aware that he can influence or change the system (*M10.1 Correction awareness*). If the user is aware of this, the perceived useful control is usually also high. [84] In table A.9 in the appendix, questions can be found to assess the perceived user control.

4.2.11 Debugability

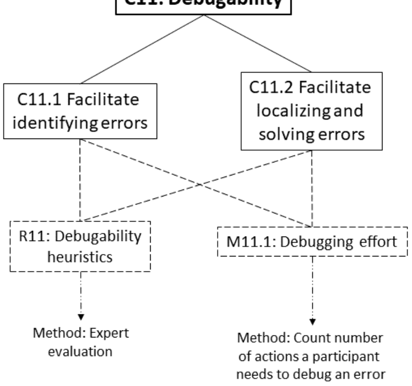

C11: Debugability
C11.1 Facilitate identifying errors
C11.2 Facilitate localizing and solving errors
R11: Debugability heuristics
M11.1: Debugging effort
Method: Expert evaluation
Method: Count number of actions a participant needs to debug an error

Figure 4.11: C11 – Debugability criteria and metrics

Debugability is the only criterion found in the literature that is mainly based on the developer's perspective. It should facilitate the process of identifying and fixing bugs in systems. A system that needs explanations that satisfy the debugability criterion is, for example, a search engine like Google Search. Google Search is a very complex system, which is worked on by many different developers, and where the developers also inevitably change over time. In such a complex system, it is difficult to keep track of all eventualities and, if a bug occurs, to find out where it came from. If such a system could now justify every action and at best also show the code locations, it would be much easier for the developers to find the code location that is responsible for the bug.

Sub-criteria

C11.1 Identify errors: Before a bug can be fixed, it must first be identified. The explanations should help the developer to recognize bugs, since this is not always immediately obvious from outcomes. [10, 35, 52, 63]
C11.2 Localize / solve errors: The error can then be corrected. The explanation should help the developer to find the code location that causes the error and thus also help to eliminate the error finally. [42, 52]

Metrics

For the debugability criterion, only one metric was found in the literature, which, however, covers both sub-criteria at the same time. This metric is very vague, but gives a rough framework by counting the number of actions a user needs to debug a part of the system (*M11.1 Debugging effort*). [47]

4.3 Ideas for Further Metrics

During the literature review, ideas for additional metrics to measure certain criteria of explainability emerged. However, these metrics are only first drafts and have not yet been confirmed with scientific methods. The metrics are presented along with the criteria they attempt to measure.

M5.4 Sentiment: The atmosphere of a text can have a great influence on the user's enjoyment. If an explanation is worded in such a way that users get a negative feeling when reading it, they will perceive its use as unpleasant. Since sentiment is something rather subjective, this evaluation is not trivial. However, there is already some research in the field of sentiment analysis, so these approaches could also be used in the analysis of explanations. This approach could be used to measure criterion *C5.1.3 Increase Subjective Enjoyment.*

M6.5 Code correctness: As already mentioned in the definition of criterion *C6.1 Honesty*, it is very difficult to verify. A very expensive method would be to check the code of the system, especially those parts where the explanations are generated. This procedure is possible in systems where the explanations are generated programmatically. In this way, it would be possible to check whether misleading or incorrect facts are included in the generation of the explanations. However, this is not possible for static explanations that were previously defined by humans.

M7.4 Explained abnormal events: In order to check whether all events that surprise the user are explained, the unexpected events must first be identified. The following procedure can be used for this purpose: First, all explanations are temporarily removed from the system, then the user is asked to perform a task that covers as much of the system as possible. As soon as an unexpected event occurs, the user marks this event (including a comment, if necessary). At the end, it is checked whether an explanation is provided in the original system at all points that were marked as unexpected for the user. This metric could be used to test criterion *C7.1.2 Focus on abnormal events.*

M11.2 Finding incorporated errors: One possibility that is already used in the field of software engineering, more precisely testing, is the incorporation of errors. Mutation tests are a possibility to check if test cases would find the inserted errors. A similar procedure would be conceivable with the evaluation of the explanations. So certain errors are built into the system, and then it is tested whether the developers better succeed with the explanations to find this error. This metric can be used to test criterion *C11.1 Identify errors.*

4.4 Conclusions

Several conclusions could be drawn from the literature review. First, upper criteria were defined to give a good overview of the aspects that constitute good explainability. These upper criteria were refined to provide more detailed insight into them and thus make them more measurable. For each of the sub-criteria, metrics from the literature were presented to measure the extent to which these criteria are met. Overall, a baseline has been established to enable an assessment of explainability.

However, the literature review has revealed a very important conclusion. The main criteria alone make it clear that the quality of explainability is fundamentally dependent on the purpose it is intended to serve. In general, explainability has a special nature compared to other NFRs (non-functional requirements). Explainability is not required for its own sake, but rather serves as a means to an end. [46] Unlike other NFRs such as security, availability, usability, etc., explainability is not required in order for the system to be particularly explainable. It is required so that users trust the system more (trustability) or so that users can use the system better (effectiveness) or so that a developer can debug the system better (debugability) or so on. The quality of explainability is thus inevitably linked to its purpose, or more precisely, to the objective it is intended to achieve. Two examples are given for illustration:
The first example is a system that recommends cars based on some attributes entered by the user. In order to make it understandable for the user why the system makes certain recommendations, the system provides explanations. This allows a user to determine exactly what the benefits of different recommendations are and to prioritize them appropriately. The use of the system may concern a lot of money and helps the user to make some kind of critical decision. Thus, the explanations for this system are designed to help the user make the best possible decision he can (effectiveness). Therefore, particularly detailed and complete explanations are required here.

The second example is a system that helps a user find a satisfactory recipe with the ingredients he has available. The user enters the ingredients he has at home, and the system recommends the recipes that are possible with those ingredients. It justifies its recommendations based on possible additional ingredients the user does not have available yet. Deciding on a recipe is more of a chore decision that needs to be made quickly than a critical decision. The explanations in the second system should therefore enable the user to quickly select a suitable recipe (efficiency). This requires short and pragmatic explanations that can be grasped at a glance.

Both systems need explainability likewise to justify the suggestions made by the system and to allow the user to make a reasonable decision. However, due to the objectives to be achieved by explainability, there are very different criteria for both explanations that make them good. Altogether, it can be seen, that explanations differ essentially in the criteria that make them good explanations, depending on their purpose. Since the evaluation of the explanations is based on the criteria, it is very important to consider this purpose, or more precisely the objective to be achieved.

Chapter 5

Concept of Evaluation of Explainability

In this chapter, a concept is developed to measure the explainability of software systems (partially) independent of user studies. Through extensive literature research, many criteria for good explainability have already been collected. However, the evaluation of these criteria was almost exclusively based on user studies. With respect to the goal of this thesis to develop a prototype that measures the explainability of systems, it becomes clear that user studies alone are not sufficient as a basis for this concept. For this reason, heuristics based on the criteria and metrics presented in chapter 4 were created to provide a preliminary assessment. For a well-founded assessment of explainability, the prototype will further present the metrics from the literature, so that, if appropriate, the results of the heuristics can be verified through user studies.

5.1 Development of Heuristics

Based on the sub-criteria, heuristics were developed to provide an initial assessment of explainability. As discussed in section 4.4, the evaluation of explainability depends on the objectives the explanations are trying to achieve. Thus, not all heuristics presented in the following section can simply be taken for evaluation, but must be selected to fit the system and its objectives to be achieved through explainability.

Table 5.1 shows three heuristics for the understandability criterion. Heuristic R1.1 refers to the fact that explanations should be as simple as possible, which refers to criterion C1.1.1. It is therefore desirable to avoid technical terms and to keep the language simple in general.

ID	Question	Based on
R1.1	The language is kept simple – it does not contain any technical words the target user does not understand.	C1.1.1 Simplicity
R1.2	Flesch Reading Ease Score	C1.1.1 Simplicity
R1.3	The elements in the explanation are logically coherent. It follows a red thread. There are no contradictions.	C1.4 Logically Coherent

Table 5.1: Heuristics based on understandability

Moreover, Vultureanu-Albişi and Bădică [89] argue that the quality of the explanation depends on the number of words or the word length. This criterion corresponds to an established method for measuring the complexity of texts – the Flesch Reading Ease score. It calculates a value based on the number of words, sentences and syllables. [26] Thus, the heuristic R1.2 is a good way to assess how easy a textual explanation is to understand. Finally, according to Vultureanu-Albişi and Bădică [89], sentences should have a logical relationship to each other, meaning that there are no contradictions and that there is some kind of red thread, as this increases the perceived understandability. This criterion is particularly important for longer explanations, since with very short explanations the risk of contradictions is very low and a red thread in the explanation is not so much needed. This is captured in heuristic R1.3.

ID	Question	Based on
R2.1	For each input parameter, it is clear why the system needs this input and what it is used for.	C2.1.1 Provide understandable justification, C2.2 Reflecting importance of Elements
R2.2	The role that the parameters the user enters have on the event being explained becomes clear.	C2.2 Reflecting importance of Elements
R2.3	It is clear which aspect the explanation targets.	C2.1 Mediate correct mental model

Table 5.2: Heuristics based on transparency

Heuristics based on the transparency of explanations can be found in table 5.2. According to Carvalho et al. [11] it is important that an explanation reflects the importance of features or parts of the explanation. For example, if certain inputs led to certain outputs, it must be apparent which inputs have the greatest influence on the outputs. In addition, Hunt and Price [35] mention that an explanation should clarify to the user why certain questions were asked – in other words, why input parameters are required. Based on this, the heuristic questions R2.1 and R2.2 were developed. Another important aspect of transparency is the mental model, which can be built up through the explanation. [47, 51, 72] This mental model should correspond as accurately as possible to the real model. Without user studies, this criterion is difficult to evaluate as a whole. However, a necessary criterion for this is whether it is clear to which aspect of the system the explanation refers(R2.3). Otherwise, the generation of a correct mental model will not be possible.

ID	Question	Based on
R3.1	The information given can help with the user's decision-making process if the user has not had this information before.	C3.1.1 Actionability, C3.1.2 Increase evaluation capability

Table 5.3: Heuristics based on effectiveness

The effectiveness of explanations is difficult to evaluate heuristically independent of user studies. Nevertheless, question R3.1 from Table 5.3 is an attempt to design a question that captures the usefulness of the information. The question is intended to guide the evaluator to consider whether the information contained in an explanation has any benefit at all, or is merely superfluous. However, the question depends on the evaluator being able to put himself in the position of the target group and to assess which information is needed in certain situations.

Table 5.4 shows heuristic questions regarding efficiency. In terms of this criterion, it is important that the user is able to quickly obtain the information that the explanation intends to convey. [11] For texts, this means that they should be kept as brief as possible (R4.1). Texts with several long sentences are rather unsuitable for efficient explanations. Second, for visualizations, this means that colors must be easily distinguishable from one another (R4.2). This ensures that information can be recognized at first glance. According to Kass and Finin [39], in order to make information quickly comprehensible, irrelevant facts should necessarily be omitted. This aspect is targeted by the heuristic question R4.3. For this heuristic, it is important that the evaluator, who has to assess this heuristic, is

ID	Question	Based on
R4.1	The Explanation is kept short. (if textual)	C4.2 Quick access to information
R4.2	The visualization uses colors that are easily distinguishable from each other. (if visual)	C4.2 Quick access to information
R4.3	The explanation does not contain any elements/information that are redundant or irrelevant. E.g. Duplicates should not occur.	C4.2.1 No irrelevance
R4.4	The generation of explanations seems immediate to the user – they do not feel like they are waiting	C4.3 Fast generation of explanation

Table 5.4: Heuristics based on efficiency

provided with examples in order to make a reasonable assessment. Examples for redundant or irrelevant elements would be information that is given twice (duplicates) and very obvious or self-explanatory functions. It is always important to consider what is irrelevant from the point of view of the target group and not from the point of view of the evaluator. Finally, especially in the field of automatically generated explanations in AI, it is important that the speed of the system is not slowed down by the generation of explanations. [11] The generation of the explanations should not take a long time (R4.4). The perfect rating of this heuristic would occur when the explanation is displayed immediately and there is no loading time.

ID	Question	Based on
R5.1	The explanation is easy to find.	C5.2 Ease of use
R5.2	The explanations are not disruptive and do not interfere with the general use.	C5.1.3 Increase subjective enjoyment, C5.2 Ease of use

Table 5.5: Heuristics based on satisfaction

With regard to satisfaction, many usability heuristics can be applied to explainability. Especially when pop-ups, dialogs or other custom UI elements are created for the explanations. However, two aspects that are particularly important are listed in table 5.5. Explainability functions should be easy to use according to Langer et al. [51] and to guarantee this they should be easy to find first of all. If the user has to search where to find an explanation, this can be very frustrating and consequently, the explanations might not be used at all. In addition, the enjoyment of using the explanations and the system itself should be increased. [44, 63, 92] Therefore, it should be ensured that explanations are not disruptive and do not interfere with general use (R5.2). It is important to note that both heuristics are only necessary requirements and not sufficient to fully evaluate the satisfaction of explainability.

ID	Question	Based on
R7.1	Short mental context analysis: can the explanations be grasped in the context in which they are displayed? – if possible, simulate the context of use and try to grasp the explanation.	C7.1 Suitable for the context
R7.2	For each target group of the system: Are the metaphors in the explanation understandable based on the cultural background?	C7.2 Suitable for the user
R7.3	For each target group of the system: Is the explanation understandable with the prior knowledge they have?	C7.2.1 Adapted to prior knowledge
R7.4	The explanations are adaptable to the user's level of prior knowledge.	C7.2.1 Adapted to prior knowledge

Table 5.6: Heuristics based on suitability

Heuristic questions regarding the suitability of explanations are presented in Table 5.6. An important aspect is the adaptation to the context. Therefore, it makes sense to ask the evaluator to perform a short mental context analysis. This means that he should quickly put himself mentally in the context in which the system is used. In the case of a navigation system, for example, the evaluator should imagine that he is sitting in a car as the driver. If possible, simulating this context is even better. Based on this, it should then be checked whether the explanation can be grasped in the specific context (R7.1). The second important aspect is the adaptation to the user. [11, 39, 77] An important issue of this, that is also often addressed in usability contexts, is the interpretation of icons or metaphors. If the system has many different user groups – for example, a website that is used across continents – it should be checked whether icons and metaphors are understandable for all users (R7.2). Another point regarding user groups are possible difference in prior knowledge. [14, 39, 89] In some situations, it therefore makes sense to make explanations adaptable to the user, especially to his or her level of prior knowledge (R7.4). In any case, the explanations shown to the user should be understandable for every user and thus for every level of prior knowledge, which is asked for in heuristic R7.3.

ID	Question	Based on
R8.1	The explanations are coherent with each other. They are related when possible and do not contradict each other in any case.	C8.2 Coherence
R8.2	Possible limitations of the system are revealed by the explanations.	C8.3 Reveal limitations of the system

Table 5.7: Heuristics based on trustability

Trust is the eighth main criterion, which was discussed in chapter 4. To assess this criterion, two heuristic questions were developed – see Table 5.7. Both Sokol and Flach [77] and Miller [55] stated that explanations must be consistent with users' prior beliefs in order to build trust. This also includes facts just learned from previous explanations. To prevent the user from becoming suspicious, explanations should be coherent with each other as required in heuristic R8.1 and should not contradict each other in any way. At best, they should even refer to each other. Furthermore, the limits of the system should be pointed out. [33, 57] This can help users trust the system more in the right situations, as they feel they know the system better. This aspect was included in heuristic R8.2, but it should be noted that not every system has limits that must be shown. This heuristic is therefore particularly appropriate for systems such as AI systems or similar, where complete certainty cannot be guaranteed.

ID	Question	Based on
R10.1	The explanation gives a direct possibility to report an error to the system, or states where/how to report this error.	C10.1 Ability to tell the system it is wrong

Table 5.8: Heuristics based on scrutability

To assess the scrutability criterion, one heuristic was found (see table 5.8). Many authors affirm that it is important for certain systems that the explanations allow the user to tell the system that it is wrong [63, 68, 84, 85]. At best, an option should be provided in the explanation that allows the user to report such an error directly. Otherwise, it should at least become clear where the error can be reported. This requirement is covered by heuristic R10.1.

Finally, table 5.9 presents heuristics for the debugability criterion. Since this criterion is required only in very special cases, it is assumed that these heuristics are evaluated by developers who have some experience with

ID	Question	Based on
R11.1	The explanations give the developer the possibility to check if everything is processed correctly or if there is a bug.	C11.1 Facilitate identifying errors
R11.2	The explanations show the developer in which code area the bug is created.	C11.2 Facilitate localizing / solving errors

Table 5.9: Heuristics based on debugability

debugging. The criterion is divided into two aspects. The first aspect that heuristic R11.1 addresses is that explanations should help developers identifying errors. [35, 52] This means that the explanations should provide more insight into the processes than the outputs, since otherwise the errors could simply be identified directly using the outputs. The second aspect is to locate and solve these bugs. [42, 52] If the explanations give clues to the part of the code or training set where the bug is produced, this helps a lot in solving them (R11.2).

5.2 Process of Metric Selection

As demonstrated in section 4.4, it is not possible to assess explainability independently of the objective it is intended to achieve. In the process of evaluation, therefore, it must first be determined which objective is to be achieved with explainability. Depending on the results, appropriate metrics can be presented. Since these metrics are almost all based on user studies, heuristics were defined above to provide an initial assessment of the explainability. These heuristics were mapped to the eleven criteria in the same way as the metrics, so that they are also linked to the corresponding objectives. Before the actual evaluation begins, the user is first asked questions aimed at finding out the objective to be achieved with explainability for this system. In addition, there are general questions to be answered about the system. For example, textual explanations can be evaluated differently than visual ones. And automatically generated explanations have different priorities (e.g. heuristic R4.4) than explanations that are static and only need to be displayed. So not all defined heuristics and not all found metrics are suitable for every type of system. These general questions are asked as soon as an objective is selected, for which this question must be clarified. The implementation of the questions that are asked before the actual evaluation of the system is shown in table 5.10. For clarification, the questions relating to the objectives are bolded and the general questions about the system are indented under the corresponding objective.

	Question	**Added metrics and heuristics**
C1	**Should the explanations be particularly easy for the user to understand?**	M0, M1.1, M1.2, M1.3
	Are the explanations (partially) presented in textual form?	R1.1, R1.2
	Are the explanations (partially) presented in visual form?	M1.3.1, R1.3
C2	**Should the explanations be used to understand the inner workings of the system?**	M0, M2.1, M2.3, M2.4, R2.3
	Does the system contain parameters that are controlled by the user and that are relevant for the events to be explained?	M2.2, R2.1, R2.2
C2	**Should the explanations aim to help the user use the system better? (e.g. make better decisions, use better functions)**	M0, M3.2, M3.4
	Is the system a recommender system?	M3.1, R3.1
C4	**Should the explanations be aimed at enabling the user to use the system more quickly?**	M0, M4.1, M4.2, R4.1, R4.2, R4.3
C5	**Should the explanations increase the ease of use, making the overall experience with the system more enjoyable?**	M0, M5.2, M5.3, R5.1, R5.2
C7	**Should the explanations be designed to allow the system to adapt to specific users, contexts, and/or usage goals?**	M0, M7.1
	Does the system contain use cases that take place in special environments?	R7.1
	Is the system targeted at very diverse user groups?	M7.2, M7.3, R7.2, R7.3, R7.4
C8 C9	**Should the explanations serve to induce trust in the user, or to persuade them?**	M0, M8.1, M8.2, M9.2, M9.3, M9.4, R8.1, R8.2
C10	**Should the explanations allow the user to tell the system that it is wrong?**	M0, M10.1, R10.1
C11	**Should the explanations help developers to debug the system?**	M11.1, R11.1, R11.2

Table 5.10: Pre-assessment questions and included metrics/heuristics

In this table, only nine questions are asked to specify the objectives, although they cover all eleven main criteria. The reason for this is as follows: Correctness was classified as always relevant, since dark patterns were ignored, and is therefore not included in the questions for the objective. Trustability and Persuasiveness focus on very similar objectives, and are also closely related overall. Therefore, these two aspects are merged and asked for as one. In total, there are nine different objectives to choose from, which can also be combined with each other.

Finally, based on the answers to all these pre-questions, a choice of metrics and heuristics is made, which are tailored to the system and the objectives to be achieved with explainability.

5.3 Implementation of the Prototype

In this section, the implementation of the prototype is presented. For simplicity, this prototype is referenced here and below as *Explainability Meter*. The steps involved in the evaluation are shown in figure 5.1.

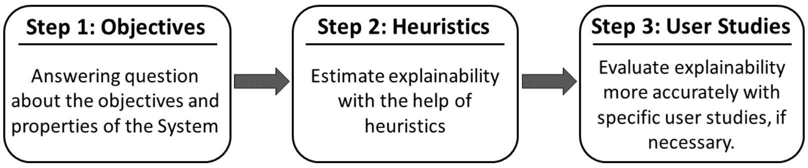
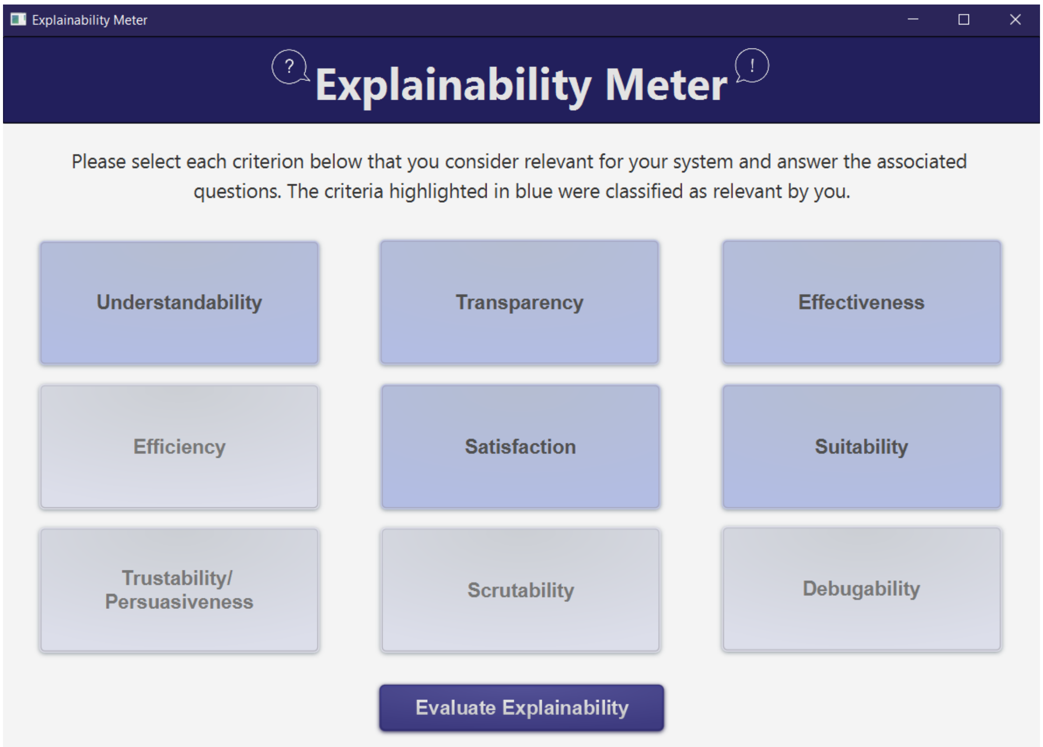

Step 1: Objectives
Answering question about the objectives and properties of the System
Step 2: Heuristics
Estimate explainability with the help of heuristics
Step 3: User Studies
Evaluate explainability more accurately with specific user studies, if necessary.

Figure 5.1: Steps for the evaluation of explainability

The first step is to determine the objective to be achieved with explainability as described above. Based on this, the heuristics and metrics for steps 2 and 3 are compiled. In the second step, the user evaluates heuristics to get a first estimation of explainability. Since the heuristics only provide an estimate, it is sometimes useful to generate additional reliable assessments from user studies. For example, if the heuristics identify an aspect as very poor, this aspect could be re-evaluated with the help of user studies before major changes are made.

Selection of objectives (step 1)

Any of the above-mentioned objectives can be activated or deactivated in the prototype. Whether an objective is activated or not can be seen in the main page shown in Figure 5.2 – activated objectives are highlighted in blue and non-activated ones are grayed out. If an objective that is still grayed out is



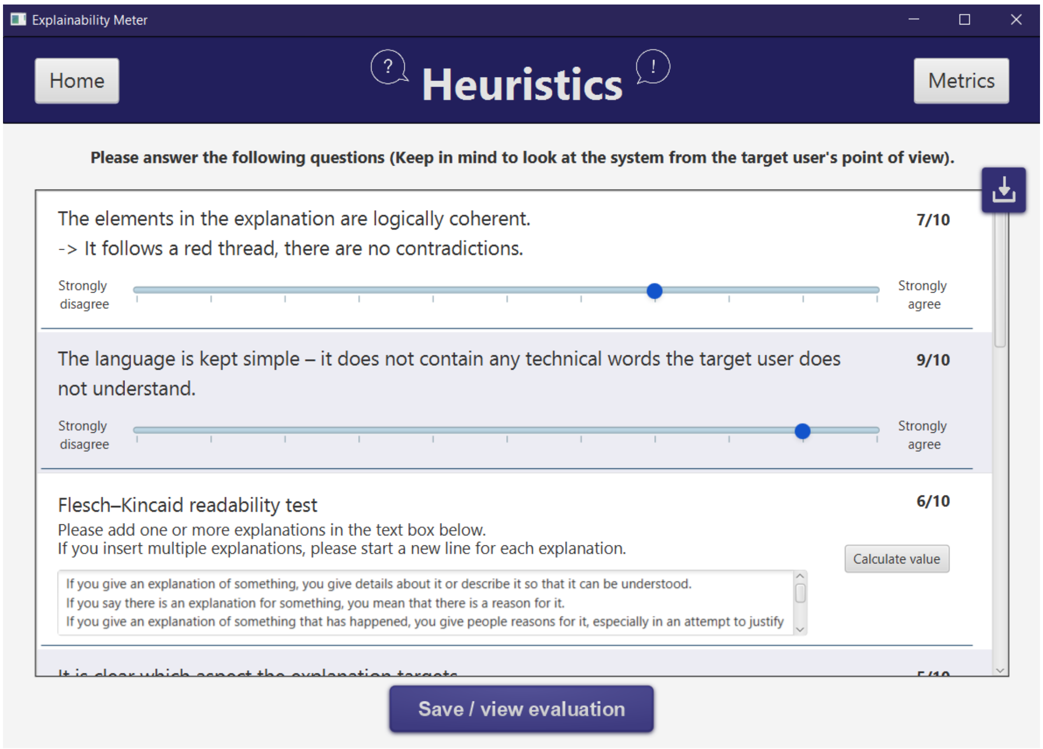
Please select each criterion below that you consider relevant for your system and answer the associated questions. The criteria highlighted in blue were classified as relevant by you.

Figure 5.2: Start page of Prototype

perceived as important, the user can press on this button and is then asked the specific questions from table 5.10. Figure B.1 in the appendix shows this interface for answering the questions using the example of the suitability criterion. The evaluator is first asked whether it is generally relevant for the system to adapt the explanations to specific circumstances or not. If he answers yes, he is asked what exactly needs to be adapted – for example, if there are uncommon contexts in which the system is used or if there are user groups with different levels of prior knowledge.

Heuristics (step 2)

Depending on the answers from step 1, heuristics are selected that the user is asked to answer. Figure 5.3 shows the corresponding screen. The heuristics are mainly estimations that the user is asked to make via Likert scales. This input was implemented using a slider, as this makes the estimation more intuitive than a number that has to be entered. The further the slider is moved to the right, the more one agrees with the statement and the better the explainability is perceived in the corresponding aspect. This is how all but one heuristic was implemented. The exception is the Flesch Reading



Please answer the following questions (Keep in mind to look at the system from the target user's point of view).

The elements in the explanation are logically coherent.
-> It follows a red thread, there are no contradictions. 7/10

Strongly disagree — Strongly agree

The language is kept simple – it does not contain any technical words the target user does not understand. 9/10

Strongly disagree — Strongly agree

Flesch–Kincaid readability test 6/10

Please add one or more explanations in the text box below.
If you insert multiple explanations, please start a new line for each explanation.

If you give an explanation of something, you give details about it or describe it so that it can be understood.
If you say there is an explanation for something, you mean that there is a reason for it.
If you give an explanation of something that has happened, you give people reasons for it, especially in an attempt to justify

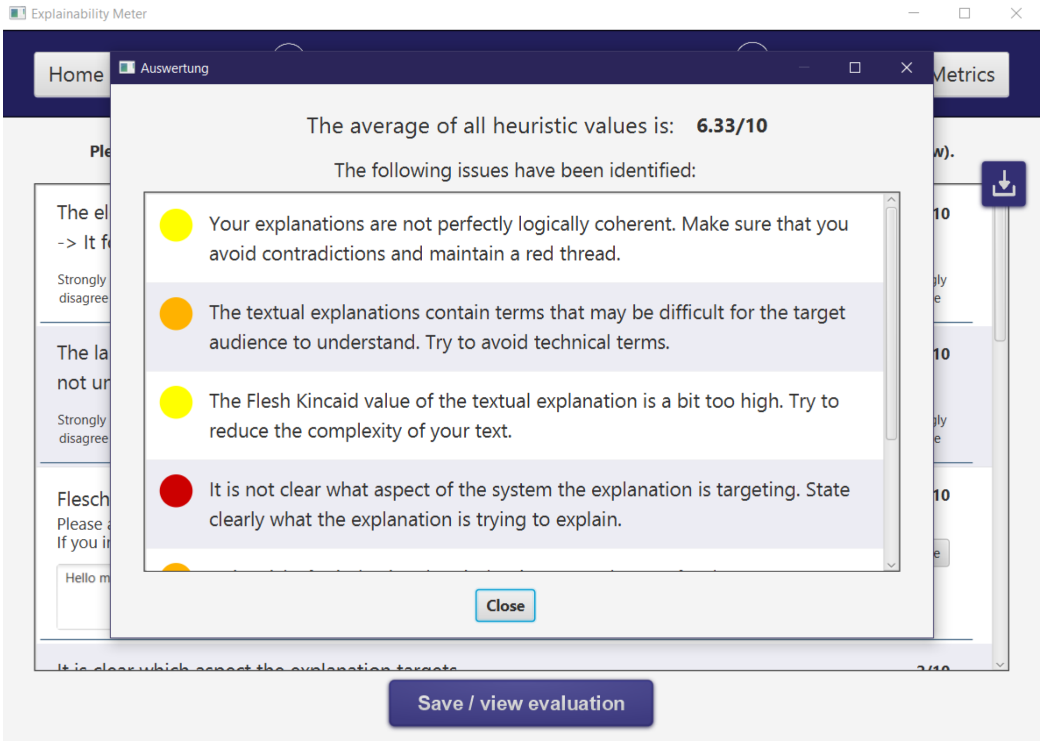
It is clear which aspect the explanation targets 5/10

Figure 5.3: Heuristic question page for the first assessment

Ease Score, which is used to assess the complexity of a textual explanation. For this purpose an existing library was used which calculates the value automatically.[1] The user simply has to copy one or more explanations and paste them into the text field.

Once the user is done with rating the heuristics, they can save their ratings and view the automatic evaluation by clicking on *Save/view evaluation*. If the ratings are saved and the program is closed and later reopened, this scene will ask if the old heuristic values should be reloaded, or if the evaluation should be restarted from scratch.This way, the ratings and evaluation can be displayed again later. After clicking on *Save/view evaluation*, the evaluation pop up will open (see Figure 5.4). In this pop-up window, elements are shown, which, according to the ratings, still need to be improved. For each heuristic that has a value less than or equal to 8, a summary is given of what needs to be changed. The formulation is adapted to how badly the heuristic was evaluated. In addition, depending on the estimation, a marker is placed next to the summary with a color between yellow and red. Red means that a change is urgently needed, and yellow means that a change is

[1] https://github.com/whelk-io/flesch-kincaid



The average of all heuristic values is: **6.33/10**

The following issues have been identified:

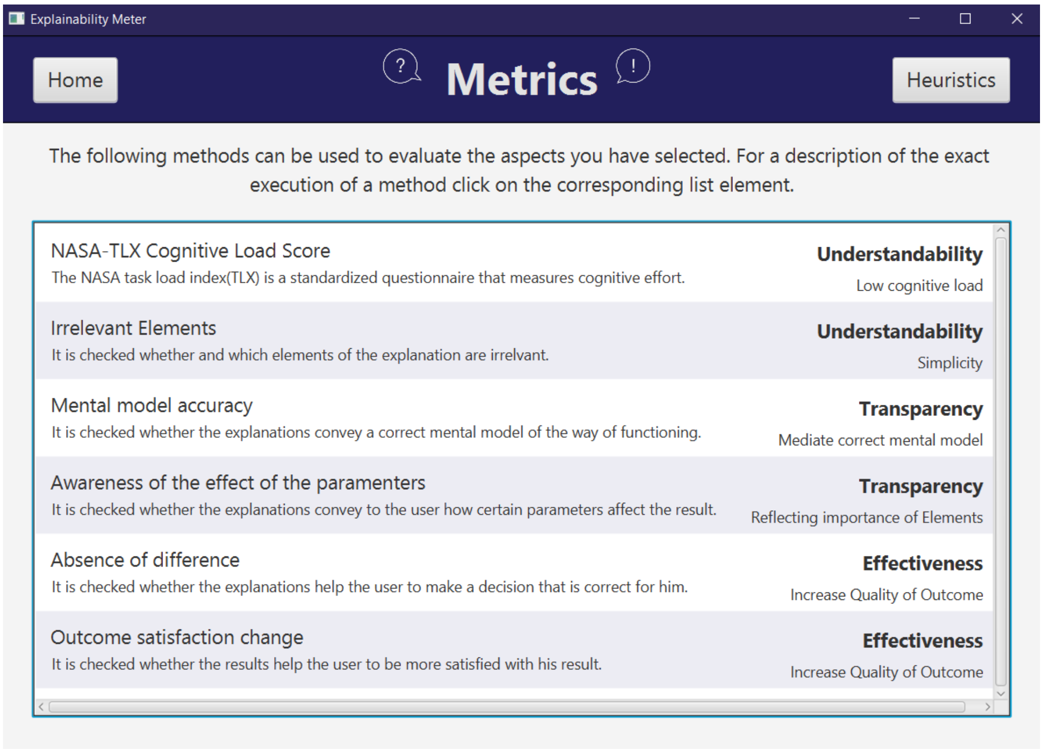
- 🟡 Your explanations are not perfectly logically coherent. Make sure that you avoid contradictions and maintain a red thread.
- 🟠 The textual explanations contain terms that may be difficult for the target audience to understand. Try to avoid technical terms.
- 🟡 The Flesh Kincaid value of the textual explanation is a bit too high. Try to reduce the complexity of your text.
- 🔴 It is not clear what aspect of the system the explanation is targeting. State clearly what the explanation is trying to explain.

Figure 5.4: Evaluation pop up

helpful but not necessary. These statements were also set as tooltips for the markers – they are displayed when hovering over them. To make it easier for the study participants to compare two systems, the average of all heuristic values is also given. If the values of the two systems are very different, the explainability of the system with the higher average value is very likely to be better. However, if the values are close to each other, this number should not be overestimated because it does not contain any weighting. In this case, the study participant can revisit the summaries on the evaluation pop up to make a decision on which system has better explainability.

User studies (step 3)

Since heuristics only provide an estimation and are usually less precise than user studies, the metrics found in the literature are also presented in the *Explainability Meter*. Again, only the metrics that match the previously selected objective are displayed. If the evaluator wants to evaluate an aspect more reliably, they can select a suitable metric on this interface. Each metric is presented with the aspect that is being measured and a brief description of the process. This allows the evaluator to choose which metric



The following methods can be used to evaluate the aspects you have selected. For a description of the exact execution of a method click on the corresponding list element.

NASA-TLX Cognitive Load Score
The NASA task load index(TLX) is a standardized questionnaire that measures cognitive effort.
Understandability
Low cognitive load

Irrelevant Elements
It is checked whether and which elements of the explanation are irrelvant.
Understandability
Simplicity

Mental model accuracy
It is checked whether the explanations convey a correct mental model of the way of functioning.
Transparency
Mediate correct mental model

Awareness of the effect of the paramenters
It is checked whether the explanations convey to the user how certain parameters affect the result.
Transparency
Reflecting importance of Elements

Absence of difference
It is checked whether the explanations help the user to make a decision that is correct for him.
Effectiveness
Increase Quality of Outcome

Outcome satisfaction change
It is checked whether the results help the user to be more satisfied with his result.
Effectiveness
Increase Quality of Outcome

Figure 5.5: Metric page

to use for a more detailed assessment of the desired aspect. Clicking on one of the metrics will open a pop-up showing the exact steps to perform the measurement. Suggestions on how to evaluate the results can also be provided here. An initial implementation of this pop up is presented in the appendix in figure B.2.

Chapter 6

User Study

To test whether the heuristics created in section 5 can be used to assess explainability, a user study was conducted. In this chapter, the research questions addressed by the study are introduced, the procedure is described and, finally, the results of the study are presented.

6.1 Research Questions

Heuristics are methods for evaluating an object with limited resources. In this case, limited resources means that time and money are saved because no user studies have to be carried out. Even if heuristics do not provide perfect results, they should provide consistent results and reveal differences in the measured aspects. Based on this, research questions were developed.

RQ 4 To what extent do the heuristics allow multiple evaluators to agree on a score for a system's explainability?

- RQ 4.1 How much do the absolute values of the ratings evaluators assign differ per heuristic?
- RQ 4.2 How much do the relative values of the ratings evaluators assign differ per heuristic (e.g. does each participant give system A a two points better score than system B)?

RQ 5 Do the heuristics reveal significant differences in terms of explainability in two systems?

The research question RQ4 aims to determine whether the heuristics enable different raters to assign the same rating to a system in terms of explainability. This can either imply that the raters assign the same values per system and heuristic. Or it can imply that raters rate the systems equally in relation to each other, which means that system one, for example, is rated about 3 points better than system two by each rater. Therefore, the research question was refined with two sub-questions. The first of these questions, RQ4.1, attempts to determine whether the heuristics produce similar absolute values. This means that several people assign approximately the same value per heuristic to a system. This would allow evaluating the explainability in absolute terms. The second question RQ4.2 aims to find out whether two systems are comparable with the heuristics – that is, whether the relative values are the same. This is a less strict requirement than the one in RQ4.1, since the values can vary somewhat as long as they vary in the same direction for both systems. Overall, this allows to find out whether the heuristics are suitable for the objective evaluation of explainability. If a heuristic produces values that scatter too much, this could indicate that the heuristic is not explicit enough or is too subjective. The last research question RQ5 intends to test whether the two systems used for the study show a significant difference in explainability. In answering this research question, it will be determined at the same time whether the heuristics are able to reveal differences in explainability of two systems.

6.2 Planning of the User Study

To evaluate the research questions, a user study was conducted in which participants were asked to use the heuristics by assigning a value to each heuristic for a preselected system. Because RQ5 required the evaluation of two systems, there was a choice between the within-subject or between-subject design. Answering research question RQ4.2 requires a within subject design, thus each participant evaluated each of the two systems. In order to mitigate possible bias, half of the participants started with the evaluation of system A and the other half started with system B.

Independent and Dependent Variables

The first independent variable in this user study are the two systems that were evaluated. In addition, the different heuristics described below are also an independent variable. Since a within-subject design is used, each system is evaluated by each participant with each heuristic. The dependent variable is the ratings given by the participants per heuristic per system.

Research Objects

Two similar systems were selected for evaluation, both of which seemed to have relatively good explainability at first sight. If a system with very good explainability and a system with very poor explainability were chosen, the results of the heuristics would most likely be clearer, but the expressiveness of the study would not be as high. With the selected systems, it is tested whether the heuristics also provide results for similarly good explanations. The systems are both online consultants for bicycles. They guide the user through questions to suggest suitable bikes at the end. The first system is an advisor from the brand Decathlon. This system is referenced here and further as system A. The second system, which is here and further referenced as system B, is from the manufacturer ROSE. A screenshot of both systems and the associated links to the systems can be found in Appendix C.1.

Participant Selection

Since the *Explainability Meter* is aimed at IT-related users and therefore includes some technical terms, it was important that the participants had an IT background. This could be fulfilled by studying or working in the field of IT. Overall, the demographics of the participants should be as close as possible to the reality of the IT industry. This means that a 50:50 ratio of women is not necessarily expected, since the proportion of women is not the same as that of men. To acquire participants, known co-students were asked first. Subsequently, other acquaintances working in the field of IT were also asked to participate. The fact that participants are known to me bears the risk of the bias that participants would rate my prototype too good. To mitigate this threat to validity, participants in the study were not asked to evaluate my prototype, but only to use the prototype to evaluate two systems that are independent of me.

Selection of Analyzed Heuristics

Since the participants in the study are not fully trained requirements engineers, the selection of the system's objectives would not be a realistic use case. Moreover, in reality, these objectives would be worked out together with the customer during the requirements engineering process. Therefore, the objectives that the system should fulfill were preselected for the study.

Firstly, the *understandability* was selected, since the system is designed for users who usually do not want to deal with the explanations for a long time in order to be able to understand them. This provided the heuristics R1.1 - R1.3. *Transparency* can also be considered important in this system, as users may be interested in understanding what questions are asked why, and how they affect the outcome. This includes the heuristics R2.1-R2.3. Since bicycles are an expensive purchase, the objective of *effectiveness* was also included, since users are more interested in making a good decision than a quick one (including heuristic R3.1). In addition, it is important for the manufacturer to make a good impression, which is why user *satisfaction* plays a major role. This adds heuristics R5.1 and R5.2. Finally, it is important that the system is appropriate for different user groups. In particular, prior knowledge varies greatly with regard to bicycles. Thus, the heuristics R7.3 and R7.4 are added from *suitability*. Overall, the following heuristics were used to evaluate the two systems:

R1.1,	**R1.2**,	**R1.3**,	**R2.1**,
R2.2,	**R2.3**,	**R3.1**,	**R5.1**,
R5.2,	**R7.3**,	**R7.4**	

6.3 Execution of the User Study

Procedure

At the beginning of the study, the participants received a document about the procedure and the data processing of the study. If they still wanted to participate, they were next given a brief introduction to explainability. Short examples were also shown using the two systems, so that the participants knew what to look out for. In this step, it was also precisely explained to the participants what they were supposed to do, and possible questions were clarified. The participants were then asked to look at the first system. When they felt they had a good overview, they were asked to assign a value to each of the eleven heuristics for the system, but were instructed to skip heuristics they could not understand or answer. At the same time, they could continue to browse through the system. Once they had answered all heuristics they wanted to answer, they were asked to save the results and could then view the summary of ratings shown in Figure 5.4. They then repeated these steps for the second system. At the end of the study, a post-study questionnaire was filled out. This contained the question which website they would recommend in terms of explainability and further demographic questions. The full questionnaire can be found in the appendix in section C.2.

Data collection

Two types of data were collected in the study. First, the assigned value was stored for each system per heuristic. The values are natural numbers in the range from 0 = strongly disagree to 10 = strongly agree (Likert scale). This means that the data are formally on an ordinal scale. In this case, however, a reasonably equal distance between the values can be assumed. Therefore, for the evaluation of the data, a mapping from the ordinal scaled values to values of an interval scale is made, so that calculations such as variance and average can be applied. If the participants did not want to or could not answer a heuristic, the value *-1* was stored so that it could be omitted from the later analysis. This only occurred with one participant for one heuristic (R5.1). The data were saved using the *Explainability Meter* as soon as a participant pressed the *Save/View Evaluation* button. The participant had to select which system they had rated in order for the data to be saved for that system. In total, the values were stored pseudonymized for each participant. The second kind of data was demographic data. This included age, occupation and, optionally, subject of study.

Demographics

Twenty participants were acquired for the study. 85% of the participants were students in the fields of computer science, technical computer science and business informatics. The remaining 15% were employees with IT background (IT specialist, system integration, public service). Due to the low percentage of women in this field, the percentage of women in this study was also rather low (15%), although representative for this field. The average age of the participants was 25.3 (min: 21 years, max: 30 years, SD: 6.32).

Conducting the experiment

For the execution of the study, a laptop was provided so that the participants did not have to install any software. The study was preferably conducted in presence, in a quiet room at the university. However, as the study was conducted during a pandemic, it was not possible for every participant to participate in presence. In this case, the study was conducted via remote desktop using the program AnyDesk[1]. The participants were not observed during the execution of the study, and by pseudonymizing the data they did not have to feel monitored. The duration of the study was 30-60 minutes.

[1] https://anydesk.com/de

Chapter 7

Evaluation of User Study

In this chapter, the research questions defined above are addressed step by step. To this end, the data obtained is first described with the aid of descriptive statistics, focusing in particular on measures of location and spread. Subsequently, the reliability of the values is discussed. For this purpose, Cronbach's alpha is used to test internal reliability and the intraclass coefficient is used to test the interrater agreement. Finally, a hypothesis test, more precisely the Mann-Whitney U test, is used to test whether the two systems show a significant difference in explainability.

The data was evaluated using Python. For the calculation of the ICC and the Cronbach's Alpha, the library Pingouin[1] was used. All remaining statistical evaluations were made with the library SciPy[2]. Additionally, all boxplots were created using Matplotlib[3] from Python.

7.1 Descriptive Statistical Analysis

For the evaluation of RQ 4.1, it is checked how much the values vary per heuristic. The sample variance s^2 is used for this purpose and to visualize the data, boxplots are provided. In addition, in the case of very strong scattering, the reasons for this are investigated and, if possible, improvements for the respective heuristic are suggested.

For RQ 4.2 it must be evaluated whether the relative values are similar enough for a reliable comparison of two systems. To visualize the relative values, the differences between the two ratings for system A and System B must be calculated. In this case, for each participant the value for System B

[1] https://pingouin-stats.org/

[2] https://docs.scipy.org/

[3] https://matplotlib.org/

System	Rater 1	Rater 2	Rater 3
A	8	4	6
B	6	2	4

Table 7.1: Example ratings for clarification

is subtracted from the value for system A. Hence, a negative number means that System B got a better rating from the corresponding participant, and a positive number means that system A got a better rating. For illustration purposes, table 7.1 shows imaginary ratings of three participants for heuristic R1.1. It is clear to see that the absolute values are highly scattered, and therefore RQ4.1 would have to be answered in the negative for heuristic R1.1. However, the relative values are very similar. In this example, each participant rated system A two points better than system B. Thus, by visualizing the differences ($value_A - value_B$), it can be illustrated how suitable each heuristic is for comparing two systems.

Overview of all values

Table 7.2 presents the median and variance for each heuristic for both systems. This table serves as a first overview of the calculated values. These are further explained in the following sections for each heuristic individually and visualized with the help of boxplots to provide a deeper insight into the values.

	System A		**System B**	
Heuristic	**Median**	**Variance**	**Median**	**Variance**
R1.1	9.0	1.26	8.0	2.7
R1.2	7.0	0.49	4.5	4.44
R1.3	9.0	2.08	9.0	3.42
R2.1	8.5	1.61	6.5	5.64
R2.2	8.0	5.82	6.0	7.42
R2.3	9.0	4.39	8.5	5.04
R3.1	8.0	5.35	7.5	4.41
R5.1	8.0	5.82	10.0	0.42
R5.2	8.5	8.44	10.0	0.54
R7.3	9.0	3.64	8.0	2.94
R7.4	0.0	8.06	8.0	12.88

Table 7.2: Variances and Medians of the Values for System A and B

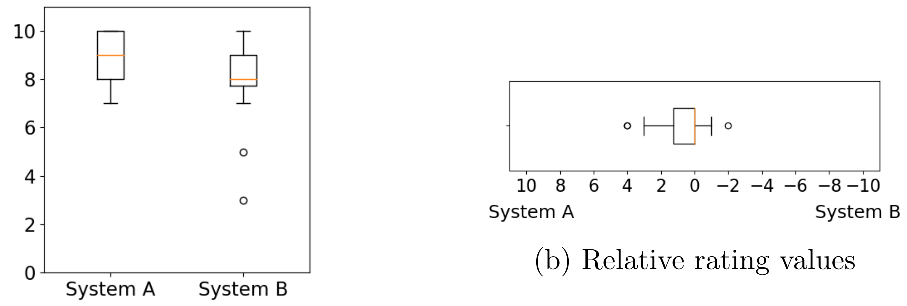
Table 7.3 shows the median and variance of the relative values per heuristic. It must be remembered that the relative values were calculated as a subtraction of the values of A and B. Positive values in the median thus mean that the majority rated system A better, and negative values mean that the majority rated system B better. Based on the medians, it can already be seen that system A was more often rated better in heuristics R1.2, R2.1 and R2.2. System B on the other hand scored better more often in R5.1, R5.2 and R7.4.

	Relative Values	
Heuristic	**Median**	**Variance**
R1.1	0.0	2.56
R1.2	2.5	4.54
R1.3	0.0	5.04
R2.1	2.0	3.98
R2.2	1.5	5.76
R2.3	0.0	5.62
R3.1	0.0	2.86
R5.1	-2.0	5.64
R5.2	-1.0	8.44
R7.3	0.0	6.04
R7.4	-6.0	20.14

Table 7.3: Variances and Medians of the relative Values

What must be noticed here is that 5 of 11 heuristics show a median of zero in the relative values. However, based on the boxplots in the following subsections, in many of these cases a tendency towards one system can be recognized. This shows that the presentation of data using boxplots is very relevant. Therefore, they are used to address each heuristic individually in the following subsections.

R1.1: The language is kept simple – it does not contain any technical words the target user does not understand.



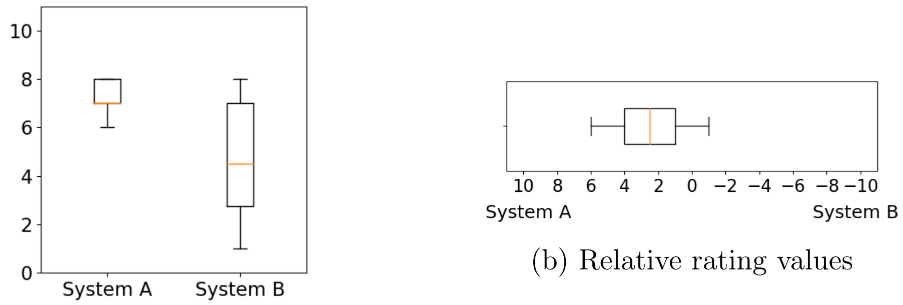
(a) Absolute rating values

(b) Relative rating values

Figure 7.1: Boxplots for R1.1

The variance for heuristic R1.1 is relatively low for both systems. System A has a variance of approximately 1.26 and System B a variance of 2.7. Overall, as you can see in figure 7.1a the ratings for both systems were very good, since no inappropriate technical terms were used. Furthermore, figure 7.1b shows that the relative values do not vary strongly either. It can also be seen from the median of zero that both systems were rated about equally well. Overall, the values indicate that this heuristic is reasonably evident.

R1.2: Flesch Reading Ease Score



(b) Relative rating values

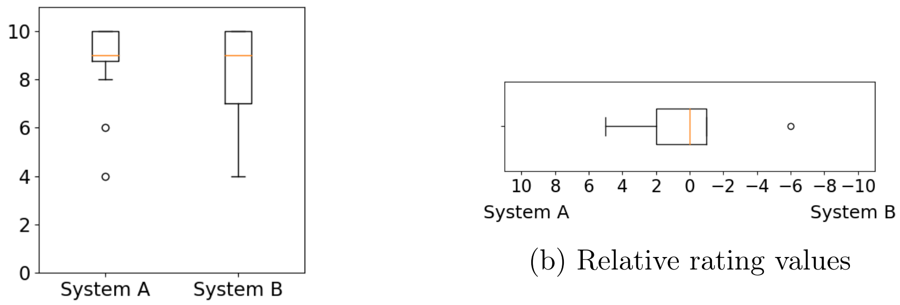
(a) Absolute rating values

Figure 7.2: Boxplots for R1.2

The second heuristic was approximated in the study by having study participants insert 5 explanations that seemed representative. This was done because it would have been too time-consuming for the study to insert every single explanation. The Flesch Reading Ease when considering all textual explanations are 5.6 for system A and 4.4 for System B. In general, this heuristic would always deliver the same values and thus have a variance of zero. What is noticeable, however, is that the Flesch Reading Ease score fluctuates very strongly for very short explanations. This can be seen well in the strongly varying values of system B, although the corresponding texts all show approximately the same complexity. Since the score is rather designed for longer texts, this heuristic does not seem to be stable enough for very short explanations. Hence, the conclusion can be drawn that the heuristic should only be used for longer textual explanations that contain several sentences.

R1.3: The elements in the explanation are logically coherent. It follows a red thread. There are no contradictions

Heuristic R1.3 shows a relatively low variance in the ratings for system A ($s^2{}_A = 2.087$) whereas the variance for system B is higher but still acceptable ($s^2{}_B = 3.427$). The higher variance could be related to the fact that the heuristic consists of two parts. It might be better to split this



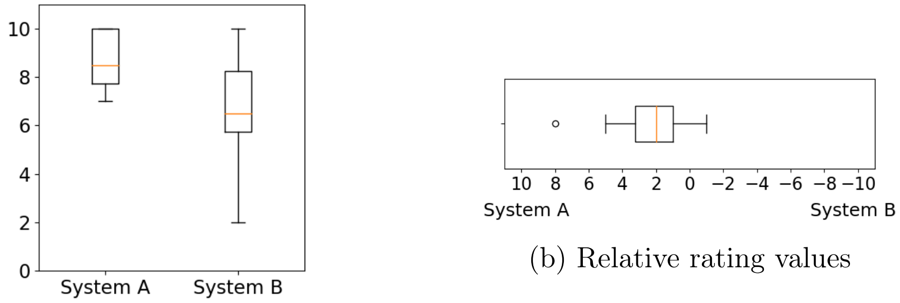
(b) Relative rating values

(a) Absolute rating values

Figure 7.3: Boxplots for R1.3

heuristic into two parts and then average their values to generate the value for this heuristic. The variance in the relative values is high ($s^2{}_{(A-B)} = 5.04$). However, it should be noted that many participants (35%) gave both systems the same score for this heuristic. Logical coherence thus seems to be similar in both systems. The higher variance in the relative scores could therefore be due to the fact that the difference in the systems was so small that a comparison was too difficult for the participants.

R2.1: For each input parameter, it is clear why the system needs this input and what it is used for.



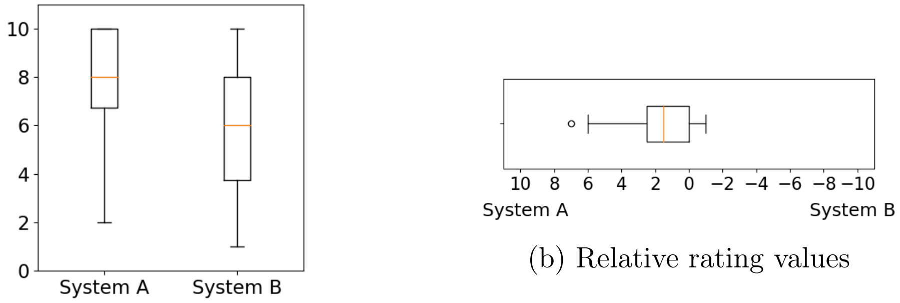
(b) Relative rating values

(a) Absolute rating values

Figure 7.4: Boxplots for R2.1

The variance of the heuristic R2.1 for system A is low with a value of $s^2{}_A = 1.609$. The value for the system B, in contrast, is high with $s^2{}_B = 5.647$. This difference may be related to the fact that this heuristic was worse fulfilled in the system B than in the system A. System B gives little information about why certain input is needed, which is why the opinions here have more room to diverge. Nevertheless, what is notable here is that the relative values clearly identify system A as better. Even though the variance of these values is somewhat high with ${s^2}_{(A-B)} = 3.987$, only one person gave system B a rating one point better and two participants gave it the same rating. By contrast, 85% gave system A a better rating.

R2.2: The role that the parameters the user enters have on the event being explained becomes clear.



(a) Absolute rating values

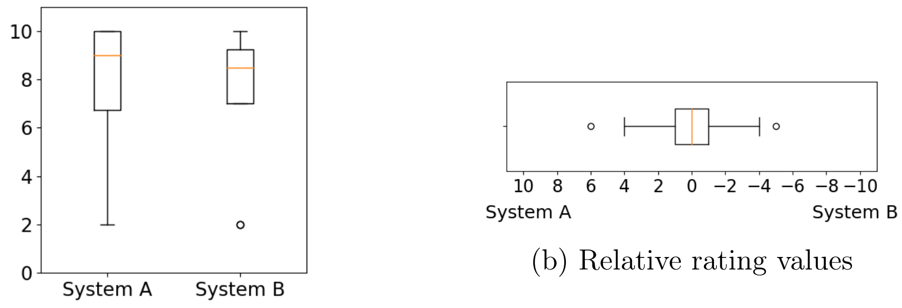
(b) Relative rating values

Figure 7.5: Boxplots for R2.2

The high variance of the absolute values for both systems (${s^2}_A = 5.827$, ${s^2}_B = 7.427$) shows that heuristic R2.2 is not formulated clearly enough or that this aspect is too subjective. It might be unclear what exactly is meant by the event which is to be explained. It would be possible to simplify the formulation of this heuristic by replacing this term with result or outcome. However, this would make the heuristic somewhat less universal, since it is not always necessary to explain the outcome, but in some cases other aspects should rather be explained. Nevertheless, it would be advisable for a more consistent evaluation. The variance of the relative values is also quite high at 5.76. But Figure 7.5b shows that, on average, system A is rated much higher. Only 20% of the participants gave system B a minimally (difference of 1) better rating. Thus, for the simple comparison of two systems, this heuristic performed reasonably well.

R2.3: It is clear which aspect the explanation targets.

Heuristic R2.3 has a rather high variance. The system of system A has a variance of 4.39 and system B a variance of 5.047. The values for both



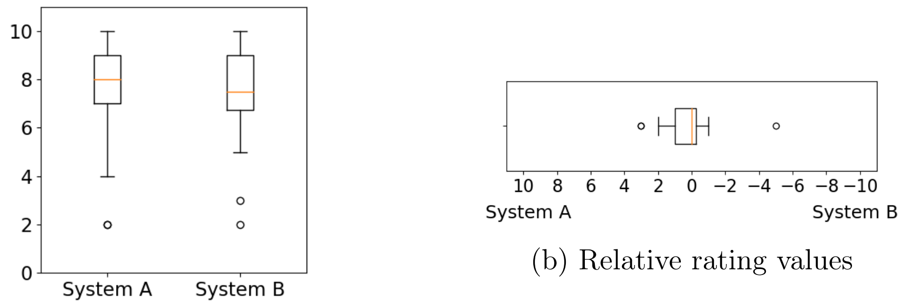
(a) Absolute rating values

(b) Relative rating values

Figure 7.6: Boxplots for R2.3

systems range from 2 to 10, which shows a very large deviation. Since the formulation of this heuristic is already kept simple, there is no possibility of reformulation. What could help this heuristic, would be concrete examples, which make clear, what this heuristic points at. Especially counterexamples could be helpful. In addition, the relative values show that the comparison of the two systems with this heuristic was also unsuccessful($s^2{}_{(A-B)} = 5.627$). 35% of the participants rated both systems the same, whereas there was one participant who gave system A six points more and at the same time one participant who rated system B five points better. This shows that there is a lot of disagreement in this aspect, and that this heuristic does not lead to sufficiently consistent values.

R3.1: The information given can help with the user's decision-making process if the user has not had this information before.



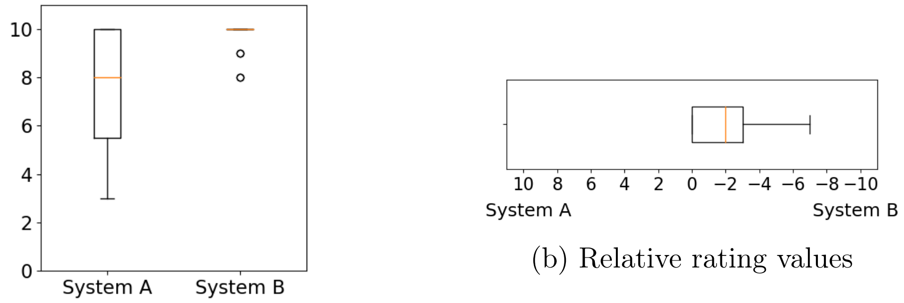
(a) Absolute rating values

(b) Relative rating values

Figure 7.7: Boxplots for R3.1

The absolute values for heuristic R3.1 also show a high variance($s^2{}_A = 5.35$, $s^2{}_B = 4.41$). This may be due to the fact that it is very difficult to assess whether this information helps other people. Nevertheless, this could be manageable for people with a lot of experience, for example in the field of context analysis, creation of personas or similar. However, the figure 7.7a clearly show that some kind of prior experience is needed to put oneself in the perspective of other people or contexts. So, for the absolute evaluation, this heuristic is unsuitable for inexperienced people. For the relative comparison, however, this heuristic shows significantly better values: $s^2{}_{(A-B)} = 2.860$. This heuristic can therefore be used by inexperienced people for a comparison of two systems.

R5.1: The explanation is easy to find.



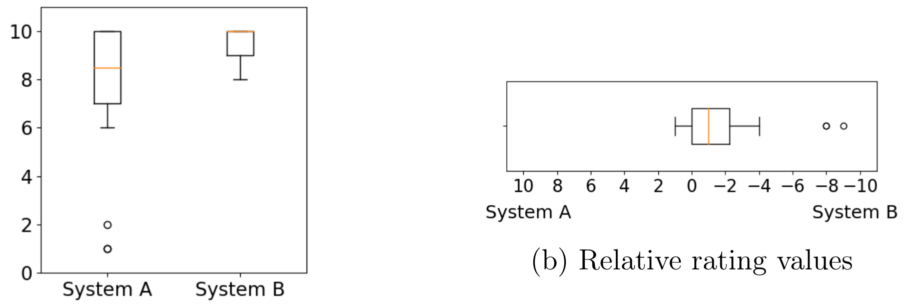
(b) Relative rating values

(a) Absolute rating values

Figure 7.8: Boxplots for R5.1

The next heuristic has an almost perfectly small variance for system B ($s^2{}_B = 0.426$). This is because the explanations in the system were very easy to find. System A's explanations, on the other hand, were not always immediately visible and could be displayed when necessary by clicking on an ⓘ icon next to the item, as you can see in figure C.1 in the appendix. Some participants considered this easy to find, and some participants considered it difficult. The variance is therefore rather high with $s^2{}_A = 5.828$. Additionally, one participant did not answer this heuristic for system A. The participants were told at the beginning of the study that heuristics they could not or did not want to answer could simply be skipped. Thus, it can be assumed that this participant found the question difficult to judge. Overall, skipping the heuristic could indicate that the heuristic was difficult to assess for system A. Due to the high variance of the values of system A, the relative variance is similarly high($s^2{}_{(A-B)} = 5.639$). It can be concluded from this that good detectability of the explanations is recognized, but if the explanations are a bit more difficult to find, the evaluation is again rather subjective. However, when evaluated by usability engineers, this could produce a more consistent result, as they are in a better position to judge such aspects.

R5.2: The explanations are not disruptive and do not interfere with the general use.



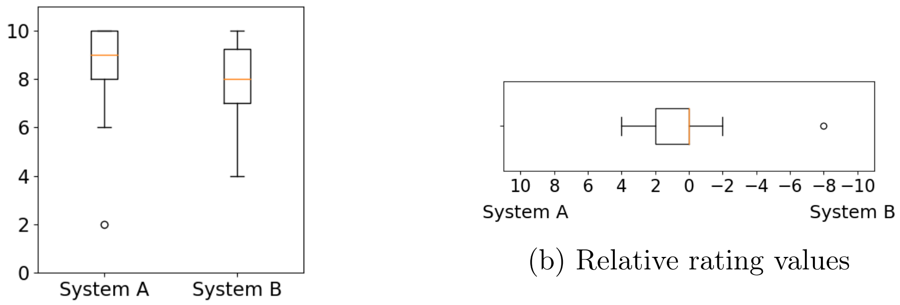
(a) Absolute rating values

(b) Relative rating values

Figure 7.9: Boxplots for R5.2

A very similar pattern emerges for heuristic R5.2. For system B, all participants agreed that the explanations did not interfere with the process. This can be seen clearly in the variance of $s^2{}_B = 0.547$. In the case of system A, on the other hand, there was a wide discrepancy of opinions. The explanations in this system are much larger and therefore take up more space (see Figure C.1 in the appendix), but are only displayed if necessary when the ⓘ icon is pressed. Since the participants were explicitly asked to view all explanations in the study, they had to press on the icon for each item. This may have confounded the results, as participants felt that the explanations were interrupting the flow. It could therefore be that some participants mistook the explanations for disruptive ones, leading to a high level of disagreement, resulting in a variance of $s^2{}_A = 8.440$. As with the previous heuristic, the relative values therefore vary widely($s^2{}_{(A-B)} = 8.447$). This heuristic is therefore also only suitable for experts, since it requires that they manage to evaluate the system independently of their own preferences.

R7.3 For each target group of the system: Is the explanation understandable with the prior knowledge they have?



(b) Relative rating values

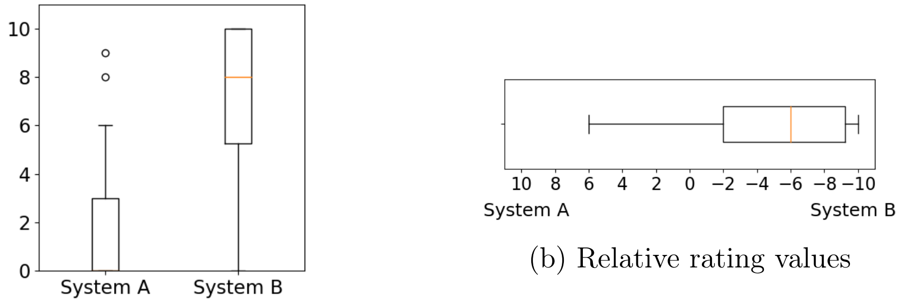
(a) Absolute rating values

Figure 7.10: Boxplots for R7.3

The variance of the absolute values of heuristic 7.3 are acceptable. The variance for system A is ${s^2}_A = 3.639$ and the variance for system B is ${s^2}_B = 2.947$. Overall, the participants also agreed that system A performs better than system B. Only 10% of the participants rated system B better. One strong outlier rated System B 8 points better. This outlier dramatically worsens the variance of the relative values. The total variance is ${s^2}_{(A-B)} = 6.047$, but after filtering out the outlier, the variance is ${s^2}_{(A-B)} = 2.316$, which is fairly good. The variance of the absolute values for system A was also very strongly influenced by this outlier. Without this outlier, the variance for system A is very good with a value of ${s^2}_A = 1.418$. It can be assumed that the heuristic was misunderstood by the one participant in the case of a single such strong outlier, so this heuristic can be considered as consistent.

R7.4: The explanations are adaptable to the user's level of prior knowledge.

The boxplots and variances of this heuristic show that these heuristics were estimated strongly different by the participants(${s^2}_A = 8.060$, ${s^2}_B = 12.887$). The relative values also show an extremely high variance(${s^2}_{(A-B)} = 20.147$). System B offered an explicit selection, where the user could choose whether he wanted detailed advice or advice for beginners. People who assigned zero points in this system were thus either inattentive and missed the selection, or misunderstood the heuristics. In system A, there was no explicit selection for the level of prior knowledge. However, the fact that the user could show explanations optionally could have been interpreted as an adjustment to the



(b) Relative rating values

(a) Absolute rating values

Figure 7.11: Boxplots for R7.4

level of prior knowledge, since expert users simply did not click on any of the explanations. Overall, this shows that this heuristic leaves too much room for interpretation or misunderstanding. This heuristic should therefore be fundamentally revised or discarded.

7.2 Reliability

The next step is to test the reliability. On the one hand, the internal reliability is calculated using Cronbach's alpha to check the extent to which the questions agree with each other. This indicates whether the questions measure the same matter. On the other hand, the intraclass coefficient (ICC) is used for the interrater reliability to check to what extent the raters agree with each other.

Internal reliability

Because the value for the internal reliability is determined among the heuristics, a separate value is determined for the data of each of the two systems. The following interpretation of the alpha value established by George and Mallery [28] is used: $>$ 0.9 = excellent, 0.8 - 0.89 = good, 0.70 - 0.79 = acceptable, 0.60 - 0.69 = questionable, 0.50-0.59 = poor, <0.50 = unacceptable.

Overall	
System	Cronbach's Alpha
A	0.691880
B	0.645047

Table 7.4: Cronbach's Alpha for all Heuristics

The internal reliability is considered as *questionable* for the data of both Systems with a Cronbach's alpha value of $0.6 \leq \alpha < 0.7$ (see Table 7.4). However, since the eleven heuristics test different aspects of explainability (Understandability, Transparency, Effectiveness, Satisfaction, and Suitability), it is not necessarily a problem that the internal reliability for all eleven heuristics is not very high. Therefore, the internal reliability was calculated again for each aspect of explainability. To do this, instead of using all 11 heuristics for the calculation, only the heuristics for the categories were used. This means that for the values for Understandability, for example, the Cronbach's Alpha value was calculated between the three heuristics R1.1, R1.2 and R1.3. Effectiveness was omitted, since there was only one heuristic for this aspect and the internal reliability could therefore not be calculated. The values are presented in table 7.5.

Understandability	
System	Cronbach's Alpha
A	0.416927
B	-0.536264

Transparency	
System	Cronbach's Alpha
A	0.713651
B	0.854113

Satisfaction	
System	Cronbach's Alpha
A	0.885934
B	0.619819

Suitability	
System	Cronbach's Alpha
A	0.259244
B	0.389114

Table 7.5: Cronbach's Alpha for separate aspects

It can be seen that the understandability aspect, consisting of the heuristics R1.1, R1.2 and R1.3, does not show satisfactory internal reliability. The Cronbach's alpha values for both systems are below 0.5, which is considered *unacceptable*. The same applies to the aspect suitability with the heuristics R7.3 and R7.4. The aspect transparency shows better values with the heuristics R2.1, R2.2 and R2.3. For the evaluation of system A, there was an *acceptable* internal reliability, and for the evaluation of system B, there was a *good* internal reliability. In the aspect satisfaction, the heuristics R5.1 and R5.2 also showed *good* internal reliability in the evaluation of system A. However, the internal reliability for System B is categorized as *questionable*.

What can be drawn from these values is that the heuristics for transparency and satisfaction have satisfactory internal reliability. So heuristics R2.1, R2.2 and R2.3 seem to measure the same matter as well as heuristics R5.1 and R5.2. In contrast, the heuristics for understandability and suitability vary a lot internally. This should not be the case, as they try to measure the same aspect. In the aspect of suitability, this can be explained by the fact that the heuristic R7.4 was already identified as unreliable by the descriptive statistics. In the aspect of understandability, it could be related to the fact that the Flesch Reading Ease has a different scale than the heuristic questions that were assessed by the participants. The score 10 will almost never be fulfilled with the Flesch Reading Ease, since extremely simple sentences are required, which is not necessarily appropriate for adult persons. An evaluator would therefore usually assign a score of 10 to sentences with a Flesch Reading Ease of 8. This is therefore a point that should be adjusted.

Interrater reliability

In order to give a reliable answer to the question RQ4, an established statistic method – the intraclass correlation coefficient (ICC) – is used. It is a method to measure the interrater reliability, which determines "the variation between 2 or more raters measuring the same group of subjects" [43]. In this thesis, the raters are the participants and the subjects measured are the heuristics. So for each system, the ICC is calculated to determine if the heuristics can be rated reliably by the different participants. Shrout and Fleiss [75] defined six different formulas for calculating the ICC depending on whether one-way or two-way ANOVA is appropriate, whether the raters are considered random or fixed effects, and whether the analysis will be based on one person or the average of multiple people. Since each participant rated each heuristic, and the participants are a randomly selected sample, the formulas ICC(2,1) and ICC(2,k) are appropriate. Formulas 7.1 and 7.2 are therefore used to calculate the interrater reliability.

$$ICC(2,1) = \frac{BMS - EMS}{BMS + (k-1)EMS + k(JMS - EMS)/n} \tag{7.1}$$

$$ICC(2,k) = \frac{BMS - EMS}{BMS + (JMS - EMS)/n} \tag{7.2}$$

BMS: Between target variance, JMS: Between judges variance, EMS: Residual variance, n: number of targets (in this case heuristics), k: number of judges

The difference between the value of ICC(2,1) and ICC(2,k) lies in how many raters are to be used for evaluation. The formula ICC(2,1) calculates the expected reliability of a single rater. In some cases, however, it is reasonable to use the mean of several raters – Shrout and Fleiss refer to the example of a team of physicians deciding together on an important treatment of a patient [75]. In the case where the mean of a certain number of raters is used, the formula of the ICC(2,k) is to be applied. To interpret the values, a much-referenced paper by Koo and Li [43] is used. According to them, an ICC below 0.5 indicates poor reliability. Values between 0.5 and 0.75 indicate moderate, values between 0.75 and 0.9 good and values grater than 0.9 excellent reliability.

	ICC(2,1)		**ICC(2,k)**	
System	ICC	CI95%	ICC	CI95%
A	**0.4347**	[0.25, 0.71]	**0.9359**	[0.87, 0.98]
B	**0.3115**	[0.16, 0.60]	**0.9005**	[0.79, 0.97]

Table 7.6: Intraclass correlations and their 95% confident intervals

The calculated intraclass correlation coefficients for both systems are shown in table 7.6. In addition, the 95% confidence intervals (CI95%) were given. From this table it can be seen that the values of ICC(2,1) predict poor reliability. Thus, if the evaluation of explainability using the heuristics is done by a single person, the values are not reliable enough. This is consistent with the values obtained from the descriptive statistics, since the variance in some of the heuristics was very high. Consequently, the heuristics as used in the study seem to be too subjective to obtain reliable values from a single rater. The modifications of the heuristics suggested in Chapter 7.1 could, however, improve the value of the ICC(2,1). Moving on to the ICC(2,k) values, it can be seen that the values predict excellent reliability. Thus, if the evaluation of explainability is taken as the average of the ratings of a group of persons, the values are very reliable.

So, the meaning of these values is that the heuristics are well suited when several persons are available for the evaluation of explainability, but when only one person makes the evaluation, the heuristics yet seem to produce too unreliable values.

7.3 Hypothesis Testing

This section examines whether significant differences between the systems could be detected using the heuristics. There are many different methods to test significance. One factor in selecting these methods is whether the data are normally distributed. To check for normal distribution, the Shapiro-Wilk test was used. The exact values from this test can be found in table C.1 in the appendix. Since some of the heuristics have a p-value below 0.05, the null hypothesis of the test is rejected, and it must be assumed that the data are not normally distributed. Accordingly, the Mann-Whitney U test is used for the hypothesis test.

First, the p-Values are calculated individually for all heuristics. The mean values for the ratings of system A and system B are also given in order to remember how the values differ approximately. Secondly, it will also be tested whether the aspects of explainability that were measured show a statistically significant difference between system A and system B. For this purpose, it is assumed that the value for an aspect is calculated using the average values of the associated heuristics (no weighting of heuristics is applied). For the evaluation of the aspect understandability, for example, the average of the values of R1.1, R1.2 and R1.3 is calculated for each participant. At the end, it is also tested whether there is a significant difference between the systems in terms of explainability overall by taking the average of all eleven heuristics. Three different null hypotheses emerge from these three segments:

$H1_0$ Heuristic X did not show any significant difference between systems A and B.

$H2_0$ The average of the heuristics belonging to aspect C did not show any significant difference between systems A and B.

$H3_0$ The average of all eleven heuristics did not show a significant difference between systems A and B.

$$X \in \{\text{R1.1, R1.2, R1.3, R2.1, R2.2, R2.3, R3.1, R5.1, R5.2, R7.3, R7.4}\}$$
$$C \in \{\text{Understandability, Transparency, Effectiveness, Satisfaction, Suitability}\}$$

Understandability

Heuristic	**p-Value**	**Mean of system A**	**Mean of System B**
R1.1	**0.15363**	8.8	8.0
R1.2	**0.00016**	7.1	4.55
R1.3	**0.91539**	8.75	8.35
Overall	**0.00011**	8.21	6.96

Table 7.7: p-Values for the aspect understandability

At a p-value below 0.05 the null hypothesis is rejected and a significant difference is assumed and at a p-value below 0.01 a highly significant difference is assumed. For heuristic R1.1, which deals with the simplicity of language, especially technical terms, it was not possible to identify any statistically significant difference. The same applies to heuristic R1.3, which deals with the logical connection within an explanation. The p-value of heuristic R1.2, on the other hand, is below 0.01, so the null hypothesis $H1_0$ can be rejected for this heuristic and a statistically highly significant difference can be assumed. System B thus seems to have a higher complexity according to the Flesch Readability test. A higher complexity is considered bad according to the criterion C1.1.1. The exact p-values for R1.1, R1.2

and R1.3 can be seen in table 7.7. The aspect understandability as a whole, consisting of the average of the three heuristics also shows a p-Value below 0.01, so the null hypothesis $H2_0$ can be rejected for this aspect. System A thus seems to perform better than System B in the aspect of understandability. The exact value of the p-Value of the averages can be seen in table 7.7 in the row *Overall.*

Transparency			
Heuristic	**p-Value**	**Mean of system A**	**Mean of System B**
R2.1	**0.00458**	8.7	6.45
R2.2	**0.06076**	7.65	5.85
R2.3	**0.73000**	8.1	7.95
Overall	**0.04313**	8.15	6.75

Table 7.8: p-Values for the aspect transparency

For heuristic R2.1 a p-value below 0.01 was calculated, which is why the null hypothesis $H1_0$ could be rejected for this heuristic. A highly significant difference can thus be assumed, indicating that system A better explains why the system needs a certain input and what it is used for. No significant difference was found, conversely, regarding the clarification of the role that the input has on the event being explained (heuristic R2.2). Here, system A also has a higher mean, but the p-value is slightly above 0.05 ($p_{R2.2} = 0.0607$), so that the null hypothesis $H1_0$ could not be rejected, and thus no significant difference can be assumed. For the heuristic R2.3, which tested whether it was possible to identify the aspect targeted by the explanation, no significant difference was found either. Overall, with a p-value below 0.05 a significant difference could be found in the aspect transparency, indicating system A performed better than system B. The exact values for the three heuristics and the overall transparency can be seen in table 7.8.

Effectiveness			
Heuristic	**p-Value**	**Mean of system A**	**Mean of System B**
R3.1/ Overall	**0.58287**	7.5	7.3

Table 7.9: p-Values for the aspect effectiveness

Since only one heuristic was measured for the aspect effectiveness, the values for heuristic R3.1 and the overall effectiveness are identical here.

Heuristic R3.1 measures whether the given information helps the user to decide about the input parameters – in the systems of the user study this could be, for example, which type of gear shift is preferred. For the two systems A and B, no significant difference could be found in this respect. The P-value is $p_{R3.1} = 0.58$.

Satisfaction			
Heuristic	**p-Value**	**Mean of system A**	**Mean of System B**
R5.1	**0.00106**	7.47	9.68
R5.2	**0.00637**	7.6	9.55
Overall	**0.00091**	7.54	9.62

Table 7.10: p-Values for the aspect satisfaction

Heuristic 5.1, which measured whether an explanation is easy to find, showed a highly significant difference. System B performed clearly better in this regard than system A. Whether the explanations are interruptive or interfere with the general use of the program was measured with heuristic R5.2. Again, B performed significantly better than system A, which was confirmed having a p-Value of $p_{R5.1} = 0.0063$, allowing $H1_0$ to be rejected for this heuristic. When averaging the values of R5.1 and R5.2 and comparing the two systems, the values showed that the aspect Satisfaction also showed a highly significant difference. System B therefore performed clearly better than system A in terms of satisfaction. The exact p-values and mean values can be seen in table 7.10.

Suitability			
Heuristic	**p-Value**	**Mean of system A**	**Mean of System B**
R7.3	**0.19278**	8.6	8.05
R7.4	**0.00054**	1.8	6.75
Overall	**0.00636**	5.2	7.4

Table 7.11: p-Values for the aspect suitability

When asked whether the explanation is understandable for each target group of the system (R7.3), no statistically significant difference was found between the two systems. Conversely, a highly significant difference was found in the question whether the explanations are adaptable to the level of prior knowledge (R7.4). The difference between the systems in terms of overall suitability was also calculated to be highly significant. The exact values can be found in Table 7.11.

Overall explainability		
p-Value	**Mean of system A**	**Mean of System B**
0.87219	7.44	7.51

Table 7.12: p-Values for the average Values from each heuristic

For the significance test of explainability as a whole, the average of all eleven heuristics was taken. When looking at the unweighted average of all eleven heuristics, it can be seen that the participants rated the two systems almost equally well. The significance test also shows with a value of $p_{Expl} = 0.87$ that the null hypothesis could not be rejected and thus no significant difference between the two systems can be assumed. The fact that individual aspects of explainability are (highly) significant, but the average of all aspects is not significant, supports the assumption that explainability cannot be measured independently of the objective to be achieved.

To summarize, the following results can be drawn from the Mann-Whitney U test:

$H1_0$ could be rejected for the heuristics R1.2, R2.1, R5.1, R5.2 and R7.4.

$H2_0$ could be rejected for the aspects Understandability, Tansparency, Satisfaction and Suitability.

$H3_0$ could not be rejected.

Overall, it can be concluded from the values that system A has better explainability in terms of understandability and transparency of the explanations. System B, on the other hand, has better explainability in terms of satisfaction and suitability. In a scenario in which a decision must be taken between the two systems with regard to the overall explainability, the importance of different objectives would have to be weighted. For example, if it is particularly important to the customer that the users understand exactly, but nevertheless easily, which parameters have which influence on the results, system A would be more suitable. If, in contrast, a customer values that users find the explanations particularly pleasant, or that the system can adapt particularly well to all target groups, system B would be the right choice.

Chapter 8

Discussion

This chapter contains a brief discussion of the research questions that could be resolved during this thesis. In addition, limitations are identified that relate to the validity of the findings obtained.

8.1 Answering the Research Questions

In chapter 4, a comprehensive concept of criteria for explainability was presented, and numerous metrics were given with which the sub-aspects of these criteria can be measured. All of these presented criteria and metrics were drawn from the existing literature. When defining the research questions RQ1 and RQ2, it was already explained that giving a short answer to these questions will not be possible, as they were used for concept development. However, Section 4.2 provides a comprehensive response to both research questions.

Answer to RQ1: What criteria have already been established in the literature that define good explainability?

Many different criteria have been found in the literature that indicate good explainability. These criteria could be grouped into eleven main criteria, which in turn contain sub-criteria. The identified main criteria are: understandability, transparency, effectiveness, efficiency, satisfaction, correctness, suitability, trustability, persuasiveness, scrutability and debugability. The associated subcriteria can be found in figures 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10 and 4.11. The literature showed that these criteria are closely related to the objective which is intended to be achieved with explainability.

Since metrics are directly linked to the criteria they seek to measure, they have been explained directly in relation to the criteria. Accordingly, the answer to research question RQ2 is similar to the answer to RQ1.

Answer to RQ2: What metrics for measuring explainability are frequently used or recommended in the literature?

Many metrics with explicit execution methods could be found to measure the identified criteria. These methods are briefly described in sections 4.2.1, 4.2.2, 4.2.3, 4.2.4, 4.2.5, 4.2.6, 4.2.7, 4.2.8, 4.2.9, 4.2.10 and 4.2.11. In addition, many questionnaires were retrieved, which were often used for subjective evaluation of explainability. From these questionnaires, questions were compiled and generalized, which can be found in appendix A.

The answer to the third research question became clear during the development of the concept and was illustrated with two examples in section 4.4. In addition, the results of the study also revealed evidence to support the answer that was formed during the conceptual phase.

Answer to RQ3: Is it possible to measure the explainability of a software system, regardless of the type of system?

Within this work, it became clear that the characteristics of explainability strongly depend on the objectives to be achieved by the explanations. With the current state of knowledge, it is not possible to develop a universal evaluation method that can achieve a satisfactory result regardless of this objective. Supporting this, the results of the study show that the two systems differ greatly in certain aspects of explainability (statically significant differences), but when all aspects are mixed together, no difference can be seen. Depending on the exact objective to be achieved with explainability, system A or system B performed better. The question RQ3 must therefore be answered negatively at this point of research.

In the second part of the thesis, heuristics were developed, which were then reviewed with the help of a user study. In the context of this user study, two further research questions were defined, one relating to the heuristics and the other to the research objects (referred to as system A and B).

Answer to RQ4: To what extent do the heuristics allow multiple evaluators to agree on a score for a system's explainability?

It can be seen from the values from the ICC(2,1) that the heuristics are not yet performing well enough for single raters to agree on their ratings. However, the value of ICC(2,k) shows that when taking the average of the ratings of a given set of evaluators, these values show a very good interrater agreement.

RQ 4.1: How much do the absolute values of the ratings evaluators assign differ per heuristic?

The values of the variances are presented in table 7.2. In addition, further measures of location and spread can be taken from the boxplots in figures 7.1a - 7.11a. Overall, the ratings of the heuristics were relatively scattered, which could indicate that the heuristics were still rather too subjective. Individual suggestions to refine the heuristics were given in section 7.1.

RQ4.2: How much do the relative values of the ratings evaluators assign differ per heuristic?

The relative values were calculated by subtracting the value of system B from the value of system A for each participant. The variances for these values can be found in table 7.3. Further measures of location and spread can be taken from the boxplots in figures 7.1b - 7.11b. The variance of the relative values behaved similarly to the variance of the absolute values. This means that also for the comparison of two systems, the heuristics still seemed too unstable.

To answer research question RQ5, the Mann-Whitney U test was applied. For this purpose, three null hypotheses were formulated, which were partially rejected (see section 7.3). The exact values for the Mann-Whitney U test can be found in tables 7.7 - 7.12.

Answer to RQ 5 Do the heuristics reveal significant differences in terms of explainability in two systems?

For explainability as an unweighted average of all eleven heuristics, no significant difference was found between the two selected systems. However, partitioning the heuristics into their aspects, they revealed significant differences. System A therefore has better explainability in terms of understandability and transparency, and system B has better explainability in terms of satisfaction and suitability.

8.2 Limitaions

This work has been done to the best ability with the resources available. Nevertheless, there is a possibility that the results may not reflect the full reality in some aspects.

Literature Review

Due to the circumstances described in chapter 3, the decision was made to conduct a secondary literature review. This carries the risk of missing out on certain literature. Since the starting set consisted of pre-filtered papers, it is possible that this already introduced a bias. In addition, the subsequent evaluation of the literature was made by a single person, which may allow subjectivity. For mitigation, explicit inclusion and exclusion criteria were defined to make the selection as objective as possible. The papers from the start set from the paper by Chazette et al. [13] were also selected with a defined procedure such that the bias was also kept as small as possible here. Overall, however, it should be noted that it is likely that not all existing papers relevant to this topic were included.

User Study

The study is limited to people who currently live in Germany, or more precisely in Hanover. A cultural bias is therefore possible. In addition, the age range of 21-30 years poses a risk of bias. However, since this study was not intended to reflect the overall picture of the population, but rather to test whether the heuristics allow the generation of consistent values, these biases are acceptable. Another threat to validity could be that the participants were given a short introduction to the topic and examples of the two systems were shown to clarify what is meant by explainability or more precisely by an explanation. Such an introduction and especially such examples related to the system would not be given in a realistic use of the prototype or the heuristics. This was however a necessary step, since the previous knowledge of explainability could not be presupposed. The final threat to validity is the fact that mainly people who I know personally participated in the study. As already described in section 6.2 *Participant Selection*, this threat was kept as small as possible by using the *Explainability Meter* to evaluate other systems rather than evaluating the *Explainability Meter* itself.

Chapter 9

Conclusion and Future Work

9.1 Conclusion

The results of this work can be divided into two parts. The first part covers the concept as a whole for evaluating explainability. The second part deals with the development and testing of heuristics for the evaluation of explainability.

9.1.1 Overall Concept

In the first part of this thesis, an extensive literature review was conducted to conceptualize an approach to make explainability in a software system measurable. It became clear that the measurement of explainability is strongly related to the objective the explanations are intended to achieve. Depending on the objective, there are various criteria to be met. Possible criteria were divided into eleven main criteria, which in turn contained sub-criteria. This allowed a good overview of the requirements that can be placed on explanations. As a next step, metrics were presented for each of the sub-criteria to measure them. An important finding is that the literature already provides many metrics to measure individual aspects of explainability, mostly with user studies. What is missing, however, is an instrument to measure the explainability of a system independent of user studies. For this purpose, heuristics were created in the second part of this thesis.

9.1.2 Heuristics as an Evaluation Method

Using heuristics is a good way to get an initial assessment of explainability. They can help to uncover problems with the explanations in order to subsequently initiate more precise evaluations on this issue – for example, user studies. This saves a lot of resources, as there is no need to conduct a user study for each (sub)criterion, since the heuristics already highlight the most important problems.

In this case, the heuristics consisted mainly of assessment questions that an evaluator was asked to answer about the system. There are two risks to this type of heuristic. The first risk is subjectivity – when an evaluator answers these questions, his or her subjective opinion will often factor into the answer. However, since objective evaluation is required, it is desirable that the heuristics produce as consistent results as possible between evaluators. To test this, the intraclass correlations coefficient was used. It showed that when several evaluators rate the heuristics and the mean of these ratings is taken at the end, the interrater agreement is very high. The evaluation can therefore be considered objective as long as several evaluators are available.

The second risk of heuristics is that the real differences between systems might be lost, since they are only estimates. For this purpose, the Mann-Whitney U test was used to check whether the heuristics reveal differences between System A and System B. These differences were found, as system A performed significantly better in the aspects of understandability and transparency. And system B performed significantly better in the aspects of satisfaction and suitability. This showed that the heuristics were able to detect differences between two systems.

Overall, the heuristics produced satisfactory results. Nevertheless, some suggestions were provided for improvements that could make the evaluation more objective if only one evaluator is available.

9.2 Future Work

The field of explainability still holds a lot of research potential, especially in areas not explicitly related to artificial intelligence. This thesis revealed many points that can be further elaborated or developed.

Mapping system types to specific objectives

In the context of this work, it became clear that the evaluation of explainability is not possible without considering the objectives that are pursued with it. Accordingly, the prototype that has been developed presupposes a step in which the user must first define this objective. However, it is conceivable that the objectives are similar for similar system types. By system types, it is not meant recommendation systems vs. communication systems or similar. But rather critical vs. non-critical systems or business vs. leisure systems. Thus, it can be investigated what properties exist that induce different objectives to be achieved with explainability. These properties can be grouped into system types. Since these system types are then already mapped to the corresponding objectives, a standardized procedure can then be introduced for evaluating the explainability within these types. Furthermore, the mapping of the objectives that explainability is supposed to achieve in certain systems is not only important for evaluation, but can be used analogously to the development of design guidelines using the corresponding criteria.

Validation of concept with experts

While the concept presented in section 3 is based on information from the literature, the compilation may still contain subjective biases. It would therefore be valuable to have the concept, and in particular the grouping of the eleven main criteria, validated by a group of experts. If these experts came to a consistent result, this could be used as a basis for further research, also beyond the scope of the evaluation.

Defining the metrics according to standards

Since it is important for the validity of results that metrics are well defined, there are already standards for the definition of metrics. IEEE, for example, provides templates that can be used to define metrics. [36] Due to limited space, this has not been done in this thesis, but could be useful to ensure that the presented metrics can be applied reliably. In particular, the own proposals for metrics in section 4.3 should be formulated using such a standard and then validated.

Analysis of the required number of evaluators

Through the calculations of the intraclass correlation coefficient (ICC), it became clear that the evaluation of explainability using the heuristics only provide reliable values if a number of multiple evaluators rate and the average is formed. An interesting further research would be to check how many evaluators exactly are needed to achieve a satisfactory interrater reliability.

Appendix A

Gathered Questionnaires

Since some of the questions were composed of several questions, or were formulated too specifically for certain systems, it is necessary to make a generalization at some points. These generalizations are listed here and are indicated in the questions by square brackets.

- *event*: An event is an action performed by the system. This can be, for example, a response of the system to an input from the user.
- *result*: A result is a response of the system wanted by the user. It is therefore a specialization of an event.
- *recommendation*: Specialization of a result. It is used for recommender systems - a recommendation of the system.
- *process*: A process is a level above an event. It contains sequences of steps which can also be events, for example.
- *action*: An interaction with the system performed by the user.
- *input*: An input is information for the system that comes from the user. This could be data input, for example, but simple clicks can also be input if they contain information.
- *system*: The overall system of which the explainability is measured.
- *explanation*: The instance that explains a matter about the system to the user. This can be, for example, a simple text, but also an illustration or an auditory explanation.

ID	Question	Based on
1	It was easy to understand why/how the system did [*event*].	[53, 80]
2	The explanations provided made sense to me.	[9, 19]
3	If explanations did not make sense to you, explain why?	[19]
4	The explanation contains terms that are confusing to me.	[53]
5	Between system A and B, whose explanations do you think can better help you understand [*event*, *result*]?	[83]
6	The task was very mentally demanding.	[19]
7	The explanation is easy to read.	[4]
8	The length of the Explanation is appropriate / is too long to be useful.	[4, 53]
9	The explanation is written in correct, appropriate English.	[4]
10	The arrangement/organization of information is very logical.	[53, 60]

Table A.1: Questions to assess the criterion understandability

ID	Question	Based on
1	I understood why *an event* happened.	[2, 6, 15, 19, 21, 38, 41, 59]
2	This explanation makes [*process*] clear to me.	[44, 45]
3	The response helps me understand what the [*result*] is based on.	[68, 95]
4	I find that the system gives enough explanation why [*event*] happens.	[65]
5	The information given by the explanation was too much/to little.	[80]
6	The explanation fails to reveal the reasoning behind [*event*].	[5, 95]
7	It understandable why the system needs [*input*].	[24]
8	Is more statistical data required? Why and what exactly?	[24]

Table A.2: Questions to assess the criterion transparency

ID	Question	Based on
1	The explanation provided has sufficient information to make an informed decision.	[2, 38]
2	The explanation helps me determine how I feel about [*system*, *result*, *recommendation*].	[5, 68]
3	Between system A and B, whose explanations can better help you make a more informed decision?	[4, 83]
4	I am very certain about what I need in respect of each attribute.	[15]
5	The explanation was very helpful.	[60]
6	htbow would you rate your knowledge about [*system*, *recommendation*]?	[15]

Table A.3: Questions to assess the criterion effectiveness

ID	Question	Based on
1	The explanations help to make [*input*] faster than without.	[4, 5, 14, 38]
2	I needed a lot of time to interpret the explanations.	[80]
3	The explanations do not contain superfluous information.	[4]

Table A.4: Questions to assess the criterion efficiency

ID	Question	Based on
1	Overall, I am satisfied with the system.	[15, 19, 25, 38]
2	Overall, the system was easy to use.	[9]
3	Overall, the [*system*, *explanation*] was useful.	[25, 53, 92, 95]
4	Overall, the system was enjoyable.	[92]
5	I would enjoy using the system when explanations like that are given.	[9, 44, 45]
6	Generally, between system A and B, whose recommendations are you more satisfied with?	[83]
7	The provided explanation: really captures my tastes.	[6]
8	The explanations that were provided were good.	[41]
9	The explanations were intuitive to use.	[80]
10	The explanation makes [*action*] easy.	[5, 60]
11	The explanation is aesthetically pleasing.	[60]
12	Content layout and order of elements in explanations are satisfying.	[4]
13	The explanation convinces you that the system is fair while doing [*action*].	[86]

Table A.5: Questions to assess the criterion satisfaction

ID	Question	Based on
1	The explanation corresponds to my own decision making process	[2]
2	In what use case would you use the explanations?	[80]

Table A.6: Questions to assess the criterion suitability

ID	Question	Based on
1	I have high trust in the [*system*, *explanation*, *result*].	[15, 19, 25, 60]
2	The explanation increased my trust.	[5, 14, 21, 59, 60]
3	Data and explanations are enough to trust the system.	[4]
4	The explanation seem consistent.	[4]
5	The system is like an expert.	[65, 91]
6	The system estimates my prior knowledge well.	[65, 91]
7	The system is honest/genuine/truthful.	[65, 91, 92]
8	The system is trying its best to support me.	[65, 91, 92]
9	The system puts my interest first.	[91, 92]
10	The system provides unbiased [*result*].	[91]
11	The system is competent.	[25, 92]
12	The system knows enough to support me well.	[65, 91, 92]
13	The system would keep its commitments.	[92]
14	I felt the system displayed a warm and caring attitude towards me.	[92]

Table A.7: Questions to assess the criterion trustability

ID	Question	Based on
1	The explanation is convincing.	[44, 45, 95]
2	The explanation made the recommendation more convincing.	[21, 59]
3	The [*recommendation*, *result*] is convincing.	[5]
4	I like the [*recommendation*, *result*].	[44, 45]
5	The system returned to me some good [*recommendations*, *results*].	[15, 44, 45]
6	The explanation made me more confident about [*input*].	[2, 38, 45]
7	I would purchase the product I just chose if given the opportunity.	[15]
8	The response makes me want to buy one of the recommended products.	[25, 68]
9	I will use the system again if I need some tool like that.	[19, 53, 65]
10	I would suggest the system to my friends.	[19, 53]

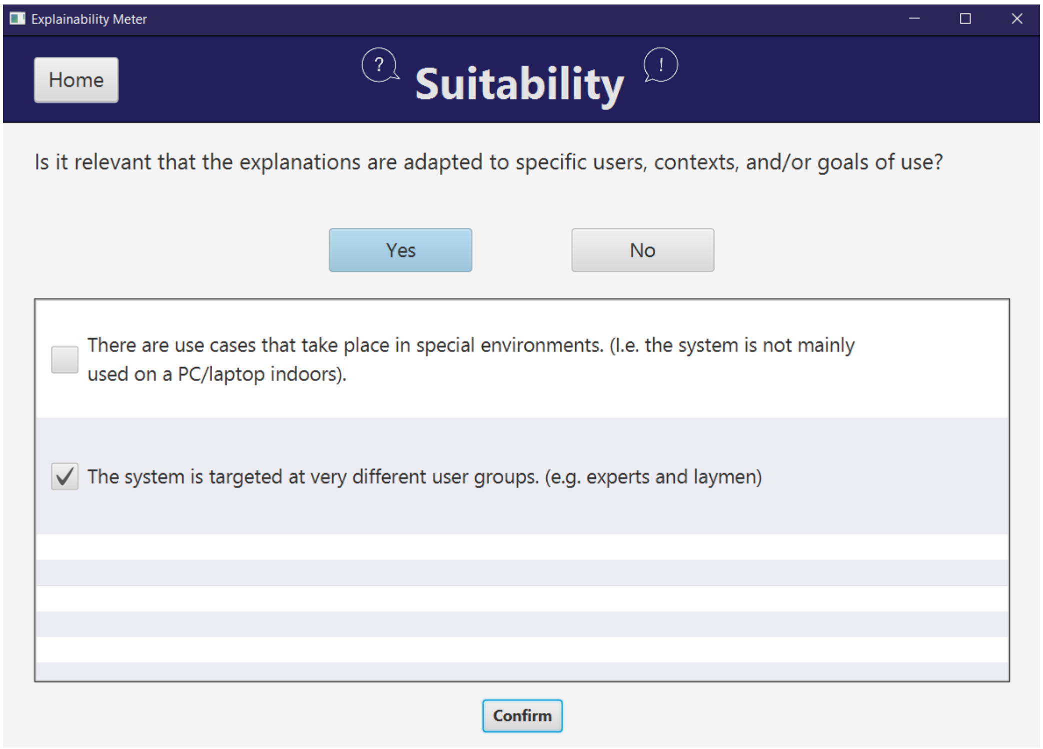
Table A.8: Questions to assess the criterion persuasiveness

ID	Question	Based on
1	The system would make it difficult for me to correct the reasoning behind the recommendation	[5]
2	The response allows me to understand if the system made an error in interpreting my request.	[68]
3	I felt in control of telling the system what I want	[38, 53]

Table A.9: Questions to assess the criterion scrutability

Appendix B

Prototype

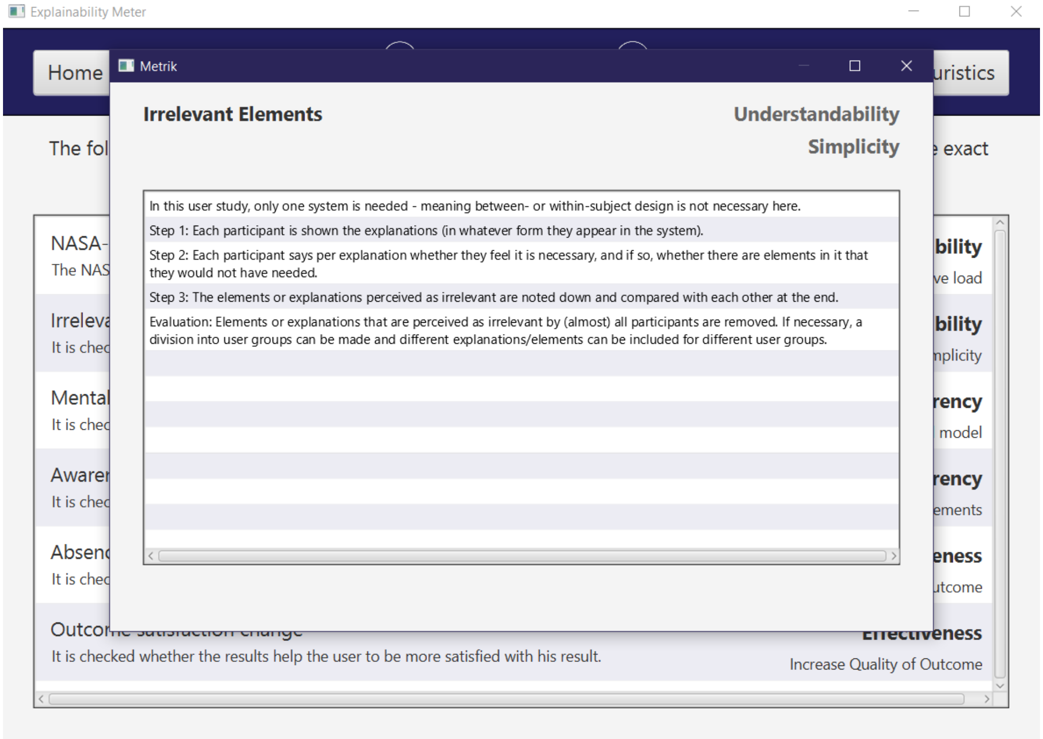


Is it relevant that the explanations are adapted to specific users, contexts, and/or goals of use?

□ There are use cases that take place in special environments. (I.e. the system is not mainly used on a PC/laptop indoors).

☑ The system is targeted at very different user groups. (e.g. experts and laymen)

Figure B.1: Questionpage for the criterion suitability



Irrelevant Elements

Understandability

Simplicity

In this user study, only one system is needed - meaning between- or within-subject design is not necessary here.

Step 1: Each participant is shown the explanations (in whatever form they appear in the system).

Step 2: Each participant says per explanation whether they feel it is necessary, and if so, whether there are elements in it that they would not have needed.

Step 3: The elements or explanations perceived as irrelevant are noted down and compared with each other at the end.

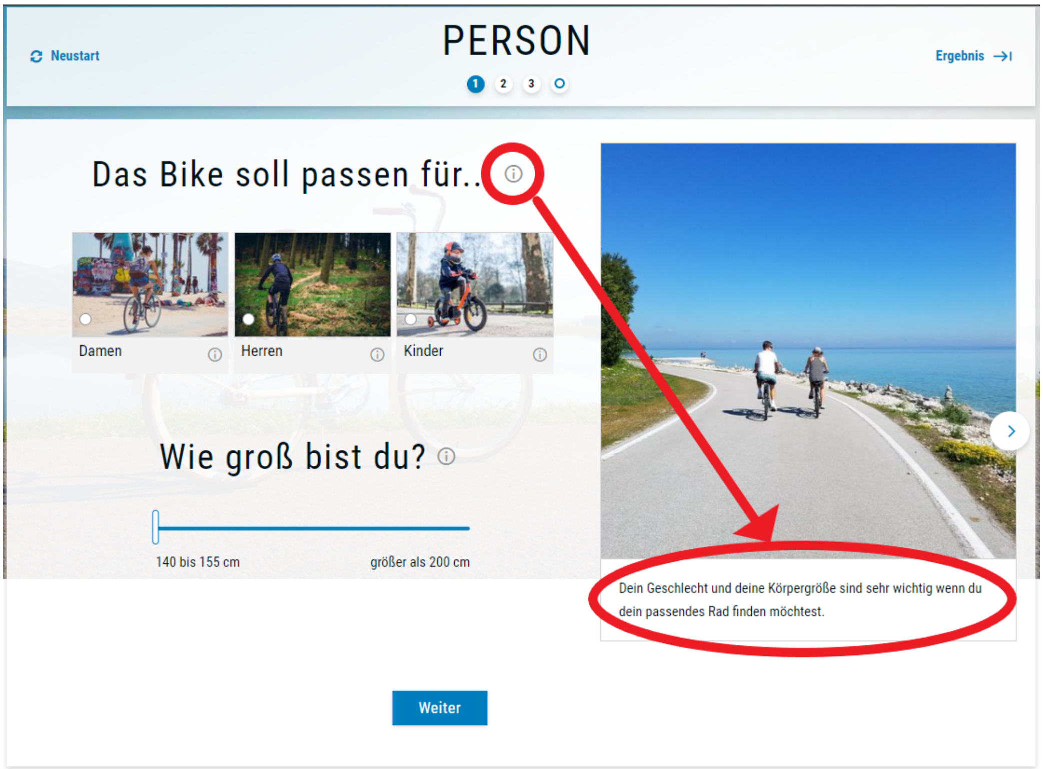
Evaluation: Elements or explanations that are perceived as irrelevant by (almost) all participants are removed. If necessary, a division into user groups can be made and different explanations/elements can be included for different user groups.

Figure B.2: Pop up for Steps of the corresponding metric

Appendix C

Study

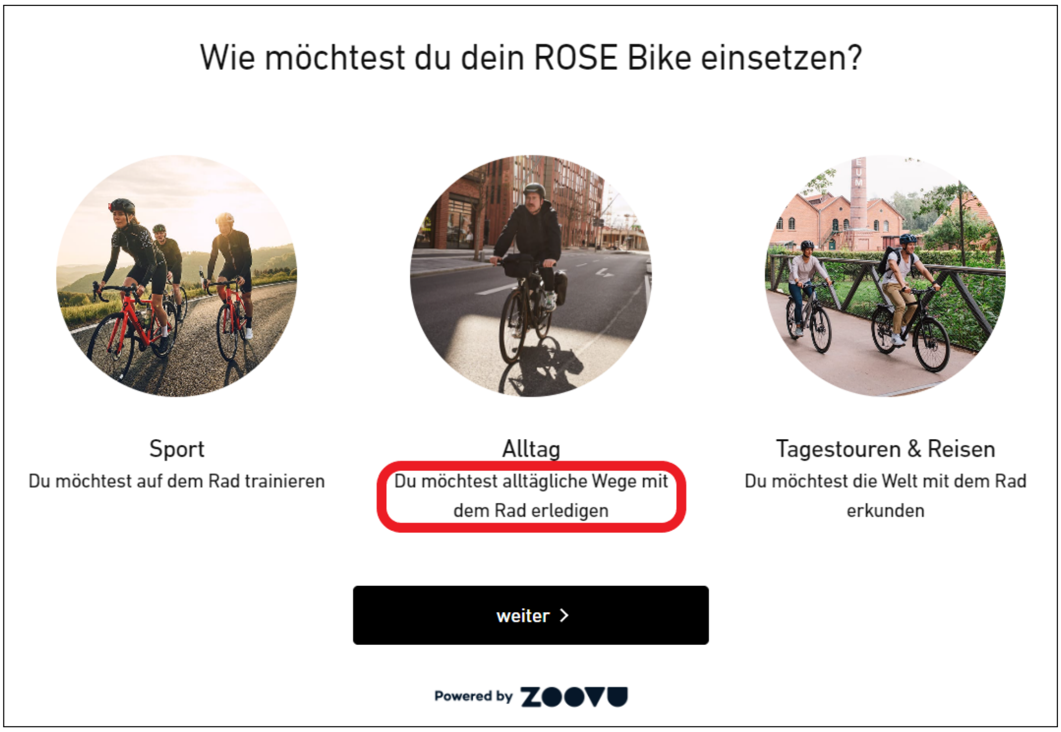
C.1 Systems for Study



Dein Geschlecht und deine Körpergröße sind sehr wichtig wenn du dein passendes Rad finden möchtest.

Figure C.1: System A

Decathlon. R-a-bikeberater.
https://www.decathlon.de/landing/bikeberater/_/R-a-Bikeberater
Accessed: 2022-08-24



Wie möchtest du dein ROSE Bike einsetzen?

Sport
Du möchtest auf dem Rad trainieren

Alltag
Du möchtest alltägliche Wege mit dem Rad erledigen

Tagestouren & Reisen
Du möchtest die Welt mit dem Rad erkunden

Figure C.2: System B

ROSE. Individuelle bike-beratung.
`https://www.rosebikes.de/bike-finder`
Accessed: 2022-08-24.

C.2 Post-study Questionnaire

Post-study Questionnaire

□ Ich habe das Dokument zur Übersicht und Einverständnis der Studie gelesen und stimme zu.

Bitte geben Sie eine selbstgewählte 4-stellige Zahl an. ____________________

Diese Zahl wird dafür genutzt, um Ihre Ergebnisse pseudonymisiert auswerten zu können.

Welche Webseite würden Sie hinsichtlich Erklärbarkeit empfehlen?
□ Webseite von Decathlon
□ Webseite von ROSE

Bitte geben Sie ihr Geschlecht an
□ Weiblich
□ Männlich
□ Divers

Bitte geben Sie ihr Alter in Jahren an. ______________________________

Bitte geben Sie ihren Beruf an. ______________________________________

Falls Sie Student sind: Bitte geben Sie ihr Studienfach an.
□ Informatik
□ Technische Informatik
□ Wirtschafts-Informatik
□ ________________________

C.3 Test of Normality

Heuristic	System A	System B
H1.1	0.0055	0.0068
H1.2	0.0017	0.1796
H1.3	0.0001	0.0011
H2.1	0.0006	0.1398
H2.2	0.0088	0.2469
H2.3	0.0010	0.0001
H3.1	0.0018	0.0332
H5.1	0.0161	0.0000
H5.2	0.0005	0.00001
H7.3	0.0001	0.0627
H7.4	0.00005	0.0019

Table C.1: P-Values of Shapiro-Wilk test for Normality

List of Tables

List of Figures

Bibliography

[1] B. Abdollahi and O. Nasraoui. Transparency in fair machine learning: the case of explainable recommender systems. In *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, pages 21–35. Springer International Publishing, Cham, 2018.

[2] A. Adhikari, D. M. J. Tax, R. Satta, and M. Faeth. Leafage: Example-based and feature importance-based explanations for black-box ml models. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–7. IEEE, 2019.

[3] V. Arya, R. K. E. Bellamy, P.-Y. Chen, A. Dhurandhar, M. Hind, S. C. Hoffman, S. Houde, Q. V. Liao, R. Luss, A. Mojsilović, S. Mourad, P. Pedemonte, R. Raghavendra, J. Richards, P. Sattigeri, K. Shanmugam, M. Singh, K. R. Varshney, D. Wei, and Y. Zhang. One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. *arXiv preprint arXiv:1909.03012*, 2019.

[4] I. Baaj and J.-P. Poli. Natural language generation of explanations of fuzzy inference decisions. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6. IEEE, 2019.

[5] K. Balog and F. Radlinski. Measuring recommendation explanation quality: The conflicting goals of explanations. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 329–338, New York, NY, USA, 2020. Association for Computing Machinery.

[6] V. Bellini, A. Schiavone, T. Di Noia, A. Ragone, and E. Di Sciascio. Knowledge-aware autoencoders for explainable recommender systems. In *Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems*, DLRS 2018, page 24–31, New York, NY, USA, 2018. Association for Computing Machinery.

[7] S. Bobek, P. Bałaga, and G. J. Nalepa. Towards model-agnostic ensemble explanations. In *Computational Science – ICCS*, pages 39–51. Springer International Publishing, 2021.

[8] W. Brunotte, L. Chazette, and K. Korte. Can explanations support privacy awareness? a research roadmap. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 176–180, 2021.

[9] G. Carenini, V. O. Mittal, and J. D. Moore. Generating patient-specific interactive natural language explanations. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, page 5. American Medical Informatics Association, 1994.

[10] M. Caro-Martínez, G. Jiménez-Díaz, and J. A. Recio-García. Conceptual modeling of explainable recommender systems: An ontological formalization to guide their design and development. *Journal of Artificial Intelligence Research*, 71:557–589, 2021.

[11] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8), 2019.

[12] L. Chazette, W. Brunotte, and T. Speith. Exploring explainability: A definition, a model, and a knowledge catalogue. In *2021 IEEE 29th International Requirements Engineering Conference (RE)*, pages 197–208. IEEE, 2021.

[13] L. Chazette, W. Brunotte, and T. Speith. Supplementary material for research paper "Exploring explainability: A definition, a model, and a knowledge catalogue". In *2021 IEEE 29th International Requirements Engineering Conference (RE)*. IEEE, 2021.

[14] L. Chen and P. Pu. Trust building in recommender agents. In *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*, pages 135–145. INSTICC, 2005.

[15] L. Chen, D. Yan, and F. Wang. User evaluations on sentiment-based recommendation explanations. *ACM Transactions on Interactive Intelligent Systems*, 9(4), 2019.

[16] Y. Chen and J. Miyazaki. A model-agnostic recommendation explanation system based on knowledge graph. In *Database and Expert Systems Applications*, pages 149–163, Cham, 2020. Springer International Publishing.

[17] L. Coba, M. Zanker, L. Rook, and P. Symeonidis. Exploring users' perception of collaborative explanation styles. In *2018 IEEE 20th Conference on Business Informatics (CBI)*, volume 01, pages 70–78. IEEE, 2018.

[18] R. Confalonieri, T. Weyde, T. R. Besold, and F. Moscoso del Prado Martín. Using ontologies to enhance human understandability of global post-hoc explanations of black-box models. *Artificial Intelligence*, 2021.

[19] V. Dominguez, I. Donoso-Guzmán, P. Messina, and D. Parra. Algorithmic and hci aspects for explaining recommendations of artistic images. *ACM Transactions on Interactive Intelligent Systems*, 10(4), 2020.

[20] V. Dominguez, P. Messina, I. Donoso-Guzmán, and D. Parra. The effect of explanations and algorithmic accuracy on visual recommender systems of artistic images. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, IUI '19, page 408–416, New York, NY, USA, 2019. Association for Computing Machinery.

[21] Y. Du, S. Ranwez, N. Sutton-Charani, and V. Ranwez. Post-hoc recommendation explanations through an efficient exploitation of the dbpedia category hierarchy. *Knowledge-Based Systems*, 2022.

[22] U. Ehsan, P. Wintersberger, Q. V. Liao, M. Mara, M. Streit, S. Wachter, A. Riener, and M. O. Riedl. Operationalizing human-centered perspectives in explainable ai. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA, 2021. Association for Computing Machinery.

[23] M. Eiband, H. Schneider, M. Bilandzic, J. Fazekas-Con, M. Haug, and H. Hussmann. Bringing transparency design into practice. In *23rd International Conference on Intelligent User Interfaces*, IUI '18, page 211–223, New York, NY, USA, 2018. Association for Computing Machinery.

[24] V. Eisenstadt, C. Espinoza-Stapelfeld, A. Mikyas, and K.-D. Althoff. Explainable distributed case-based support systems: Patterns for

enhancement and validation of design recommendations. In *Case-Based Reasoning Research and Development*, pages 78–94, Cham, 2018. Springer International Publishing.

[25] A. Felfernig and B. Gula. An empirical study on consumer behavior in the interaction with knowledge-based recommender applications. In *The 8th IEEE International Conference on E-Commerce Technology and The 3rd IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services (CEC/EEE'06)*, pages 37–37. IEEE, 2006.

[26] R. Flesch. A new readability yardstick. *Journal of applied psychology*, 32(3):221, 1948.

[27] F. Gedikli, D. Jannach, and M. Ge. How should i explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, 72(4):367–382, 2014.

[28] D. George and P. Mallery. Spss for windows step by step: A simple guide and reference. *Boston: Allyn & Bacon*, pages 8–10, 2003.

[29] B. Ghai, Q. V. Liao, Y. Zhang, R. Bellamy, and K. Mueller. Explainable active learning (xal): Toward ai explanations as interfaces for machine teachers. *Proc. ACM Hum.-Comput. Interact.*, 4, 2021.

[30] M. S. Gönül, D. Önkal, and M. Lawrence. The effects of structural characteristics of explanations on use of a DSS. *Decision Support Systems*, 42(3):1481–1493, 2006.

[31] M. Harbers, K. v. d. Bosch, and J.-J. C. Meyer. A study into preferred explanations of virtual agent behavior. In *International Workshop on Intelligent Virtual Agents*, pages 132–145. Springer, 2009.

[32] D. C. Hernandez-Bocanegra and J. Ziegler. Effects of interactivity and presentation on review-based explanations for recommendations. In *Human-Computer Interaction – INTERACT 2021*, pages 597–618, Cham, 2021. Springer International Publishing.

[33] R. R. Hoffman, G. Klein, and S. T. Mueller. Explaining explanation for "explainable ai". *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 62(1):197–201, 2018.

[34] T. Honglei, S. Wei, and Z. Yanan. The research on software metrics and software complexity metrics. In *2009 International Forum on Computer Science-Technology and Applications*, volume 1, pages 131–136. IEEE, 2009.

[35] J. Hunt and C. Price. Explaining qualitative diagnosis. *Engineering Applications of Artificial Intelligence*, 1(3):161–169, 1988.

[36] Institute of Electrical and Electronics Engineers. IEEE standard for a software quality metrics methodology. *IEEE Std 1061-1992*, pages 1–96, 1993.

[37] R. Iyer, Y. Li, H. Li, M. Lewis, R. Sundar, and K. Sycara. Transparency and explanation in deep reinforcement learning neural networks. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 144–150, New York, NY, USA, 2018. Association for Computing Machinery.

[38] S. Karga and M. Satratzemi. Using explanations for recommender systems in learning design settings to enhance teachers' acceptance and perceived experience. *Education and Information Technologies*, 24(5):2953–2974, 2019.

[39] R. Kass and T. Finin. The need for user models in generating expert system explanation. *International Journal of Expert Systems*, 1(4):345–375, 1988.

[40] B. Kim, C. Rudin, and J. Shah. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 1952–1960, Cambridge, MA, USA, 2014. MIT Press.

[41] A. Kleinerman, A. Rosenfeld, F. Ricci, and S. Kraus. Supporting users in finding successful matches in reciprocal recommender systems. *User Modeling and User-Adapted Interaction*, 31(3):541–589, 2021.

[42] P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, ICML'17, page 1885–1894. PMLR, 2017.

[43] T. K. Koo and M. Y. Li. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, 15(2):155–163, 2016.

[44] P. Kouki, J. Schaffer, J. Pujara, J. O'Donovan, and L. Getoor. Personalized explanations for hybrid recommender systems. In *Proceedings of the 24th International Conference on Intelligent User*

Interfaces, IUI '19, page 379–390, New York, NY, USA, 2019. Association for Computing Machinery.

[45] P. Kouki, J. Schaffer, J. Pujara, J. O'Donovan, and L. Getoor. Generating and understanding personalized explanations in hybrid recommender systems. *ACM Transactions on Interactive Intelligent Systems*, 10(4), 2020.

[46] M. Krishnan. Against interpretability: a critical examination of the interpretability problem in machine learning. *Philosophy & Technology*, 33(3):487–502, 2020.

[47] T. Kulesza, S. Stumpf, M. Burnett, and I. Kwan. Tell me more? the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 1–10, New York, NY, USA, 2012. Association for Computing Machinery.

[48] M. A. Köhl, K. Baum, M. Langer, D. Oster, T. Speith, and D. Bohlender. Explainability as a non-functional requirement. In *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pages 363–368. IEEE, 2019.

[49] I. Lage, E. Chen, J. He, M. Narayanan, B. Kim, S. Gershman, and F. Doshi-Velez. An evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1902.00006*, 2019.

[50] H. Lakkaraju, E. Kamar, R. Caruana, and J. Leskovec. Interpretable & explorable approximations of black box models. *arXiv preprint arXiv:1707.01154*, 2017.

[51] M. Langer, K. Baum, K. Hartmann, S. Hessel, T. Speith, and J. Wahl. Explainability auditing for intelligent systems: A rationale for multi-disciplinary perspectives. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 164–168. IEEE, 2021.

[52] M. Langer, D. Oster, T. Speith, H. Hermanns, L. Kästner, E. Schmidt, A. Sesing, and K. Baum. What do we want from explainable artificial intelligence (xai)? – a stakeholder perspective on xai and a conceptual model guiding interdisciplinary xai research. *Artificial Intelligence*, 296, 2021.

[53] M. Li and S. Gregor. Outcomes of effective explanations: Empowering citizens through online advice. *Decision Support Systems*, 52(1):119–132, 2011.

[54] B. Y. Lim and A. K. Dey. Assessing demand for intelligibility in context-aware applications. In *Proceedings of the 11th International Conference on Ubiquitous Computing*, UbiComp '09, page 195–204, New York, NY, USA, 2009. Association for Computing Machinery.

[55] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[56] T. Mladenova. Software quality metrics – research, analysis and recommendation. In *2020 International Conference Automatics and Informatics (ICAI)*, pages 1–5. IEEE, 2020.

[57] S. Mohseni, N. Zarei, and E. D. Ragan. A multidisciplinary survey and framework for design and evaluation of explainable ai systems. *ACM Transactions on Interactive Intelligent Systems*, 11(3–4), 2021.

[58] K. I. Muhammad, A. Lawlor, and B. Smyth. A live-user study of opinionated explanations for recommender systems. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, IUI '16, page 256–260, New York, NY, USA, 2016. Association for Computing Machinery.

[59] C. Musto, M. de Gemmis, P. Lops, and G. Semeraro. Generating post hoc review-based natural language justifications for recommender systems. *User Modeling and User-Adapted Interaction*, 31(3):629–673, 2021.

[60] S. Nagulendra and J. Vassileva. Providing awareness, explanation and control of personalized filtering in a social networking site. *Information Systems Frontiers*, 18(1):145–158, 2016.

[61] R. Nakatsu and I. Benbasat. Improving the explanatory power of knowledge-based systems: an investigation of content and interface-based enhancements. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 33(3):344–357, 2003.

[62] S. Naveed, T. Donkers, and J. Ziegler. Argumentation-based explanations in recommender systems: Conceptual framework and empirical results. In *Adjunct Publication of the 26th Conference on User*

Modeling, Adaptation and Personalization, UMAP '18, page 293–298, New York, NY, USA, 2018. Association for Computing Machinery.

[63] I. Nunes and D. Jannach. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction*, 27(3):393–444, 2017.

[64] I. Nunes, P. Taylor, L. Barakat, N. Griffiths, and S. Miles. Explaining reputation assessments. *International Journal of Human-Computer Studies*, 123:1–17, 2019.

[65] J. Ooge, S. Kato, and K. Verbert. Explaining recommendations in e-learning: Effects on adolescents' trust. In *27th International Conference on Intelligent User Interfaces*, IUI '22, page 93–105, New York, NY, USA, 2022. Association for Computing Machinery.

[66] J. Ooge and K. Verbert. Explaining artificial intelligence with tailored interactive visualisations. In *27th International Conference on Intelligent User Interfaces*, IUI '22 Companion, page 120–123, New York, NY, USA, 2022. Association for Computing Machinery.

[67] A. Papadimitriou, P. Symeonidis, and Y. Manolopoulos. A generalized taxonomy of explanations styles for traditional and social recommender systems. *Data Mining and Knowledge Discovery*, 2012.

[68] G. Penha, E. Krikon, and V. Murdock. Pairwise review-based explanations for voice product search. In *ACM SIGIR Conference on Human Information Interaction and Retrieval*, CHIIR '22, page 300–304, New York, NY, USA, 2022. Association for Computing Machinery.

[69] A. D. Preece, D. Harborne, D. Braines, R. Tomsett, and S. Chakraborty. Stakeholders in explainable AI. *arXiv preprint arXiv:1810.00184*, 2018.

[70] Z. Qi, S. Khorram, and L. Fuxin. Embedding deep networks into visual explanations. *Artificial Intelligence*, 292, 2021.

[71] R. Ramberg. Construing and testing explanations in a complex domain. *Computers in Human Behavior*, 12(1):29–48, 1996.

[72] M. T. Ribeiro, S. Singh, and C. Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

[73] A. Rosenfeld and A. Richardson. Explainability in human–agent systems. *Autonomous Agents and Multi-Agent Systems*, 33(6):673–705, 2019.

[74] T. Schrills and T. Franke. Color for characters - effects of visual explanations of ai on trust and observability. In *Artificial Intelligence in HCI*, pages 121–135, Cham, 2020. Springer International Publishing.

[75] P. E. Shrout and J. L. Fleiss. Intraclass correlations: Uses in assessing rater reliability. *Psychological bulletin*, 86(2):420–428, 1979.

[76] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[77] K. Sokol and P. Flach. Explainability fact sheets: A framework for systematic assessment of explainable approaches. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 56–67, New York, NY, USA, 2020. Association for Computing Machinery.

[78] H. J. Suermondt and G. F. Cooper. An evaluation of explanations of probabilistic inference. *Computers and Biomedical Research*, 26(3):242–254, 1993.

[79] H. Suresh, K. M. Lewis, J. Guttag, and A. Satyanarayan. Intuitively assessing ml model reliability through example-based explanations and editing model inputs. In *27th International Conference on Intelligent User Interfaces*, IUI '22, page 767–781, New York, NY, USA, 2022. Association for Computing Machinery.

[80] M. Szymanski, M. Millecamp, and K. Verbert. Visual, textual or hybrid: The effect of user expertise on different explanations. In *26th International Conference on Intelligent User Interfaces*, IUI '21, page 109–119, New York, NY, USA, 2021. Association for Computing Machinery.

[81] K. Takami, Y. Dai, B. Flanagan, and H. Ogata. Educational explainable recommender usage and its effectiveness in high school summer vacation assignment. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, LAK22, page 458–464, New York, NY, USA, 2022. Association for Computing Machinery.

[82] W.-K. Tan, C.-H. Tan, and H.-H. Teo. Consumer-based decision aid that explains which to buy: Decision confirmation or overconfidence bias? *Decision Support Systems*, 53(1):127–141, 2012.

[83] Y. Tao, Y. Jia, N. Wang, and H. Wang. The fact: Taming latent factor models for explainability with factorization trees. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 295–304, New York, NY, USA, 2019. Association for Computing Machinery.

[84] N. Tintarev and J. Masthoff. Designing and evaluating explanations for recommender systems. In *Recommender systems handbook*, pages 479–510. Springer, 2011.

[85] N. Tintarev and J. Masthoff. Evaluating the effectiveness of explanations for recommender systems. *User Modeling and User-Adapted Interaction*, 22(4):399–439, 2012.

[86] T. N. T. Tran, M. Atas, A. Felfernig, V. M. Le, R. Samer, and M. Stettinger. Towards social choice-based explanations in group recommender systems. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '19, page 13–21, New York, NY, USA, 2019. Association for Computing Machinery.

[87] C.-H. Tsai and P. Brusilovsky. Evaluating visual explanations for similarity-based recommendations: User perception and performance. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '19, page 22–30, New York, NY, USA, 2019. Association for Computing Machinery.

[88] G. Vilone and L. Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106, 2021.

[89] A. Vultureanu-Albişi and C. Bădică. Recommender systems: An explainable ai perspective. In *2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, pages 1–6. IEEE, 2021.

[90] N. Wang, D. V. Pynadath, and S. G. Hill. Trust calibration within a human-robot team: Comparing automatically generated explanations. In *11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 109–116. IEEE, 2016.

[91] W. Wang and I. Benbasat. Recommendation agents for electronic commerce: Effects of explanation facilities on trusting beliefs. *Journal of Management Information Systems*, 23(4):217–246, 2007.

[92] W. Wang, L. Qiu, D. Kim, and I. Benbasat. Effects of rational and social appeals of online recommendation agents on cognition- and affect-based trust. *Decision Support Systems*, 86:48–60, 2016.

[93] X. Wang, Y. Chen, J. Yang, L. Wu, Z. Wu, and X. Xie. A reinforcement learning framework for explainable recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 587–596. IEEE, 2018.

[94] G. Wiegand, M. Schmidmaier, T. Weber, Y. Liu, and H. Hussmann. I drive - you trust: Explaining driving behavior of autonomous cars. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019. Association for Computing Machinery.

[95] L. R. Ye. The value of explanation in expert systems for auditing: An experimental investigation. *Expert Systems with Applications*, 9(4):543–556, 1995. Expert systems in accounting, auditing, and finance.

[96] R. Yu, Z. Pardos, H. Chau, and P. Brusilovsky. Orienting students to course recommendations using three types of explanation. In *Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '21, page 238–245, New York, NY, USA, 2021. Association for Computing Machinery.