**Gottfried Wilhelm
Leibniz University Hannover
Faculty of Electrical Engineering and Computer Science
Institute of Practical Computer Science
Software Engineering Department**

Concept and Implementation of Trigger-Based Explanations

Master Thesis

in the Computer Science degree program

by

Gitta Kirana Anindya Ekaputeri

**Examiner: Prof. Dr. rer. nat. Kurt Schneider
Secondary Examiner: Dr. rer. nat. Jil Ann-Christin Klünder
Supervisor: Hannah Luca Deters, M. Sc.**

Hannover, 22.04.2024

Erklärung der Selbständigkeit

Hiermit versichere ich, dass ich die vorliegende Master Thesis 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 22.04.2024

Gitta Kirana Anindya Ekaputeri

Abstract

Explainability could serve as a means of achieving goals such as understandability and transparency. Previous research suggests that explanations are intended to guide users and improve their experience, but can have adverse effects if not delivered correctly. Therefore, it is important to strategically integrate explanations to ensure that users receive relevant information when they need it. This thesis aims to explore trigger-based explanations, focusing primarily on identifying potential triggers that could signal the need for an explanation. This research focuses on identifying triggers, evaluating their effectiveness in providing timely and contextually relevant information, and their impact on the user experience. A literature review and workshop were conducted to explore user needs for explanation. Afterward, four triggers were selected and conceptualized: first-time use, repeated errors, task interruption, and repetitive action. These triggers were then integrated into a prototype and subjected to a user study with 30 participants. The evaluation included both quantitative and qualitative analyses examining pre-defined metrics. The results indicate that while some triggers successfully addressed users' need for explanations promptly and met their expectations, others were perceived as less effective, either because the explanations were delayed or absent when needed. Furthermore, the analysis and participant feedback suggest that trigger-based explanations can enhance user experience positively if delivered at the appropriate time and in the relevant context.

Zusammenfassung

Erklärbarkeit könnte als Mittel dienen, um Ziele wie Verständlichkeit und Transparenz zu erreichen. Frühere Untersuchungen haben gezeigt, dass Erklärungen zwar dazu dienen, die Nutzer zu leiten und ihre Erfahrungen zu verbessern, dass sie aber auch negative Auswirkungen haben können, wenn sie nicht richtig vermittelt werden. Daher ist es wichtig, Erklärungen strategisch zu integrieren, um sicherzustellen, dass die Nutzer die relevanten Informationen erhalten, wenn sie sie benötigen. Aus diesem Grund zielt diese Arbeit darauf ab, triggerbasierte Erklärungen zu untersuchen. Diese Masterarbeit konzentriert sich auf die Identifizierung von Triggern, die Bewertung ihrer Effektivität bei der Bereitstellung zeitnaher und kontextbezogener Informationen und ihre Auswirkungen auf die Benutzererfahrung. Eine Literaturrecherche und ein Workshop wurden durchgeführt, um den Erklärungsbedarf der Benutzer zu ermitteln. Anschließend wurden vier Trigger ausgewählt und konzeptualisiert: Erstnutzung, wiederholte Fehler, Unterbrechung der Aufgabe und wiederholte Aktion. Diese Trigger wurden dann in einen Prototyp integriert und in einer Studie mit 30 Teilnehmern untersucht. Die Bewertung umfasste sowohl quantitative als auch qualitative Analysen, bei denen vordefinierte Metriken berücksichtigt wurden. Die Ergebnisse deuten darauf hin, dass einige Trigger den Erklärungsbedarf der Nutzer unmittelbar ansprechen und deren Erwartungen erfüllen, während andere als weniger effektiv bewertet werden, weil die Erklärungen entweder zu spät oder gar nicht gegeben werden, wenn die Nutzer sie benötigen. Darüber hinaus deuten die Analyse und das Feedback der Teilnehmer darauf hin, dass triggerbasierte Erklärungen die Benutzererfahrung positiv beeinflussen können, wenn sie zum richtigen Zeitpunkt und im richtigen Kontext gegeben werden.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Solution Approach 2

1.3 Thesis Structure 2

2 Background and Related Work **3**

2.1 Explainability 3

2.1.1 Quality Impacts of Explanations 3

2.1.2 Challenges in Implementation of Explainability 4

2.2 Trigger-Driven System 5

2.3 Goal Question Metric 6

2.4 Related Work 6

3 Literature Review and Workshop **9**

3.1 Research Questions 9

3.2 Goal Question Metric (GQM) 10

3.3 Literature Review 13

3.3.1 Procedure 13

3.3.2 Results 14

3.4 Workshop 15

3.4.1 Questions 15

3.4.2 Procedure 16

3.4.3 Results 18

4 Concept **27**

4.1 Triggers 27

4.1.1 First-Time Use 28

4.1.2 Repeated Errors 28

4.1.3 Task Interruption 29

4.1.4 Repetitive Action 30

4.2 Use Case 31

4.2.1 Save a product to a collaborative collection 32

4.2.2 Create an account with a username 32

4.2.3 Change the number of items in the shopping cart . . . 33
4.2.4 Search for a specific product 33
4.3 Prototype Implementation 33

5 User Study 41
5.1 Preparation 41
5.1.1 Target Participants 42
5.1.2 Task Design 42
5.1.3 Questionnaire Design 43
5.2 Execution 43
5.2.1 Demographics 44
5.2.2 Prototype Interaction and Feedback Survey 44

6 Evaluation 47
6.1 Data Analysis Procedure 47
6.2 Quantitative Data Analysis 48
6.2.1 First-time Use - Save a product to a collaborative collection 48
6.2.2 Repeated Errors - Create an account 50
6.2.3 Task Interruption - Change the number of items in the shopping cart 54
6.2.4 Repetitive Action - Search for a specific product . . . 56
6.3 Qualitative Data Analysis 58
6.3.1 First-time Use - Save a product to a collaborative collection 58
6.3.2 Repeated Errors - Create an account 59
6.3.3 Task Interruption - Change the number of items in the shopping cart 60
6.3.4 Repetitive Action - Search for a specific product . . . 60
6.3.5 Trigger-based explanations in general 61

7 Discussion 63
7.1 Answers to the Research Questions 63
7.2 Limitations and Threats to Validity 67

8 Conclusion and Future Works 71
8.1 Conclusion 71
8.2 Future Works 72

A Task Instructions 75

B Questionnaire 79

Chapter 1

Introduction

No matter how useful a system may be, a user's willingness to utilize it may diminish if they are unable to comprehend how it works. This poses a significant challenge as software systems advance: how can a system maintain user friendliness while expanding in functionality? Users look for efficient tools to help them accomplish their tasks, and many systems offer an array of features to meet diverse user needs. However, complex software systems can overwhelm and frustrate users. A good software application should meet both functional and non-functional requirements. One of the critical non-functional requirements in software quality is explainability. This means, aside from offering features that aid users in completing their tasks, well-developed software should also facilitate efficiency and accuracy by providing clear guidance and insights into the inner workings of the system. Therefore, integrating well-crafted explanations into software systems enhances user support and elevates the overall user experience.

1.1 Motivation

Although explanations are intended to guide the user and positively influence their experience, they can have a negative impact if not provided correctly [4, 8, 9, 10]. For instance, explanations can be distracting if not given at the appropriate time and situation [4]. For this reason, it is crucial to implement explanations strategically to ensure that users receive guidance or information when needed, at the right time, and in the right context. This thesis aims to investigate trigger-based explanations, recognizing the importance of timing in optimizing user experiences. To the best of our knowledge, no prior research has been conducted on this specific topic. Therefore, this thesis introduces different triggers for the concept of trigger-based explanations and evaluates them through a user study.

1.2 Solution Approach

The solution approach began with a literature search to identify existing studies and findings. However, the literature proved to be inadequate, as no prior research had comprehensively covered the target topic. To overcome this limitation, a workshop was organized to generate and gather ideas regarding potential triggers for explanations in software systems. Based on the outcomes of this collaborative session, a prototype was implemented, incorporating various use cases aligned with each identified trigger. The prototype demonstrates the practical application of triggers and serves as an interactive representation of the intended concept. Additionally, a user study was conducted where participants were asked to complete predefined tasks using the prototype. During the process, explanations will be provided to them if the corresponding triggers are activated. This user study is instrumental in assessing the impact of these triggers on the user experience, providing valuable insights into their effectiveness. This approach allows for systematic evaluation and analysis of whether the conceptualization and implementation of trigger-based explanations are feasible and suitable within the examined context.

1.3 Thesis Structure

This thesis consists of eight chapters. After this introductory chapter, chapter 2 provides fundamental knowledge and some research relevant to this topic, giving readers a contextual understanding of the subject. Thereafter, chapter 3 presents the research questions and introduces the Goal Question Metric as the framework for addressing them. This chapter also includes the methodology and findings from the literature review, as well as insights from the workshop conducted. In chapter 4, I define the possible triggers of explanations and how the prototype is implemented. Furthermore, chapter 5 explains how the user study was conducted and presents its results. These results are then analyzed in chapter 6. Afterwards, in chapter 7, I answer the research questions stated in chapter 3 and discuss limitations and threats to the research's validity. Finally, chapter 8 provides a summary of the thesis and presents ideas and suggestions for future research on this topic.

Chapter 2

Background and Related Work

2.1 Explainability

Liu et al. [16] describe explainability as the ability to provide understandable reasonings for the decisions, recommendations, and predictions that software generates. Further, Droste et al. [11] suggest the importance of addressing broader explainability requirements beyond interpretability, such as identifying needs for explanations in privacy, system interaction, and domain-specific information. Moreover, explainability depends on the context and goals of the explainer based on the requirements [10]. A more precise definition is provided by Chazette et al. [4] in the context of software and requirements engineering, where they state that "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". This suggests that different individuals with varying goals may need explanations at different times, such as how certain operations work within the system or why certain errors occur. This concept is significant in various domains, such as artificial intelligence (AI) and machine learning (ML), where understanding the explanations that support a model's output is crucial, especially as AI-powered systems continue to advance without much human intervention [1]. Understanding how decisions are made is important, for example, so that errors or biases in the model's reasoning can be identified. In the pursuit of achieving explainability, explanations serve as mechanisms that offer insight into how and why a particular decision was made or a specific output was generated, making the system's behavior more transparent and understandable to users, developers, or other stakeholders.

2.1.1 Quality Impacts of Explanations

Explanations have an impact on a variety of quality goals, such as transparency [4, 10], satisfaction [4, 10], understandability [4, 10, 17] and can serve as a means to achieve these goals [4, 10]. Within complex software systems, explanations can help users understand the behavior of a system and also serve as guidance to help users navigate and use the software's functionality. Contextual guidance, such as tutorials or tool tips, helps users navigate menus and reduces their learning curve.

Moreover, explanations not only help users understand the inner workings of a system [10], but also allow for faster familiarization, facilitate faster judgments, prevent users from making mistakes, and assist them in decision-making situations [8]. Explanations provide users with the encouragement and comfort to freely navigate and explore program features by providing them with the knowledge necessary to use the features. In doing so, they also prevent underutilization and user reliance on basic features.

Furthermore, effective explanations are critical to minimizing user frustration by ensuring that each feature is accompanied by an easy-to-understand description. When users encounter difficulties or errors, well-written explanations can guide them through troubleshooting processes. After all, functional and feature-rich software is worthless if people lose confidence and abandon it after a few attempts. In this case, providing explanations can lead to a reduction in support requests because users are likely to be able to find solutions on their own. In summary, clear explanations contribute to greater satisfaction [4, 10] and trust [8, 10] among users. They are essential for successful software development in achieving its quality goals and should therefore be considered as a means to an end [10].

2.1.2 Challenges in Implementation of Explainability

While explainability is generally expected to have a positive impact, it is a double-edged sword that can have both positive and negative effects on the user experience [5, 8, 9, 10]. For example, while increased transparency may seem beneficial, it can also lead to decreased comprehension and user satisfaction [8]. In a study conducted by Chazette et al. [8], some participants expressed concern that explanations could disrupt the workflow and be too distracting.

Developing user-friendly interfaces and seamless interactions is critical to optimizing the user experience. However, when introducing complex features, there are many challenges in providing appropriate explanations to software users. First, identifying the explainability requirements is not a simple task, as it depends on several factors, such as the quality goals to be achieved [10], user and contextual considerations [9], and the entity responsible for providing the explanation [4]. For example, because users have different needs and preferences [4, 10], the usefulness of an explanation depends on its recipient [19, 20]. Consequently, explanations that are useful to one user may be confusing or even disruptive to another.

Another challenge is to ensure that explanations are well-written.

Unnecessary or irrelevant information not only fails to engage users but also detracts from the user experience [8]. In addition, striking a balance between providing comprehensive explanations and maintaining simplicity and clarity is not easy. If explanations are not well-elaborated, they might result in additional development costs and impose unnecessary cognitive burdens on users [9], or even hinder the user from understanding them [8].

Timing must also be taken into account because users have different preferences about when explanations should be provided [8]. To increase user satisfaction when using the system, explanations should be easy to discover and non-disruptive [10]. While it is possible to simply provide explanations whenever they might occur, this approach will increase the burden on users and hence is not a satisfying option [9]. Users are more interested in explanations when they encounter unexpected system behavior, especially in scenarios where making quick and accurate decisions is critical [8]. It is important to present information at the right moment and in the appropriate context to avoid distraction and interruption of a task. If information is not presented in the right way, users will become distracted and therefore cannot finish their tasks efficiently. Even worse, users may be frustrated and abandon the system altogether. As valuable as the information presented is, the way it is implemented should be taken into account so that it can fulfill the goals while creating as few negative effects as possible [10].

2.2 Trigger-Driven System

A trigger-driven system operates on the principle of timely responses to specific events. These events, known as trigger conditions, prompt the system to carry out predefined actions automatically when they occur [15]. Each trigger is associated with one or a set of actions, which will be executed in response to the trigger event. A trigger can come from a variety of contexts, from user interactions to system events and external inputs. For instance, a response might react to a user clicking a button, encountering multiple errors, or aborting a task midway through the process. In each scenario, the system responds dynamically by executing predefined actions to maintain seamless functionality and user experience. This responsiveness ensures that users receive timely and relevant feedback or assistance based on their interactions with the system. Such a system could offer a personalized approach to users based on their behavior [15]. Overall, trigger-driven systems play a crucial role in enabling efficient and personalized interactions between users and software applications.

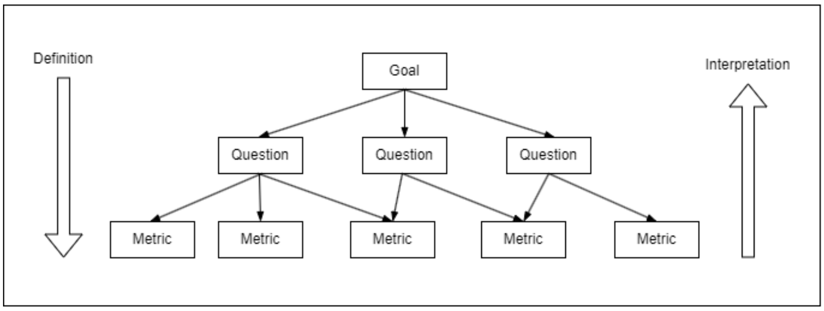


Figure 2.1: Representation of Goal-Question-Metric Framework

2.3 Goal Question Metric

The Goal-Question-Metric (GQM) model is a structured approach to measuring software process or quality, which is illustrated in Figure 2.1. The approach begins by defining a specific goal that includes the purpose, object, question, and viewpoint of the measurement [3]. This approach ensures that different viewpoints are considered and allows for the measurement of different perspectives on the same issue. In a top-down approach, the goal is then broken down into related questions and measurable metrics to address the goal. These questions can be either quantitative or qualitative, and the data collection will be done accordingly. In addition, a single metric can address multiple questions within the same objective. Finally, after measurement, the results can be interpreted from the bottom up [14].

By identifying specific goals related to software quality, GQM facilitates the formulation of targeted questions to assess these aspects. Furthermore, by aligning goals and metrics with specific measurement settings, unnecessary data collection efforts are minimized and the interpretation of metrics becomes more straightforward, reducing the risk of misinterpretation [14].

2.4 Related Work

Previous studies in the field of explainability in software systems have been focusing on understanding the importance and impact of explainability in software systems, as well as the development of methodologies and frameworks to evaluate and integrate this explainability effectively. Among these studies, there are differences in the specific approaches taken, such as user-centered design, heuristic evaluations, framework development. In 2020, Chazette et al. [8] investigated the relationship between explainability and usability, offering concrete recommendations for reconciling explanations with usability through user-centered design (UCD) techniques. They emphasized the importance of aligning requirements with user needs and context to enhance system usability and effectiveness. In the following year, Chazette et al. [4] explored how explainability interacts with various quality aspects and its impact on different dimensions of system quality.

Furthermore, Chazette et al. [7] introduced a quality framework for explainability, providing guidelines to integrate explainability into systems effectively. This framework proved to enhance usage frequency, system acceptance, and user satisfaction. Another comprehensive framework was proposed by Chazette et al. [5] for explainable systems, consisting of 4 artifacts: a definition of explainability, a conceptual model, a knowledge catalogue, and a reference model. They highlighted explainability as a preferred solution to address system transparency issues. Moreover, Droste et al. [13] suggested the creation of end-user personas for general explainability requirements, aiding in estimating user needs early in the development process. These personas facilitate the identification of specific explainability requirements as development progresses and have been shown to be effective in eliciting and communicating requirements.

In 2023 Deters et al. [10] explored methods for evaluating explainability in software systems, aligning with intended goals of explanations. They developed ten heuristics grouped into four categories: understandability, transparency, satisfaction, and suitability. In the same year, Deters et al. [9] conducted a user study allowing users to immerse themselves realistically in the system's context where explanations are provided upon request. The results revealed that users have different needs for explanations, depending on individual characteristics and prior knowledge, suggesting the need for tailored explanations.

Based on prior research, it is evident that quality goals can be accomplished through various methods or frameworks of providing explanations and that addressing diverse user needs is crucial. These studies emphasized the significance of considering users during the explanation development process. Building upon this foundation to enhance explainability further, this thesis will explore the ideal timing for delivering explanations via triggers. To the best of our knowledge, there has not yet been any research that covers suitable triggers for explanations. This thesis will focus on trigger-based explanations, implementing and evaluating their impact on user experience.

Chapter 3

Literature Review and Workshop

Before beginning the conceptualization phase of the triggers, several preparatory steps were required to lay a solid foundation and gather initial ideas. This chapter explains these steps, starting with defining the research questions. Afterward, the goal-question-metrics (GQM) were determined to identify appropriate questions and metrics for evaluation. Next, a literature review was conducted to identify relevant studies and publications that may provide insight into the topic. The goal was to discover existing findings that may contribute to answering the research questions and potentially serve as a basis for conceptualization. Finally, a workshop with brainstorming sessions was carried out to also generate preliminary ideas.

3.1 Research Questions

The main goals of this thesis are to conceptualize and implement trigger-based explanations. By formulating research questions and keeping them in mind throughout the research process, a concept will be developed, implemented, and evaluated with the goal of answering these questions. In order to understand the significance and effectiveness of trigger-based explanations and their impact on the user experience, the following research questions will be answered in this thesis:

RQ 1 What are the possible indications that users need an explanation?

This question aims to identify specific user behaviors that indicate the user needs an explanation. By understanding the user's perspective and identifying the scenarios where a user needs explanations, it provides a basis for determining potential triggers for explanations. Moreover, this question promotes a user-centered approach to software design, emphasizing the importance of comprehending how end users engage with the system. In essence, by addressing this question first, a concept of trigger-based explanations can be developed and implemented in appropriate contexts. This thesis focuses on interaction explanation needs. Within the taxonomy of explainability needs outlined by Droste et al. [12], explanations in the interaction category emerged as the most needed in everyday software systems. This category includes the need to understand the procedures to execute specific operations, the need for explanation regarding navigation within the software, as well as the use of new features.

RQ 2 What effects do trigger-based explanations have on the overall user experience?

- RQ 2.1 How do users evaluate the impact of trigger-based explanations on their user experience?
- RQ 2.2 How do trigger-based explanations influence task completion outcomes?

RQ 2 delves into the implications of trigger-based explanations on the user experience. Studying the effects of trigger-based explanations allows us to discern whether they positively contribute to the user experience, for example by fostering understanding and trust, or instead negatively impact the user experience. This insight is invaluable for guiding design decisions and optimizing the integration of trigger-based explanations into software systems to maximize their benefits. While RQ 2.1 evaluates this effect from the perspective of users, RQ 2.2 shifts the focus from subjective user opinions to objective task performance. By examining the impact of trigger-based explanations on task completion outcomes such as efficiency and effectiveness, one can assess their practical effectiveness in facilitating user interactions and achieving desired goals. This empirical evaluation provides concrete evidence of the practical benefits or drawbacks of incorporating trigger-based explanations into software systems.

3.2 Goal Question Metric (GQM)

In this thesis, trigger-based explanations are evaluated in terms of suitability, effectiveness, efficiency, and user satisfaction, as these quality characteristics capture important aspects of their performance and impact. While other quality aspects may also be relevant to trigger-based explanations, these four aspects are particularly crucial due to their direct connection to the core objectives of trigger-based explanations. The goal is to provide contextually relevant assistance to users during their interaction with the software system without being intrusive or distracting to them. These aspects focus on ensuring that the explanations are appropriate to the user's needs, effective in achieving their intended purposes, timely, and have a positive impact on the user experience.

Suitability refers to the extent to which trigger-based explanations are appropriate for addressing user needs and are provided in the right contexts within the software system. By evaluating suitability, we can determine whether the timing of the explanations is well-suited to the specific situations in which they are provided and helps achieve the intended purpose. Furthermore, evaluating the effectiveness of trigger-based explanations determines whether they achieve their intended goals, such as improving user understanding and facilitating decision-making. This will measure the degree to which trigger-based explanations assist users in making better decisions or utilizing functionalities that align most with their needs.

Another thing that is evaluated is the impact of trigger-based explanations on efficiency. Efficiency evaluation can determine whether trigger-based explanations are delivered in a timely manner and help users execute tasks faster. Lastly, the user satisfaction rating assesses the degree to which users are content with the trigger-based explanations provided. It includes factors such as helpfulness, overall user experience, and how well trigger-based explanations meet user expectations. Overall, these insights into all four aspects are essential for validating the value of trigger-based explanations and identifying areas for improvement. These aspects are evaluated using the following goal-question-metrics:

G1 Evaluate the suitability of each selected trigger during task completion

  Q1 When did users anticipate receiving an explanation but no explanation was provided?

    M1 Number of instances where users anticipated an explanation but did not receive one.

    M2 Instances or contexts where additional guidance was needed.

  Q2 How do users perceive the relevance of each explanation in the given context?

    M3 User ratings on the alignment of the provided explanations with their expectations regarding guidance.

    M4 Percentage of total trigger activations among users who anticipated guidance.

    M5 Percentage of total trigger activations among users who did not anticipate guidance.

- M6 User ratings on the timing.
- M7 User feedback on the timing.
- M8 User ratings on the helpfulness.
- M9 User ratings on the helpfulness among users who did not anticipate guidance.
- M10 Analysis of reasons why users did not read the given explanation.

G2 Evaluate the effectiveness of each selected trigger during task completion

- Q3 How do users perceive the effectiveness of each explanation?
  - M8 User ratings on the helpfulness.
  - M11 Proportion of users who understood the explanation.
  - M12 User ratings on the impact of the explanation on users' overall experience.
  - M13 Ratio between successful and unsuccessful task completions after users read the explanation.
  - M14 Correlation between reading explanation and successful task completion.
  - M15 Proportion of users who apply the given guidance in the subsequent tasks.

G3 Evaluate the efficiency of each selected trigger during task completion

- Q4 How does each explanation affect the user's ability to complete tasks more quickly?
  - M13 Ratio between successful and unsuccessful task completions after users read the explanation.
  - M16 Analysis of total time spent to complete tasks between users who read and did not read explanations.

G4 Evaluate the user satisfaction of each selected trigger during task completion

- Q5 How do users rate their satisfaction with each trigger-based explanation?
  - M3 User ratings on the alignment of the provided explanations with their expectations regarding guidance.
  - M6 User ratings on the timing.
  - M7 User feedback on the timing.
  - M8 User ratings on the helpfulness.
  - M12 User ratings on the impact of the explanation on users' overall experience.
  - M17 User satisfaction ratings.

3.3 Literature Review

3.3.1 Procedure

During the literature review, research papers were searched via Google Scholar and Elicit. While Google Scholar retrieves relevant documents based on keyword-based searches, Elicit is a paper discovery tool using artificial intelligence that also identifies related papers using semantic similarity. That way, papers related to the given search keywords can also be discovered, even if they have different keywords. In addition, Elicit also automatically explores forwards and backwards in the citation graph to uncover more relevant papers. In the literature review process for this study, the following search strings were used:

- ("explanation need" OR "guidance need" OR "explainability need") AND (software OR system OR application)
- (condition OR circumstance OR time OR indication) AND (explanation OR guidance OR explainability) AND need AND (software OR system OR application)

Furthermore, to ensure relevance in the search process, the paper must fulfill all of the defined inclusion criteria and none of the exclusion criteria to be selected. Since it was difficult to find papers that explicitly mention triggers or signs that a user needs an explanation, papers that describe scenarios regarding explanation needs were also selected. It was believed that these papers would still provide insightful findings for the further steps of this thesis. The inclusion and exclusion criteria are as follows:

Inclusion Criteria:

1. The paper must explain at least one of the following topics:
   - 1.1 Indication or trigger that software users need an explanation
   - 1.2 Explanation needs of users in software systems
2. The paper is peer-reviewed.

Exclusion Criteria:

1. The paper is not written in English.

In the process of filtering relevant papers, the first step was to exclude papers that were irrelevant by their title. Then, the abstract and conclusion of the remaining papers were further reviewed. After this process, the papers that were found to be relevant were read in more detail. Due to the difficulty in finding a sufficient number of valuable papers as a starting set from the initial searches, a proper systematic literature research was not carried out. After reviewing a number of papers and realizing that there is not that much literature covering the topic of this thesis, more papers were searched and reviewed by backward and forward snowballing from the initial papers found. Another approach taken was to search for relevant papers through the authors of the papers that were found. These flexible approaches helped to discover additional relevant literature that may not have appeared in the initial searches. Moreover, conducting a systematic literature review in this scenario might require considerable effort with limited significant results.

3.3.2 Results

The findings in this literature review were gained from the papers listed in Table 3.1. There are many different scenarios in which users need an explanation. Firstly, in scenarios where users encounter unexpected or unclear behavior from the system [22, 12, 8, 6]. These situations often arise when there is a discrepancy between user expectations and the actual presentation of the system. Other cases are, for instance, when there are privacy concerns raised by users [9, 2, 12, 21], where they may seek explanations about how their data is recorded and processed. Furthermore, explanations are necessary for assisting users in understanding how to operate the system and perform specific functions [21, 18, 9, 12]. The need for this kind of explanation was found to be the most apparent in the study conducted by Droste et al. [12].

Topic	**Paper**
Unexpected system behavior	[22, 12, 8, 6]
Privacy	[9, 2, 12, 21]
Operation	[21, 18, 9, 18, 12]
New system or feature	[12, 21]
Decision-making situation	[5, 8, 6]
Inner working of the system	[9, 4, 21]
After errors	[21, 18]
Upon request	[8, 6, 22]

Table 3.1: Overview of the findings source from the literature review

Another situation where explanations are needed is when users are faced with a new system or feature for the first time. They may seek an explanation to know how to use it [12, 21] or explanation regarding new updates, to obtain information about unfamiliar changes or modifications [21]. Whether it is learning how to use a new feature or understanding changes in the system's interface or workflow, clear explanations can mitigate confusion. Additionally, explanations play a role in decision-making scenarios, especially where users have limited time to act [5, 8, 6]. By providing insights into the mechanism of the system, explanations can facilitate better decision making [21, 9, 4].

Moreover, explanations are also needed after facing errors or failures [21, 18]. Users may inquire about the causes of errors and seek guidance on how to resolve them, thereby facilitating debugging and troubleshooting processes. Additionally, there are instances where users feel that explanations should only be provided upon request [22, 8, 6].

3.4 Workshop

Since the existing literature is not sufficient to develop a concept for trigger-based explanations, a workshop was organized to brainstorm additional ideas. This chapter introduces the main questions of the workshop and presents the procedure and results. The results of this workshop, combined with insights from the literature review, will be the basis for the conceptualization and implementation of trigger-based explanations in this thesis.

3.4.1 Questions

The primary goal of this workshop was to identify possible triggers, explore the expected impact on the user experience, and discuss potential implementation challenges. In order to achieve these goals, the following questions were defined:

Q1 What are possible triggers of explanations? How might we identify or recognize that a user needs an explanation?

Q2 How can we make sure that trigger-based explanations bring positive rather than negative impact to user experience?

Q3 Are there any potential drawbacks that might affect the user experience negatively or raise concerns among users?

Q4 Are there difficulties or challenges (technical and non-technical), that might be encountered during the implementation process?

Q5 How can we tailor the triggers to accommodate diverse user preferences?

Q1 serves as a basis for generating ideas about the triggers that might indicate the need for explanations within the software system. Identifying these indications helps to ensure that explanations are provided in relevant scenarios and meet the user's needs. During the workshop, this question was also presented in an alternative formulation (*How might we identify* *or recognize that a user needs an explanation?*) to better stimulate ideas among the participants. Moreover, Q2 raises ideas about how to ensure that trigger-based explanations have a positive impact on the user experience.

In addition, Q3 explores the potential downsides or negative impacts of trigger-based explanations, an important step in anticipating and mitigating potential disadvantages. This initiates a broader understanding of potential negative impacts. By acknowledging concerns regarding user experience, such as privacy issues or intrusiveness, we can identify areas for improvement and refinement in the design and implementation of trigger-based explanations.

Q4 focuses on identifying potential challenges that may arise during the implementation of trigger-based explanations. This involves discussing technical limitations, resource constraints, ethical considerations, or integration complexities. Addressing implementation challenges early on helps streamline the development process and ensures the scalability of the proposed ideas. Furthermore, this question assists in prioritizing triggers that are more feasible to implement within the scope of this thesis. Another important point is the consideration of the diverse needs and preferences of users, which is essential for designing trigger-based explanations. Q5 encourages participants to explore ways to tailor triggers based on factors such as user expertise, accessibility, or interaction preferences. Customizing triggers to accommodate a range of user profiles is expected to enhance the overall user experience in the long run.

In summary, discussing these questions during the workshop provides valuable insights into the design, implementation, and potential impacts of trigger-based explanations. By addressing key considerations related to triggers, user experience, implementation challenges, and user preferences, participants can collaboratively generate innovative ideas for trigger-based explanations. This goes beyond the mere conceptualization of triggers in general, ensuring that the solution prioritizes user needs and preferences.

3.4.2 Procedure

The workshop was held online and involved six participants: four computer science students and two non-computer science students from different backgrounds. This mix of perspectives was strategically chosen to ensure a well-rounded discussion, as the non-computer science students could provide direct insight into the user experience without being burdened by technical feasibility concerns, while the computer science students could provide valuable input on the technical implementation aspects. In addition, since they all use software on a daily basis, they can provide insight into scenarios where they feel explanations are needed but not provided, or not provided well.

To kick off the session, an introductory segment was provided that presented explanations in software systems in general, the challenges associated with them, and the concept of trigger-based explanations to address some of these challenges. Concrete examples were included to illustrate potential scenarios that could be encountered in real-life situations and to deepen the participants' understanding of the topic. After the introductory segment, the workshop transitioned into a collaborative brainstorming session consisting of six segments where open dialogue, sharing of perspectives, and exchange of ideas were encouraged. The process of the brainstorming session is depicted in Figure 3.1.

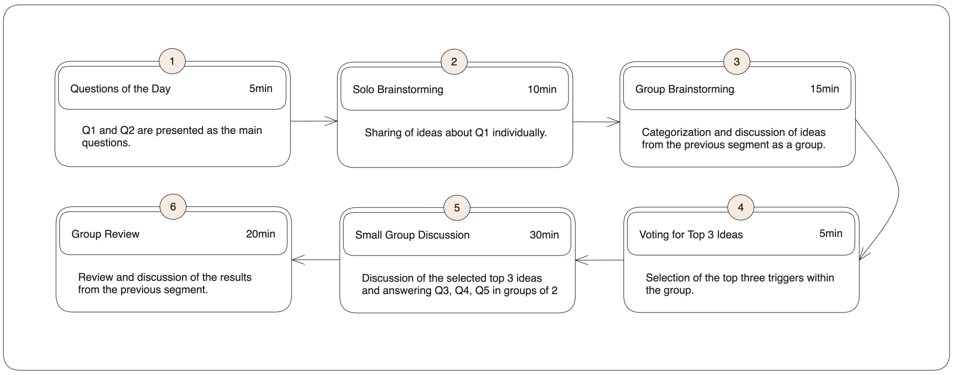

1
Questions of the Day
5min
Q1 and Q2 are presented as the main questions.
2
Solo Brainstorming
10min
Sharing of ideas about Q1 individually.
3
Group Brainstorming
15min
Categorization and discussion of ideas from the previous segment as a group.
4
Voting for Top 3 Ideas
5min
Selection of the top three triggers within the group.
5
Small Group Discussion
30min
Discussion of the selected top 3 ideas and answering Q3, Q4, Q5 in groups of 2
6
Group Review
20min
Review and discussion of the results from the previous segment.

Figure 3.1: Workflow diagram of the brainstorming session.

The session began with participants logging into the Mural Board, a digital workspace designed to facilitate collaborative brainstorming. In the first segment, Q1 and Q2 were presented to initiate the exploration of potential triggers for explanations in software systems. Following this, the second segment started, where participants engaged in solo brainstorming. Each participant was given 10 minutes to individually share their ideas about Q1 and Q2 on the Mural board. With no minimum or maximum number of ideas, participants were encouraged to freely express any ideas that came to mind.

Moving to the third segment, participants reassembled for group brainstorming. Each individual had the opportunity to review all ideas generated during the individual brainstorming session. Then, working together, participants grouped similar ideas into categories and engaged in discussions to address questions and build on each other's inputs. Any new ideas that emerged during this process were also added to the board. Then, in the fourth segment, each participant had three votes to select the top three triggers from the pool of ideas generated. Based on this voting, the group's top three triggers were determined.

In the fifth segment, participants formed small groups to delve deeper into these selected triggers. Questions related to potential drawbacks, implementation challenges, and customization for diverse user profiles (Q3, Q4, Q5) were presented to guide discussions within each group. Finally, in the sixth segment, participants came together as a large group to review and discuss the results of each small group. This facilitated a comprehensive understanding of the different perspectives and insights gathered throughout the workshop, and fostered a collective exploration of trigger-based explanations in software systems.

The collaborative setting encouraged idea sharing and creativity, allowing participants to make meaningful contributions based on their experience with software systems. By the end of the two-hour session, participants had collectively generated a number of innovative ideas and insights, providing inspiration for the conceptualization and implementation of trigger-based explanations.

3.4.3 Results

This section presents the results of the participants' collaborative brainstorming. It outlines the responses to Q1, Q2, Q3, Q4, and Q5, which explore trigger-based explanations. These responses provide a comprehensive summary of the workshop findings, as well as a conclusion drawn from the collective insights and perspectives shared during the brainstorming session.

Q1 What are possible triggers of explanations? How might we identify or recognize that a user needs an explanation?

- Inactivity: Extended periods of idleness or lack of activity.
- Mouse movements: Indicators such as aimless mouse movements or hovering over specific areas.
- Errors: Errors or repeated errors.
- Software-based: Instances like first-time usage, post-software updates, or engagement with complex software or tasks.
- Time spent or unusual duration: Extended time spent on tasks beyond typical durations.
- Other interactions: Repetitive actions, task interruptions, contacting customer support, navigating back to previous pages or steps.

In response to Q1, participants identified several instances, listed above, that indicate a need for explanation within software systems. Inactivity emerged as a potential trigger, characterized by extended periods of idleness or lack of user activity. Such instances suggest potential points where users may require additional guidance or clarification in order to navigate effectively through the software interface. Similarly, mouse movements were highlighted as potential cues for triggering explanations. Aimless mouse movements or hovering over certain areas within the software interface were identified as signals that users may have difficulty or confusion in understanding certain elements. These two occurrences may also suggest hesitation from the user.

Moreover, errors or repeated errors were recognized as key triggers, indicators that users may benefit from more detailed explanations of what is happening or troubleshooting assistance to address their challenges. Repeated errors may be an indicator that users are experiencing persistent difficulties. By providing an explanation after the user has made repeated errors, the system would be more likely to provide a relevant and helpful explanation in that particular context. In addition, it can avoid unnecessary interruptions or distractions, for instance, for users who may simply need more time to familiarize themselves with the system before needing guidance.

Furthermore, software-based triggers, such as first-time use, after software updates, or when engaging in complex tasks, were identified as key moments when users may seek additional assistance or clarification to facilitate a smoother interaction. When software undergoes updates, explanations can help inform users about new features, changes in functionality, or improvements, ensuring they are aware of the latest functionalities. Furthermore, participants also mentioned the importance of considering the duration of user interactions with the software as a trigger for explanations. Instances of extended time spent on tasks beyond typical durations were recognized as potential areas where users may require targeted explanations or guidance to overcome challenges.

Additionally, several other interactions were identified as potential triggers, including repetitive actions such as clicking the same button multiple times or repeatedly refreshing a web page. Task interruptions, such as cancelling a process or unexpected behavior that does not progress toward the task goal, were also identified as moments when users may need guidance to resume their tasks. Other instances where users contact customer support for assistance or navigate back to previous pages or steps within the software interface may also indicate uncertainty or confusion, highlighting the need for explanation.

Q2 How can we make sure that trigger-based explanations bring positive rather than negative impact to user experience?

As mentioned in Section 2.1.2, explanations can have both positive and negative effects on the user experience. Therefore, to ensure that trigger-based explanations have a positive impact on the user experience, careful consideration must be taken before, during, and after implementation. Q2 was raised to address this concern. The following are the considerations mentioned by the participants during the workshop:

Before or during implementation:

- Ensure intuitiveness.
- Strive for simplicity.
- Aim for effectiveness.
- Timing should be appropriate, ensure non-intrusiveness.
- Provide precise and concise solutions/explanations.
- Consider the type of software.
- Understand the main target demographic of the software.

After implementation:

- Monitor user progress towards the goal.
- Observe how errors are corrected by users.
- Gather feedback from users.
- Conduct expert reviews.
- Analyze statistics (e.g., overall task completion duration before and after explanations are provided).

Prior to implementation, it's crucial to prioritize intuitiveness, striving for explanations that are clear, concise, and easily understandable to users. Users should be able to grasp the meaning of the explanation without confusion or ambiguity. An intuitive explanation helps users understand why certain events occur in the software and how to proceed. Simplicity should also be pursued, avoiding unnecessary complexity that may overwhelm or confuse users. Aiming for effectiveness is important to ensure that explanations serve their intended purpose of helping users make better decisions and utilize the features.

Additionally, explanations should be non-intrusive, appearing at appropriate moments without disrupting the user's workflow. Timing plays a vital role, with explanations delivered precisely when needed to aid users in their interactions with the software. Tailoring explanations to the specific type of software and understanding the main target demographic are also essential considerations to ensure relevance and suitability. For example, in gaming applications, explanations are better provided gradually as the levels progress and interactively integrated into the storyline. On the other hand, in business software, providing real-time explanations when users hover over the "Info" button may be a safer option than providing explanations every couple of steps.

After implementation, continuous monitoring and evaluation are necessary to gauge the impact of trigger-based explanations on the user experience. This includes monitoring users' progress toward their goals and observing how they interact with the software when errors occur and explanations are provided. Gathering feedback directly from users would also provide valuable insights into their experiences and allow for iterative improvements. Furthermore, expert reviews can bring additional perspectives and identify areas for improvement. Statistical analysis, such as comparing overall task completion durations before and after explanations are implemented, provides quantitative data to evaluate the effectiveness of trigger-based explanations. By applying these pre- and post-implementation strategies, it is possible to assess whether trigger-based explanations are improving the user experience and making a positive contribution to user satisfaction.

Q3 Are there any potential drawbacks that might affect the user experience negatively or raise concerns among users?

In segment four, participants determined the top three triggers for further exploration. These triggers were selected for further analysis in the small-group discussions in segment five: **repeated errors, unusual mouse movements, first-time use**. In segment five, participants explored the potential drawbacks associated with these top three triggers. Here are the responses to Q3 collected during the workshop:

- Repeated errors: May lead to frustration and a steep learning curve.
- Unusual mouse movements: May restrict user interaction, possibly causing irritation and a feeling of being monitored.
- First-time use: May be problematic due to the lack of pre-established expectations from both users and the software, potentially overwhelming users with excessive information.
- Overall: Timing issues may surface, potentially leading to intrusive experiences.

One of the triggers investigated was repeated errors, with participants highlighting concerns about user frustration and the emergence of a steep learning curve if the user does not understand it immediately. When users repeatedly encounter the same errors and receive explanations each time without being able to resolve the errors, it can create a cycle of frustration and confusion. This constant repetition can prolong the learning process as users struggle to understand and resolve the underlying issues, resulting in a steep and challenging learning curve.

Another trigger that came under closer review was unusual mouse movements. Participants expressed concern that this trigger could limit user interaction, leading to potential irritation and a sense of being monitored. If the unusual mouse behavior is misinterpreted as hesitation and incorrectly triggers an explanation, it could disrupt the flow of user interactions and create a sense of discomfort, reducing the usability and appeal of the software.

The third trigger, first-time use, also sparked discussion among participants. Concerns were raised about the challenges posed by the lack of pre-established expectations of both the user and the software. Participants noted that this lack of familiarity could overwhelm users with excessive information, complicating the onboarding process and hindering their ability to navigate the software effectively. Users who are unfamiliar with the software may be dealing with a high cognitive load as they attempt to process new information. In this context, lengthy explanations can add to the cognitive load, potentially overwhelming users and hindering their ability to absorb and retain the information provided. In addition, without prior user data or historical usage patterns to draw upon, the software may struggle to provide personalized information. For example, the explanations provided upon first use may contain content that is uninteresting or unimportant to some users. From the developer's point of view, however, this content may be important for users to know.

Overall, the small-group discussions highlighted the nuanced concerns and potential drawbacks associated with these triggers. Participants agreed on the importance of timing in delivering explanations and that trigger-based explanations should help to address timing issues. Moreover, to ensure a good user experience, information delivery should be contextually relevant and non-intrusive. If explanations are intrusive, they could distract users from their primary tasks, which could affect overall focus.

Q4 Are there difficulties or challenges (technical and non-technical), that might be encountered during the implementation process?

- Repeated errors: Establishing error thresholds and providing clear and concise explanations for errors and solutions.

- Unusual mouse movement: Understanding the reasons behind mouse behaviors, avoiding generalizations about user actions or task completion methods, and addressing memory and performance issues.

- First-time use: Tailoring information for new users, determining which explanations to prioritize (avoiding overwhelming users with excessive information).

- Overall: Addressing the complexities of UI design, ethical considerations (such as privacy agreements), and computational limitations of certain methodologies.

During the small-group discussions that focused on the challenges of implementing trigger-based explanations, participants identified several difficulties that could arise, both technical and non-technical. One of the challenges discussed was how to deal with repeated errors effectively. Establishing error thresholds and developing clear and concise explanations of errors and their solutions emerged as crucial steps in reducing user frustration and improving the learning experience. In addition, participants emphasized the importance of tailoring explanations to different error contexts to ensure that users receive relevant and actionable guidance.

Another challenge highlighted in the discussions was unusual mouse movements as a trigger for explanations. Participants recognized the complexity of understanding the underlying reasons for mouse actions. They emphasized the need to avoid making generalizations about user actions or task completion methods, as these movements can vary significantly based on individual preferences and interaction patterns. In addition, addressing the memory and performance issues associated with processing and analyzing mouse motion data emerged as a technical challenge that requires careful consideration during implementation.

In the context of unusual mouse movements, software systems need to track and analyze mouse movements to identify potential triggers and provide explanations. However, this process can strain system resources and impact performance, especially in cases where extensive data processing is required, as user interactions generate large amounts of data. Storing and retrieving detailed information about user interactions, such as coordinates, timestamps, and movement patterns, can consume significant memory resources, especially when dealing with large numbers of users or long sessions.

Participants also discussed the challenges associated with facilitating explanations during first-time use. Tailoring support to new users while avoiding overwhelming them with too much information seemed to be a tricky balance. Participants suggested prioritizing explanations based on their relevance and potential impact on user understanding, thereby facilitating the onboarding process and promoting a positive initial user experience. In addition, UI design complexities may also be another challenge. Developers must carefully consider factors such as placement and style of explanation elements to ensure that they are noticeable yet unobtrusive. Balancing visibility with minimal disruption to the user's primary tasks is crucial to prevent distraction and maintain a clean UI design.

Another example of challenges is ethical considerations. A primary ethical concern is the collection and use of user data to personalize trigger-based explanations. User data must be collected and used in compliance with relevant privacy regulations and ethical guidelines. Furthermore, in the case where the system detects the triggers autonomously, for example through machine learning algorithms, optimizing computational efficiency could be an additional challenge. This optimization may involve minimizing the computational complexity of algorithms, reducing memory usage, and optimizing data structures to ensure that trigger detection and explanation generation can be performed efficiently even on devices with limited processing power.

Q5 How can we tailor the triggers to accommodate diverse user preferences?

- Repeated errors: Adjust thresholds appropriately.
- Unusual mouse movements: Gather data or patterns, employ appropriate learning methods, focus on the target group, and ensure compliance with basic data privacy regulations.
- First-time use: Offer general assistance without excessive tailoring, allow users to explore freely initially, and provide an option to skip explanations.
- Overall: Consider user accessibility.

In addressing the question of tailoring triggers to different user preferences, participants explored several strategies presented above to ensure that trigger-based explanations are personalized and relevant to individual needs and preferences. Regarding repeated errors, participants pointed out the importance of defining error thresholds. By setting appropriate thresholds, trigger-based explanations can be triggered when users encounter errors at a level that indicates the need for assistance without overwhelming them with explanations for minor errors. For minor errors that have minimal impact on task completion, it may be preferable to provide subtle cues or notifications rather than interrupting the user with a pop-up explanation every time.

Another approach to dealing with unusual mouse movements was also discussed. Participants mentioned that collecting data and patterns associated with mouse movements is important to better understand how users interact with software. By using appropriate learning approaches, such as machine learning algorithms, insights from mouse motion data can be used to implement trigger-based explanations. Since mouse movements can indicate many different meanings, one may prioritize the target user group and draw insights primarily from the mouse interactions of that demographic. In addition, participants mentioned the need to ensure compliance with privacy regulations when collecting and analyzing mouse movement data.

When considering triggers for first-time use, participants discussed the challenge of providing assistance without overwhelming users who are new to the software. Participants suggested offering general assistance that is not overly tailored to individual user preferences initially, allowing users to explore the software freely and familiarize themselves with its features. Furthermore, participants recommended providing users with the option to skip explanations when they feel confident in their ability to navigate the software independently. This approach balances the need to provide support for new users with the importance of respecting user autonomy and preferences.

In general, participants emphasized the importance of considering user accessibility when designing triggers for trigger-based explanations. By ensuring that trigger-based explanations are accessible to users of all abilities and backgrounds, software developers can improve the user experience and promote inclusivity, for example by providing alternative formats such as audio or visual explanations for users with different abilities.

Key Takeaways and Concluding Remarks

After the discussions in segment six, the most prominent triggers identified by the group were repeated errors and first-time use. These triggers received the highest number of votes and were unanimously agreed upon as pivotal points for initiating the concept of trigger-based explanations. Compared to mouse movements, which can be difficult to interpret accurately and may not always indicate user intent or a need for assistance, both first-time use and repeated errors provide more concrete and actionable triggers for providing explanations.

Participants underlined the complexities involved in analyzing mouse movement and how it is significantly more challenging than other selected triggers. They stressed the inherent limitations of inferring user intent from mouse actions alone, citing the risk of misinterpretation and the potential for significant disruption to the user experience. In contrast, first-time use and repeated errors represent distinct user behaviors or interactions that signal a clear need for guidance or clarification, making them more reliable and effective triggers for providing explanations. Despite the challenges of detecting unusual mouse movements, participants remained optimistic about the potential of mouse movement as a trigger, highlighting the importance of further research and refinement to maximize its effectiveness in improving user experience.

Furthermore, participants also discussed challenges in general that are not unique to specific triggers. As mentioned in 2.4, previous studies have shown that users have different needs for explanations. In this case, participants suggested that focusing on the main target population of the software might be a good enough solution and the most effective way in most cases. Another challenge that was agreed upon was that determining triggers based on user actions is not straightforward. The group believed that we cannot generalize that trigger-based explanations are always better than providing explanations on demand. Instead of automatically displaying explanations for certain actions or events, users can access the Help menu or hover over certain items to trigger explanations as needed, providing flexibility to the user and avoiding unnecessary interruptions. Finally, the balance between simplicity and providing enough information is also not an easy task.

To ensure that trigger-based explanations have a positive impact on the user experience, several strategies can be implemented. First, allowing users to provide feedback after each explanation allows developers to gauge the effectiveness and relevance of the information provided. In addition, monitoring users' progress toward their goals after receiving explanations provides valuable insight into the effectiveness of the explanations in facilitating task completion. Finally, conducting user and public testing allows developers to observe how real users interact with the explanations in different scenarios, allowing them to refine and adjust explanations and triggers based on real-world usage patterns and feedback. Collectively, these activities help optimize trigger-based explanations to improve user experience and satisfaction and allow for iterative improvements.

In conclusion, participants agreed that meeting diverse user needs is a significant challenge due to their inherent variability. It was acknowledged that meeting these diverse needs depends not only on user preferences but also on the nature of the software itself. For example, whether it serves as a gaming platform, an integrated development environment (IDE), or a communication tool such as a chat platform greatly influences the approach to trigger-based explanations and whether they are appropriate. Repeated errors, as well as first-time use, are considered appropriate and important triggers for explanation, and the most technically feasible to investigate further. Finally, while it may not be easy to get satisfaction from all users, continuous user feedback can improve the functionality in the long run.

Chapter 4

Concept

This chapter introduces the concept of trigger-based explanations within this thesis. It includes the definition of the selected triggers, the respective roles of explanations, and the rationale behind their selection. Additionally, the chapter details the prototype implementation, outlining the different use cases for each trigger and how the concept was realized.

4.1 Triggers

Among the triggers identified during the workshop, four were selected for the conceptualization and implementation within this thesis: **first-time use, repeated errors, task interruptions, and repetitive actions**. The decision to focus on these particular triggers was made based on several considerations. First, their relevance to real-world scenarios was high. That means the results of the user study on these triggers are more likely to accurately reflect situations or events that commonly occur in real-world contexts. This ensures that the research results and findings are applicable, meaningful, and representative of real user experiences, thereby increasing the practical utility and impact of the study.

In addition, each selected trigger reflects common user interactions and behaviors found in various software applications, making them suitable for empirical study. Another critical factor in the selection process was feasibility. The selected triggers were deemed to be technically feasible to implement within the constraints of this thesis and well-fitted to the scope of the user study conducted during this work. While other triggers may offer interesting opportunities for exploration, their implementation may require more extensive resources or expertise. By taking these things into account and opting for technically feasible triggers, this thesis ensures a practical and manageable scope for experimentation and analysis.

4.1.1 First-Time Use

A summary of the first-time use concept in this study is shown in Table 4.1.1. First-time use refers to the initial interaction of a user with a software system or a specific feature within the system. For example, this could include interactions with uncommon features, newly introduced features following software updates, and tasks involving complex actions or specialized tools. In this context, the user is encountering or using a feature for the first time and may be unfamiliar with the interface, functionalities, or workflows of the software. Explanations in the case of first-time use play a role in introducing users to the software, providing guidance on navigating the interface, and offering explanations of key features or functionalities. They serve to guide users in learning to use certain features effectively, thereby also contributing to increasing overall effectiveness.

First-Time Use	
Definition	User interacts with a software system or a specific feature for the first time.
What it may indicate	User is unfamiliar with the interface, functionalities, or workflows of the software.
Case examples	Uncommon or beta features introduction, changes summary after software updates.
Roles of explanations	Introduce users to the software and features, enhance user understanding and confidence.

Table 4.1: An overview of the first-time use trigger concept in this thesis

Providing explanations during first-time use is not only essential for enhancing users' understanding of the software system but also for onboarding users smoothly, reducing frustration, and increasing users' confidence in using the software effectively. As an example of first-time use, after software updates involving alterations in shortcuts or menu placements, an explanation regarding the changes and how to utilize them can be provided by the first time a user interacts with the updated software. This assists users in understanding the changes made to the interface, thereby minimizing confusion, particularly among frequent users of the interface.

4.1.2 Repeated Errors

In the context of trigger-based explanations in this study, errors refer to instances where users deviate from the expected or desired behavior while interacting with a system or application. These deviations can manifest in various forms, such as incorrect inputs and failed actions. Incorrect inputs occur for instance when users provide inaccurate, invalid, or incomplete data. Failed actions arise when users attempt tasks within the system, but due to errors in execution or understanding, the desired outcome is not achieved.

It's essential to distinguish between errors that are significant enough to warrant trigger-based explanations and those that are considered minor or inconsequential. Significant errors are errors that impede users' progress, cause frustration, or lead to incorrect outcomes, negatively affecting the user experience. These errors typically require intervention or guidance to resolve effectively. Minor errors, such as typographical errors or minor formatting issues, may not significantly impact users' ability to complete tasks or achieve their goals within the system. In this thesis, this trigger focuses on addressing significant errors, as explaining minor errors could potentially cause more harm than good, resulting in disruption rather than assistance.

Table 4.2 provides a concise introduction to the concept of repeated errors that are investigated within this study. Repeated errors occur when a user encounters the same error or makes the same mistake multiple times consecutively while interacting with the system. This indication suggests a potential lack of clarity in the instructions or a learning barrier for the user. The role of explanations in this context is to provide corrective guidance, clarify confusing aspects of the interface or task, and reduce user frustration. By providing explanations when repeated errors occur, users gain insight into what went wrong, how to correct it, and how to avoid similar errors in the future, thereby improving user proficiency and satisfaction.

Repeated Errors	
Definition	User encounters the same error or makes the same mistake multiple times in a row.
What it may indicate	User is struggling with the same problem multiple times.
Case examples	Incorrect or invalid inputs.
Roles of explanations	Provide corrective guidance, suggestions or solutions.

Table 4.2: An overview of the repeated errors trigger concept in this thesis

4.1.3 Task Interruption

Another trigger that is explored in this thesis is task interruptions, focusing on interruptions by a user. Within Table 4.3, a summary of the concept of the third trigger in this study, task interruptions, is presented. Interruptions by users refer to instances where users manually interrupt or terminate a process or task before its completion within the software system. This action indicates a break in the workflow or sequence of operations, potentially leading to incomplete tasks or data loss. Such interruptions might indicate that users do not understand how to complete certain tasks, need assistance in completing them, or are unsure about providing certain information.

Examples of task interruptions during user interaction include instances where a user prematurely terminates a process or session by clicking the "Cancel" or "Close" button or exits the application prematurely. Another scenario of task interruption is when a user selects "Remove All" or "Delete All" options to eliminate data or selections, interrupting ongoing processes. Lastly, another example occurs when a user shifts focus to another task or application, navigates away from the page or window, or closes a pop-up without providing the necessary input to proceed.

In this context, explanations can offer help to various use cases. Firstly, they inform users about the consequences of interrupting the ongoing task, if there is any, such as unwanted data loss. Secondly, they provide guidance on how to resume or undo the interrupted action and offer alternatives if necessary. In conclusion, providing explanations for task interruptions caused by user actions helps users understand the implications of their actions, recover from unintended interruptions, and avoid potential data loss or errors.

Task Interruption	
Definition	User terminates a process before its completion.
What it may indicate	User needs assistance in completing tasks or is unsure about providing certain information.
Case examples	User clicks "Cancel" or "Close" button, user selects "Remove All" option.
Roles of explanations	Inform users about the consequences of interrupting the ongoing task and provide guidance on how to resume or undo the interrupted action.

Table 4.3: An overview of the task interruption trigger concept in this thesis

4.1.4 Repetitive Action

This section presents the last trigger in this thesis, which is repetitive actions. The summary shown in Table 4.4 offers an introduction to the concept of repetitive actions explored in this study. As a trigger for explanations, repetitive actions involve instances where users engage in the same interactions multiple times within a short period. Such actions may indicate potential inefficiencies or frustrations within the current workflow or interface design, prompting users to repeat actions to achieve their desired outcomes. This repetition may signify that users are encountering difficulties or inefficiencies in completing tasks smoothly and may not be aware of more efficient methods.

Repetitive Action	
Definition	User performs the same interactions multiple times within a short period.
What it may indicate	There is potential confusion or inefficiencies or frustrations with the current workflow or interface design.
Case examples	Repeated form submission, frequent clicks on the same element.
Roles of explanations	Optimize user workflows and clarify confusion.

Table 4.4: An overview of the repetitive action trigger concept in this thesis

Explanations tailored to address repetitive actions aim to identify opportunities to streamline workflows, offer shortcuts or automation features, and educate users on more efficient ways to complete tasks. By providing such explanations, users can optimize their workflows, save time, resolve confusion, and enhance satisfaction by empowering them with more efficient methods to finish their tasks.

Examples of repetitive actions that might warrant trigger-based explanations include scenarios such as repeated form submissions, where users submit the same form or data multiple times within a short period of time, possibly indicating uncertainty about the success of the action. Additionally, frequent clicking on the same element, such as a button or link may suggest user confusion or signal difficulty understanding the navigation flow or producing the desired output. Similarly, repetitive search queries, where users repeatedly search for the same terms without satisfactory results, could indicate a need for clarification or guidance on refining their search.

4.2 Use Case

This section explains the use cases implemented within the user study prototype in the context of online shopping. Each use case is related to an online shopping scenario and corresponds to a trigger identified in Section 4.1, illustrating the practical application of trigger-based explanations in real-world contexts. It's important to note that while each trigger is demonstrated in a specific use case, these use cases are not the only scenarios where the triggers could be applicable. They only serve as examples of the possible scenarios in which the trigger could be used in practice to provide explanations.

4.2.1 Save a product to a collaborative collection

A scenario illustrating the **first-time use** trigger involves a user's initial encounter with a specific feature within the system. In this study, the feature is a shared or collaborative collection. Through collaborative collections, two users can add products to the same collection and see the products saved by either of them. When a user explores the website and discovers a specific product they wish to save for later, they can add it to a collection for easy access. This use case is defined for first-time use because, to the best of our knowledge, this feature is not a common feature within online shopping websites. Hence, it aligns well with this trigger, as discussed earlier in the previous section.

After the user reaches the collection page and enters a name for the newly created collection, an explanation pop-up is triggered. This pop-up introduces the concept of a collaborative collection, where users can collaborate with another user by inviting them to join the collection. This triggered explanation serves to familiarize users with the shared collections feature, facilitating a smoother onboarding experience and ensuring users fully leverage the platform's capabilities.

4.2.2 Create an account with a username

The use case illustrating the trigger of **repeated errors** involves the process of creating an account on an online shopping website. During this process, users are prompted to complete a form, including the selection of a username. In this scenario, if the user attempts to create a username that does not meet the username requirements for at least two consecutive attempts, the trigger for repeated errors is activated. Providing explanations after at least two failed attempts ensures that significant errors are addressed and avoids unnecessary interruptions and distractions. This approach avoids the inconvenience of receiving explanations for minor errors and allows users more time to become familiar with the system before requiring assistance. This use case simulates a scenario where a user does not know how to do certain operations, however, the system expects the user to be familiar with it, for example because it is a common operation in many software systems. Consequently, the system refrains from offering an explanation immediately. Upon detection of an invalid username, a dialog notifies the user of the discrepancy and advises them to select an alternative username. Additionally, a pop-up window appears to outline the necessary criteria for a valid username, assisting users in understanding the requisite parameters for successful submission.

4.2.3 Change the number of items in the shopping cart

A use case simulating the **task interruption** trigger involves the scenario of adjusting the number of items in a shopping cart on an online shopping platform. In this situation, on the shopping cart page, the trigger is activated when a user interrupts the shopping process by clicking the "Remove All" button instead of adjusting the number of items in the cart. Upon detecting this task interruption, the system responds by displaying a confirmation dialog to ensure the user's intent in emptying the shopping cart. This dialog effectively communicates the potential consequences of removing all items, giving users the information they need to make an informed decision. At the same time, an explanatory pop-up offers guidance on how to change the number of items in the shopping cart in order to continue the shopping process smoothly. This use case simulates a scenario where the system expects the user to be familiar with certain common operations, in this case how to edit the number of items. However, when the user tries to interrupt the task, the system assumes that the user's action to empty the shopping cart was unintentional and recognizes that the user may not know how to change the number of items. The explanatory pop-up is designed to help aims to help them understand how to seamlessly resume the buying or ordering process. This proactive approach of providing explanations through pop-ups serves to inform users of alternative actions and guide them to effectively complete their potentially intended tasks.

4.2.4 Search for a specific product

A scenario that demonstrates the trigger of **repetitive actions** involves browsing for a specific item on an online shopping site, where users navigate through multiple product pages in sequence to find the desired item. If the user repeatedly clicks the "Next Page" button to navigate through the products instead of using the search function available on the home page, this action will be detected as a repetitive action. Upon detecting this repetitive action, the system then displays an explanation pop-up. This pop-up informs the user of the existence of a search bar feature and encourages them to use it for faster and more efficient product searches. By providing this explanation through a pop-up, the system aims to guide the user to a more effective approach to finding the desired product, thereby improving the user experience and streamlining the search process.

4.3 Prototype Implementation

In the process of transforming the ideas into a more concrete realization, the implementation of a prototype is the next step. This section marks the beginning of this transformation process, where the concepts laid out in the previous chapters are realized into working prototypes. With a clear focus on facilitating the upcoming user study, this chapter delves into the framework and resources used in the implementation phase. It outlines the detailed process of implementing each defined use case mentioned in the previous section, providing insight into the techniques used and the challenges encountered along the way. In addition, this section provides a peek into the prototypes through selected snippets and screenshots, offering a visual representation of the conceptualized ideas.

The use cases are implemented in a scenario of an online sticker shopping experience through a website. The website prototype was implemented using the framework of Axure RP 10, a tool for creating realistic and functional prototypes. In addition, the product images featured in the online store, which are stickers, are created using an advanced AI technology called DALL-E 3 via ChatGPT. DALL-E 3 enables the transformation of text-based concepts and ideas into automatically generated visual representations. As users navigate through the sticker shop prototype, they encounter a diverse and vibrant collection of AI-generated visuals that enhance their browsing experience.

In the prototype, all trigger-based explanations are designed in pop-up formats. By incorporating pop-ups, the prototype can strategically draw users' attention to critical information or features within the interface. Unlike static content embedded within the page, pop-ups have a dynamic and interactive nature that inherently commands attention. This dynamic presentation style can be particularly beneficial when highlighting important information, ensuring that users are immediately informed and engaged. This is also deliberately implemented so that participants in the user study will notice and hopefully not ignore the explanations so that the impact of the explanations can be evaluated.

The primary difficulty faced during the implementation is configuring scenarios that provoke the desired user responses. Aside from the first-time use trigger, the activation of the other triggers is not guaranteed, as they depend on specific instances of user confusion or error. Therefore, when developing the website prototype, usability and ease of use must be compromised somewhat in order to direct participants into situations that will activate the triggers and present explanations. This is not an easy way to design a software because this approach differs from conventional software design practices, which prioritize improving usability to streamline the user experience. However, without deliberately incorporating these schemes, the likelihood of users activating triggers and receiving explanations is significantly low, making it difficult to assess the impact of these explanations on the user experience during the user study. Therefore, certain features are intentionally designed to be less intuitive, thereby simulating situations where an explanation is needed.

Figure 4.1 depicts the homepage of the website prototype in full size,

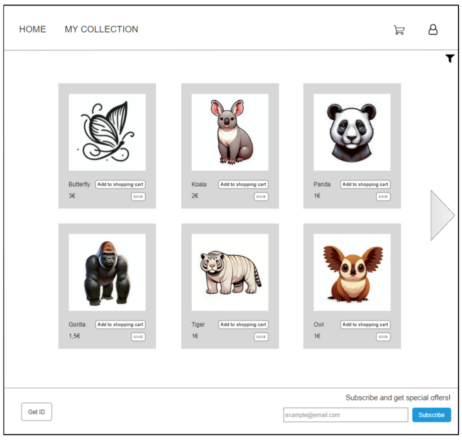


Figure 4.1: Homepage of the website prototype featuring navigation options and product page

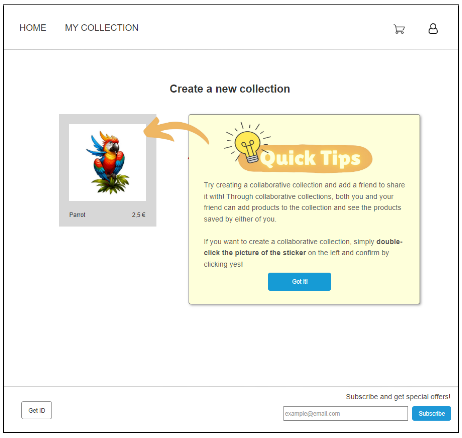
Name	**Save a sticker to a collaborative collection**
Actor	User
Description	A user saves a sticker to a collaborative collection.
Precondition	The user finds a product they want to save.
Trigger	The user saves a sticker.
Success scenario	1. The user saves a sticker. 2. The system directs the user to a page where they input the collection name and they may add another user to share the collection with. 3. The user inputs the name of the collection. 4. The system provides information about the "Collaborative Collection" feature in a pop-up dialog. 5. The user inputs the username of another user and saves the collection. 6. The system displays a message, indicating successful creation of the collection.
Extensions	
Exceptions	5a. IF the collection name is empty, THEN the system displays a message prompting user to fill in the collection name.

Table 4.5: Use case demonstrating the first-time use trigger in a scenario of saving a sticker in a collaborative collection

showcasing navigation options to other pages like collections, shopping carts, and profiles. In the footer, the "Get ID" button serves solely to assign participant IDs during the user study and is not relevant to the use case scenarios. The following parts of this section detail the implementation of each of the use cases outlined in Section 4.2.

First-time Use - Save a product to a collaborative collection

Table 4.5 outlines the scenario of the first-time use trigger within this project. Moreover, Figure 4.2 illustrates the collection page featuring a pop-up explanation introducing the collaborative collection feature. This explanation is triggered when the user has never received any information about the feature and the mouse loses focus on the collection name field.



Try creating a collaborative collection and add a friend to share it with! Through collaborative collections, both you and your friend can add products to the collection and see the products saved by either of you.

If you want to create a collaborative collection, simply **double-click the picture of the sticker** on the left and confirm by clicking yes!

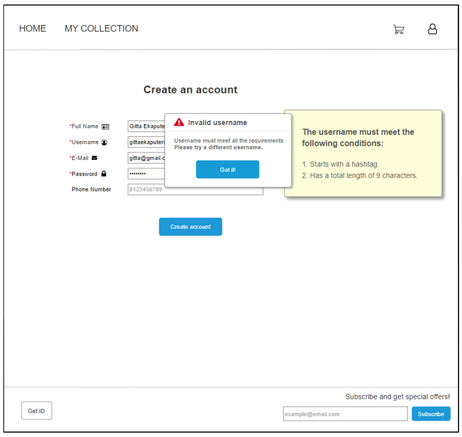
Figure 4.2: Collection page from the website prototype, featuring a pop-up introducing the collaborative collection feature

Repeated Errors - Create an account

Name	**Create an account**
Actor	User
Description	A user creates an account.
Precondition	The user is on the website without an existing account.
Trigger	The user navigates to the profile page.
Success scenario	1. The user navigates to the profile page. 2. The system displays a registration form prompting the user to input their profile information, including full name, username, password, email address, and phone number. 3. The user enters the required information and submits the form. 4. The system checks if all required inputs meet the requirements. 5. The system displays a message, indicating successful creation of the account.
Extensions	
Exceptions	4a. IF any of the inputs are empty, THEN the system displays a message prompting user to fill in all required information. 4b. IF the given username does not meet the username requirements, THEN the system displays a message informing that the username is invalid and it must meet all the requirements. 4c. IF user submits form with an invalid username for at least two times, THEN the system shows the list of username requirements in a pop-up dialog.

Table 4.6: Use case demonstrating the repeated errors trigger in a scenario of creating an account

The use case for the second trigger, repeated errors, in the context of creating a user account with a username, is described in Table 4.6. In addition, Figure 4.3 shows the profile page with an invalid username error message, along with a pop-up that lists the requirements of a username. This pop-up is displayed after at least two consecutive attempts to enter a username that does not meet the requirements. Notably, instead of revealing the requirements before the user enters the username, the requirements are revealed only after two consecutive occurrences of an invalid username. While this approach differs from common design practices, it is considered necessary in this study to deliberately induce user errors. Otherwise, there may be a lack of errors during the user study and therefore not enough instances of users receiving the pop-up to evaluate. Moreover, this use case also simulates the possible scenario mentioned in Section 4.2.2, where the system expects the user to be familiar with the conditions, and only provide an explanation after multiple errors.



Create an account

Invalid username

Username must meet all the requirements. Please try a different username.

The username must meet the following conditions:

1. Starts with a hashtag.
2. Has a total length of 9 characters.

Subscribe and get special offers!

Figure 4.3: Profile page of the website prototype displaying an error message for an invalid username and the corresponding list of username requirements

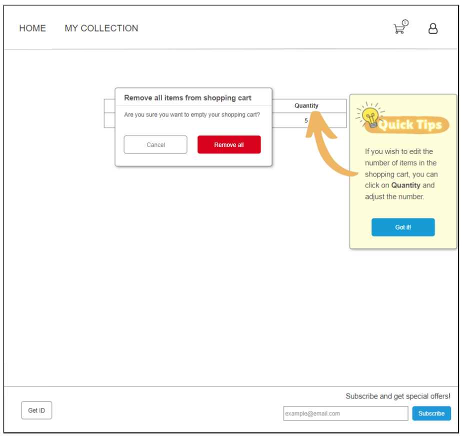
Task Interruption - Change the number of items in the shopping cart

Name	Change the number of items in the shopping cart
Actor	User
Description	A user changes the number of items in the shopping cart.
Precondition	The shopping cart is not empty.
Trigger	The user opens the shopping cart and attempts to modify the quantity of items.
Success scenario	1. The user opens the shopping cart. 2. The system presents a list of items in the user's shopping cart. 3. The user modifies the number of items in the shopping cart. 4. The system updates the shopping cart to reflect the changes made by the user.
Extensions	
Exceptions	2a. IF the user removes all items from the shopping cart, THEN the system prompts a confirmation dialog to ensure the user's intention AND displays a pop-up message informing the user about how to edit the number of items in the shopping cart.

Table 4.7: Use case demonstrating the task interruption trigger in a scenario of changing the number of items in the shopping cart

The scenario involving the third trigger, task interruption, is executed in the context of adjusting the quantity of items in the shopping cart, as shown in Table 4.7. In addition, Figure 4.4 shows the shopping cart interface with an explanatory pop-up that details the process of modifying the item quantity. This explanation is prompted when the user clicks the "Remove All" button, which indicates a task interruption as described in Section 4.1.3.

The method for adjusting item quantities was intentionally developed to be non-intuitive in order to entice users to empty the cart and redo the process of adding stickers to their shopping cart. Moreover, this scenario also simulates a possible situation mentioned in Section 4.2.3, where the system assumes that most users know how to edit the number of items.



Remove all items from shopping cart

Are you sure you want to empty your shopping cart?

If you wish to edit the number of items in the shopping cart, you can click on **Quantity** and adjust the number.

Subscribe and get special offers!

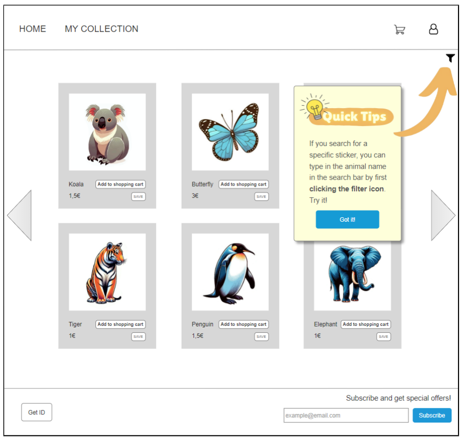
Figure 4.4: Shopping cart view in the website prototype, displaying a confirmation dialog and informational message about how to change item quantities

Repetitive Action - Search for a specific product

Name	**Search a specific sticker**
Actor	User
Description	A user searches for a specific sticker.
Precondition	The user is on the website homepage.
Trigger	The user browses through the product pages.
Success scenario	1. The user browses through the product pages. 2. The system displays pages of stickers. 3. The user finds the searched sticker.
Extensions	1a. IF the user enters a keyword into the search bar, THEN the system shows stickers containing the provided keyword in their names. 1b. IF the user navigates to the next page at least four times, THEN the system displays a pop-up message informing about the search. feature.
Exceptions	

Table 4.8: Use case demonstrating the repetitive action trigger in a scenario of searching for a specific product

The final use case is outlined in Table 4.8 and covers repetitive actions related to searching for a specific sticker. A pop-up introducing the search bar functionality is triggered when the user navigates to the next page four times, as this behavior may indicate a lack of awareness of the search feature. Figure 4.5 illustrates the presentation of this explanation, with an arrow directing attention to a filter icon. This icon is deliberately chosen over a search icon with the intention that users will overlook the presence of the search feature before the pop-up is activated. This is intended to cause the user to navigate through the pages in search of a sticker and then trigger the pop-up. Consequently, this setup allows us to evaluate the impact of trigger-based explanations on user experience and task completion. Furthermore, depicted in Figure 4.6 is an instance where the search bar is utilized, demonstrating that only stickers containing the keyword in their name will be displayed.



If you search for a specific sticker, you can type in the animal name in the search bar by first **clicking the filter icon**. Try it!

Figure 4.5: Triggered explanation appears following the detection of repetitive actions, providing information about the search feature

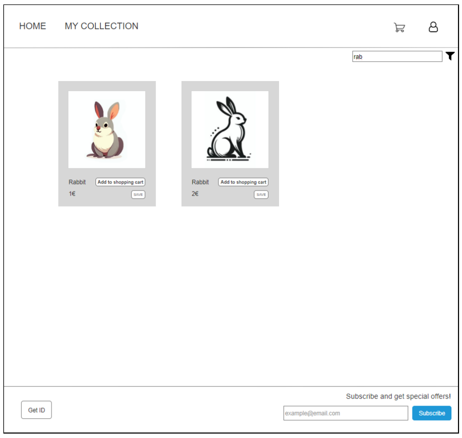


Figure 4.6: Utilization of the search bar, showcasing stickers filtered by keywords in their names

Chapter 5

User Study

This chapter describes the user study conducted as part of this thesis, which focuses on the evaluation of trigger-based explanations within the prototype presented in the previous chapter. Along with the execution of the study, this chapter also presents the preparation phase involving the target participants and the design of the tasks and questions.

The primary goal of the study is to evaluate the concept and implementation of trigger-based explanations through user testing, with a strong focus on exploring their effectiveness, efficiency, suitability, and user satisfaction. By investigating these aspects, we aim to gain a deeper understanding of how users interact with and perceive trigger-based explanations within the prototype and the impact of the explanations on the user experience. The results of the study are expected to provide answers to the defined goal-question-metrics and ultimately address the research questions posed in this thesis.

5.1 Preparation

The preparation for the user study included planning and ensuring effective data collection. The study was divided into two parts: user interaction with the prototype and feedback collection through a questionnaire. In the first part, participants were presented with a series of tasks designed to evaluate the implemented prototype through user interaction. These tasks were carefully designed to cover different use cases of the prototype that allow the evaluation of trigger-based explanations. For the second part, participants were asked to complete a feedback survey via a questionnaire. This questionnaire was designed to gain insight into the participants' experiences, perceptions, and preferences regarding their interaction with the prototype. Overall, both parts of the user study were carefully planned in order to maximize the quality and reliability of the data collected. This is discussed in more detail in the following subsections.

5.1.1 Target Participants

Selecting individuals with prior experience in online shopping was aimed to enrich the study with insightful feedback. Their familiarity with the standard features and functionalities of such platforms may offer informative opinions on the incorporation of trigger-based explanations. This familiarity mitigates the potential effects that unfamiliarity might cause. Moreover, including everyday software users with diverse technical proficiencies is another strategy to ensure the study's findings are comprehensive and reflective of a wide user base, as these individuals may exhibit unique behaviors, preferences, and expectations in their interactions with software. These approaches aim to gather a comprehensive array of feedback on the integration of trigger-based explanations into the prototype and overall user experience.

5.1.2 Task Design

For the first session of the user study, participants will engage with the prototype of an online shopping website to carry out a series of tasks designed to evaluate various aspects of the user experience. These tasks are divided into two categories: four core tasks representing specific use cases with triggers for providing explanations, as outlined in Section 4.2 and Section 4.3, and four additional tasks aimed at enhancing overall user engagement and familiarity with the website interface. All of these tasks are not ordered based on their category.

The core tasks, each representing a distinct scenario, are designed to activate trigger-based explanations within the prototype. Participants will be asked to complete actions such as searching for a specific item, modifying quantities in the shopping cart, or creating a username, triggering explanations when specific conditions are met. These tasks serve to assess the effectiveness, suitability, and user satisfaction with the implemented triggers, providing valuable insights into their impact on the user experience.

In addition to the core tasks, participants will also engage with four supplementary tasks designed to offer more exploration of the website functionalities. These tasks serve multiple purposes, including boosting participant confidence by providing additional opportunities for successful interactions, mitigating potential frustration that may arise from encountering difficulties in completing the core tasks and familiarizing users with the website interface and navigation. By including these extra tasks, we aim to create a more positive and engaging user experience while gathering valuable insights into various aspects of the explanations provided during the interaction.

Written instructions and guidelines are provided to participants to streamline the user study process, as well as ensure consistency and minimize potential biases. The complete task instructions can be found in Appendix A. The expected duration for completing all tasks is approximately 10 minutes, with each task formulated to be clear, concise, and representative of real-world scenarios. Through these tasks, we aim to gather data on user behaviors and reactions to trigger-based explanations within the prototype.

5.1.3 Questionnaire Design

To gather feedback from participants, a Google Form was utilized, comprising four sections, each representing a core task. Afterwards, demographic questions were presented to capture essential participant information such as age, gender, technical proficiency, and frequency of visiting online shopping websites. The questionnaire was designed with a particular focus on obtaining feedback on the trigger-based explanations integrated into the prototype.

The questions are designed based on the metrics defined in the goal-question-metrics in Section 3.2. The questionnaire encompassed various question types, including Likert scale, multiple-choice, and open-ended questions. Each core task was systematically evaluated, starting with a question into whether participants anticipated receiving guidance during task completion. If affirmative, participants were prompted to rate the extent to which the provided explanation met their expectations.

Subsequently, participants were asked whether they actually received an explanation for the task. Only if an explanation was received did the questionnaire provide further questions regarding the participant's perception of the explanation. Questions ranged from assessing helpfulness, satisfaction, and timing of the explanation to its overall impact on the user experience, all rated on a Likert scale ranging from 1 to 5.

Moreover, participants were also given the opportunity to provide elaborative insights through open-ended questions. They were encouraged to articulate their opinions regarding the timing of the explanations and whether additional explanations were perceived as necessary during task completion. These optional, open-ended questions were designed to capture nuanced feedback and identify areas for improvement in the implementation of trigger-based explanations. The complete questions are listed in Appendix B.

5.2 Execution

Following the planning and preparation phase of the user study, this section outlines the execution of the study. It provides insights into the demographics of the participants, giving an overview of the user base involved in the study. Furthermore, it also details how the study was conducted and covers the settings in which it took place.

5.2.1 Demographics

The user study involved 30 participants, primarily divided into two age groups: 18-24 and 25-34, with the latter representing the majority with 21 participants. The gender distribution within the study showed a predominance of male participants, with 21 (70%) of the total, while female participants made up 8 (27%), and one individual preferred not to disclose their gender. Moreover, participants were also asked to indicate their level of technical proficiency. Table 5.1 shows the different levels and definitions offered as response options in the questionnaire. The result was a diverse group in terms of technological proficiency. 11 participants (37%) identified themselves as intermediate users, capable of performing most tasks independently, while an equal number identified as advanced users, capable of handling tricky tasks. Expert users, with extensive experience in advanced tasks, made up 4 (13%) of the participants, adding depth to the study with their high level of expertise. The study also included a smaller contingent of beginner (3, 10%) and novice (1, 3%) users, providing insights into the impact of trigger-based explanations across varying levels of technical familiarity.

Proficiency	**Description**
Novice	You are new to using technology and software and you find simple tasks challenging.
Beginner	You are starting to get the hang of using technology and software and can do basic tasks with some help.
Intermediate	You are pretty comfortable using technology and software and can do most tasks on your own.
Advanced	You are really good with technology and software and can easily handle tricky tasks.
Expert	You are a pro at using technology and software and have lots of experience with advanced tasks.

Table 5.1: Levels of technical proficiency and definitions, provided as answer options in the questionnaire.

5.2.2 Prototype Interaction and Feedback Survey

Prior to the start of the study, participants were reminded that the focus of the study was on their overall interaction with the prototype and their opinions about it, rather than on individual task performance. They were also informed of the expected duration of the study, including the completion of the questionnaire, which was expected to take approximately 25 minutes in total. Explicit consent was also obtained from all participants in advance for the collection of screen recordings of their interactions with the prototype, their experiences, feedback, and demographic details.

Participants were given the option to participate in the study either online or in person. In the online setting, a communication platform called Big Blue Button was used, and participants were offered the flexibility of either recording their own screen or sharing their screen for me to record. In addition, the in-person sessions were conducted in quiet environments and participants were given the choice of using their own laptop or my laptop. In both settings, participants engaged with the prototype without being observed by a third party.

To streamline the process and focus the recording on the interaction with the prototype, participants were advised to open the task instructions on a second monitor or device. This setup ensured that the main monitor only captured their engagement with the prototype, facilitating a smoother execution of the tasks. Upon completion all tasks, participants were directed to fill out a questionnaire to provide feedback on their experience and opinions of the prototype.

Chapter 6

Evaluation

6.1 Data Analysis Procedure

Data analysis was conducted to examine the responses collected through both quantitative and qualitative methods. For each trigger, participants' responses were evaluated to address the metrics defined in Section 3.2. The data collected included a mix of closed and open-ended questions as explained in Section 5.1.3, such as Likert scales ranging from 1 to 5, with 1 being the lowest (no impact at all or very negative) and 5 being the highest. Moreover, there were also nominal data, for example whether the participants anticipated an explanation or not, and rational data, which was the task completion duration. Furthermore, there were also optional open-ended questions that allowed participants to express their feedback in more detail.

Although Likert scales are typically treated as ordinal scales, in this thesis, they will be treated as interval scales in order to facilitate more flexible quantitative analysis, for example to enable the calculation of means and standard deviations. This approach to treat Likert scale as ordinal scales is also supported by Wu & Leung [24], with the assumption that the difference between each point on the scale is equal. The analysis began by looking at the distribution of ratings across the five aspects: expectation fulfillment, helpfulness, timeliness, satisfaction, and overall impact on user experience. Expectation fulfillment ratings were only collected from participants who confirmed that they anticipated receiving guidance. Moreover, the rest of the aspects, which are helpfulness, timeliness, satisfaction, and overall impact on user experience, were only rated by those who actually received the explanation.

Descriptive statistics such as mean, median, and mode were then computed to summarize the overall tendency of the responses. Standard deviation and frequency counts were also calculated to analyze the distribution of responses. These calculations were also visually presented through some graphs to provide better readability. At the end of the quantitative analysis, the proportion of participants who stated that they understood the given explanation among those who received an explanation is presented. Although the clarity of the explanation is not the main focus of this thesis, it can also somehow affect the evaluated aspects to some extent. Therefore, insights into their feedback regarding understanding are provided to give a better idea of whether a clarity issue arose during their experience in the user study. A number of other calculations were also carried out with regard to other metrics.

To provide more depth and context to the quantitative findings, the data were also supplemented with qualitative findings. The qualitative analysis focused on answers to open-ended questions and was conducted through the processes of open, axial, and selective coding of the grounded theory methodology. This process began with open coding, in which the data were extracted from the questionnaire responses and labeled with codes. Axial coding was then executed to categorize these data by identifying relations between the codes. Afterwards, selective coding was applied to identify a core category based on the existing categories. This qualitative approach not only enriched the understanding of the quantitative results, but also provided detailed and more genuine insights into the participants' perspectives.

6.2 Quantitative Data Analysis

6.2.1 First-time Use - Save a product to a collaborative collection

Figure 6.1 displays a frequency distribution of participant ratings of five different aspects of the given explanation triggered by first-time use corresponding to metrics **M3, M6, M8, M12, and M17**. In total, there were actually 29 participants who received the explanations. However, four of them stated during the survey that they did not receive or were not sure if they received an explanation during the user study. Therefore, this analysis consists of an evaluation of 25 feedback responses related to the first-time use trigger. One participant did not receive an explanation because the task was aborted before they found the sticker they were looking for.

In Figure 6.1 we can recognize that in all aspects the highest score (5) was given by most of the participants. This indicates that the explanation provided by the first-time use was found to be highly positive by a significant number of participants. However, it is also important to note that some participants also gave a rating between the lower ranges (1 or 2), suggesting that for a few of them, the explanation made a very minimal or no positive contribution to these aspects.

Table 6.1 presents descriptive statistics of the responses collected during the user study. The **expectation fulfillment** aspect evaluates how well

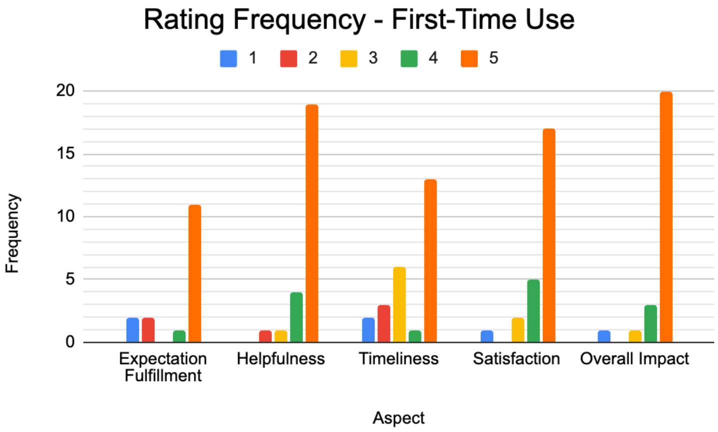


Figure 6.1: Rating distribution of the first-time use trigger

users felt their expectations were met by the explanations. The mean rating for this aspect was 4.06 with a median and mode of 5, indicating that most users felt that the given explanation aligned with their expectations. However, a standard deviation of 1.57 in this data indicates that the ratings are widely dispersed from the mean, suggesting a considerable amount of inconsistency in the ratings among all participants.

Aspect	Mean	Median	Mode	SD
Expectation Fulfillment	4.06	5	5	1.57
Helpfulness	4.64	5	5	0.76
Timeliness	3.80	5	5	1.41
Satisfaction	4.48	5	5	0.96
Overall Impact	4.64	5	5	0.91

Table 6.1: Descriptive statistics summary of the first-time use trigger

Helpfulness, with a standard deviation of 0.76, suggests a generally consistent opinion between the respondents. With a median and mode of 5 and a mean of 4.64, which is very close to the highest score, it can be inferred that users generally agreed that the explanation given was very helpful.

Similarly, the **timeliness** aspect also scored a mode and median of 5. However, the mean of this aspect was 3.80, which is the lowest mean of all aspects for this explanation. This aspect also had a standard deviation very similar to that of **expectation fulfillment**, which was 1.41.

Moving on to **satisfaction** and **overall impact**, both aspects showed a median and mode of 5, with mean ratings of 4.48 and 4.64, respectively.

These two aspects also resulted in similar standard deviations of around 0.9, indicating a high degree of agreement among the participants that the explanation has a very positive impact on their satisfaction and overall user experience.

Figure 6.2 illustrates how many participants understood and how many did not understand the given trigger-based explanation among 25 participants who stated receiving it. 96.0% reported that they understood the explanation, while 4.0% reported they did not. Since a very high proportion of them understood the explanation, it is very likely that most ratings for the five previously mentioned aspects were not influenced by comprehension issues.

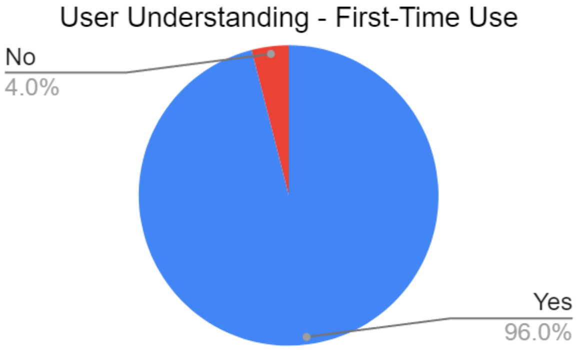


Figure 6.2: Percentage of participants understanding and not understanding the given explanation

Table 6.2 provides a summary of the analysis results for the rest of the defined metrics. M15 is not relevant to this trigger because there are no follow-up tasks associated with it. Moreover, M16 is also irrelevant because the prototype implementation was such that the participants would have to rely on the explanation to successfully complete the task. The table shows that all participants were able to complete the task upon reading the explanation. In addition, despite a significant number of trigger activations by participants who did not anticipate guidance (M5), their **helpfulness** ratings were remarkably high, at 4.5 on average (M9).

6.2.2 Repeated Errors - Create an account

Figure 6.3 shows a frequency distribution of user ratings across five different aspects of the given explanation triggered by repeated errors. 29 participants in total received the explanations. However, one of them stated that they did not receive one, resulting in 28 feedback responses related to this trigger. One participant did not receive the explanation because they did not start the related task.

Metric	Result
M1: Number of instances where users anticipated an explanation but did not receive one.	0
M4: Percentage of total trigger activations among users who anticipated guidance.	100%
M5: Percentage of total trigger activations among users who did not anticipate guidance.	92.86% (13/14)
M9: User ratings on the helpfulness among users who did not anticipate guidance. (Mean)	4.5
M10: Analysis of reasons why users did not read the given explanation.	Every user who received an explanation read it.
M11: Proportion of users who understood the explanation.	96%
M13: Ratio between successful and unsuccessful task completions after users read the explanation.	25 : 0
M14: Correlation between reading explanation and successful task completion. (chi-square test of independence, $\alpha = 0.05$)	Insufficient evidence of a statistically significant relationship (p-value=0.00004).
M15: Proportion of users who apply the given guidance in the subsequent tasks.	Irrelevant for this trigger.
M16: Analysis of total time spent to complete tasks between users who read and did not read explanations.	Irrelevant for this trigger.

Table 6.2: Results of the further analysis in regards to the defined metrics for first-time use trigger

A noticeable trend from the graph is that the aspect of **helpfulness** has the highest frequency of the highest ratings, indicating that a significant number of users found the trigger-based explanations to be very helpful. In contrast, timeliness received the lowest frequency of top ratings, and a relatively high number of participants gave it low ratings (1 and 2). This indicates that most participants found the timing of the explanations to be inappropriate.

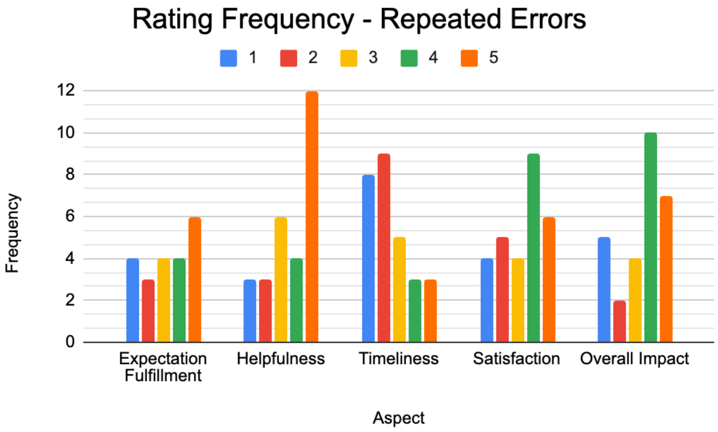


Figure 6.3: Rating distribution of the repeated error trigger

Table 6.3 presents a data summary of participants' responses, showing the mean, median, mode, and standard deviation. In terms of **expectation**

Aspect	Mean	Median	Mode	SD
Expectation Fulfillment	3.24	3	5	1.51
Helpfulness	3.68	4	5	1.42
Timeliness	2.43	2	2	1.32
Satisfaction	3.29	4	4	1.38
Overall Impact	3.43	4	4	1.43

Table 6.3: Descriptive statistics summary of the repeated error trigger

fulfillment, the mean rating for this aspect was 3.24 with a median of 3. A mode of 5 indicates that the distribution of ratings is skewed toward higher values. Although most participants responded that the explanations met their expectations very well, Figure 6.3 shows that there are still users at the lower end of the scale, suggesting that some of them may have had different expectations or experiences that were not very well addressed. This is also supported by the standard deviation for this aspect, which was 1.51, reflecting a high variability in responses.

Regarding **helpfulness**, participants rated this aspect higher with a mean of 3.68, a median of 4, and a mode of 5, suggesting that on average, users found the explanations to be quite helpful. Moreover, the standard deviation for helpfulness at 1.42 was lower than for expectation fulfillment, indicating closer data points to the mean.

Timeliness received the lowest mean rating of 2.43, paired with a median and mode of 2, indicating that the timing of the explanations generally did not meet participants' expectations. With a standard deviation of 1.32, there is little variation in opinions on this aspect.

Satisfaction showed a mean of 3.29 with a median and mode of 4, indicating a tendency toward slightly more positive responses. Nevertheless, there are varied responses in satisfaction among the participants, as shown by a standard deviation for satisfaction of 1.38.

Finally, the **overall impact** aspect had a mean rating of 3.43. Both the median and mode were 4, indicating that participants generally perceived the explanations to have a positive impact. Furthermore, the standard deviation was 1.43, very similar to the distribution of the helpfulness ratings. The numbers suggest that despite some timing issues, the explanations made a positive contribution to the user experience for most participants.

Figure 6.4 depicts the distribution of user responses to their understanding of the trigger-based explanations. Among the participants who received the explanations, 82.1% indicated that they understood the explanation, 7.1% indicated that they did not understand, and 10.7% were not sure. The majority's understanding suggests that for most of the participants, the ratings in the aspects of expectation fulfillment, helpfulness, timeliness, satisfaction, and overall impact were most likely not negatively affected by a comprehension problem. On the other hand, the comprehension difficulty reported by 17.8% of participants may negatively affect their evaluation of the explanation provided.

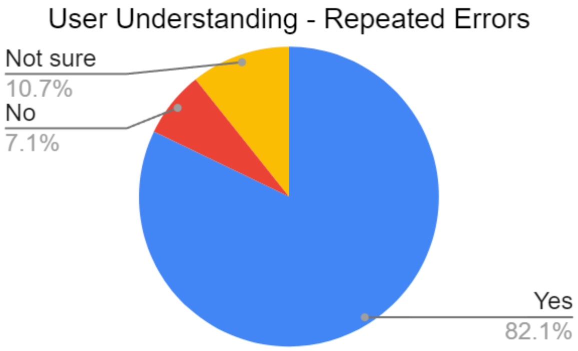


Figure 6.4: Percentage of participants understanding, not understanding, and not sure about the given explanation

Metric	Result
M1: Number of instances where users anticipated an explanation but did not receive one.	0
M4: Percentage of total trigger activations among users who anticipated guidance.	100%
M5: Percentage of total trigger activations among users who did not anticipate guidance.	88.89% (8/9)
M9: User ratings on the helpfulness among users who did not anticipate guidance. (Mean)	4.38
M10: Analysis of reasons why users did not read the given explanation.	Every user who received an explanation read it.
M11: Proportion of users who understood the explanation.	82%
M13: Ratio between successful and unsuccessful task completions after users read the explanation.	3 : 1
M14: Correlation between reading explanation and successful task completion. (chi-square test of independence, $\alpha = 0.05$)	Insufficient evidence of a statistically significant relationship (p-value=0.56).
M15: Proportion of users who apply the given guidance in the subsequent tasks.	Irrelevant for this trigger.
M16: Analysis of total time spent to complete tasks between users who read and did not read explanations.	Irrelevant for this trigger.

Table 6.4: Results of the further analysis in regards to the defined metrics for repeated error trigger

Table 6.4 reveals that about one-forth of the participants who engaged with the explanation were unable to complete the task successfully. Despite the invalid username errors that occurred during the interaction, some participants did not expect to receive guidance; however, over 88% of them activated the explanation trigger. When asked to rate the helpfulness of the explanation provided, those who did not expect to receive an explanation gave an average rating of 4.38. Furthermore, M15 and M16 are again considered irrelevant for the same reasons as the previous trigger.

6.2.3 Task Interruption - Change the number of items in the shopping cart

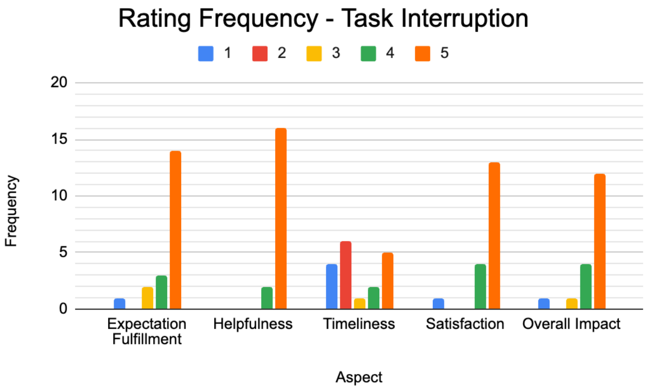


Figure 6.5: Rating distribution of the task interruption trigger

Figure 6.5 presents the rating distribution of the task interruption trigger. In the use case of changing the number of items in the shopping cart, 2 participants claimed to have received an explanation, although they actually did not receive one. Therefore, their feedback on the explanation was considered invalid and not further evaluated during the analysis, except for the aspect of expectation fulfillment. Within the other 10 participants who reported they did not receive an explanation, two of them actually received it, while the other eight did not activate the trigger during the study. The trigger was not activated either because the task was aborted before they completed it or because they were able to finish the task without an explanation.

Figure 6.5 shows a noticeable pattern in which the majority of ratings in various aspects leaned toward higher scores, with the exception of timeliness. In particular, over 85% of the responses in the five aspects assigned the two highest scores (4 or 5). On the other hand, for the **timeliness** aspect, less than 40% of the responses were rated 4 or 5. Another noteworthy trend is that the ratings for **helpfulness** were clustered in the 4 to 5 range, with a greater frequency of the highest rating, 5.

The descriptive statistics of the task interruption trigger in Table 6.5 support the findings from the frequency distribution chart of ratings, with a median and mode of 5 and a mean of at least 4.42 in all aspects except **timeliness**. It also reveals that **helpfulness** has a standard deviation of 0.37, suggesting that the opinions of the participants were highly consistent.

With a median and mode of 2, the timeliness aspect has a mean of 2.95 and a standard deviation of 1.58. This shows that the responses varied, with the majority of them being in the lower range.

Aspect	Mean	Median	Mode	SD
Expectation Fulfillment	4.45	5	5	1.05
Helpfulness	4.84	5	5	0.37
Timeliness	2.95	2	2	1.58
Satisfaction	4.53	5	5	0.96
Overall Impact	4.42	5	5	1.02

Table 6.5: Descriptive statistics summary of the task interruption trigger

Metric	Result
M1: Number of instances where users anticipated an explanation but did not receive one.	6
M4: Percentage of total trigger activations among users who anticipated guidance.	70%
M5: Percentage of total trigger activations among users who did not anticipate guidance.	60% (6/10)
M9: User ratings on the helpfulness among users who did not anticipate guidance. (Mean)	4.83
M10: Analysis of reasons why users did not read the given explanation.	Every user who received an explanation read it.
M11: Proportion of users who understood the explanation.	100%
M13: Ratio between successful and unsuccessful task completions after users read the explanation.	18 : 0
M14: Correlation between reading explanation and successful task completion. (chi-square test of independence, $\alpha = 0.05$)	Insufficient evidence of a statistically significant relationship (p-value=0.073).
M15: Proportion of users who apply the given guidance in the subsequent tasks.	Irrelevant for this trigger.
M16: Analysis of total time spent to complete tasks between users who read and did not read explanations.	Irrelevant for this trigger.

Table 6.6: Results of the further analysis in regards to the defined metrics for task interruption trigger

Table 6.6 offers a summary of further analysis results in regards to task interruption trigger. Among the cases observed, there were six instances where users expected an explanation but did not receive one. Moreover, only 70% of users who anticipated guidance actually received one. On the other hand, there were 60% trigger activations among users who did not anticipate guidance. Although all participants who read the given explanation were able to complete the task, there is no statistically significant correlation between reading the explanation and successful task completion based on the observed instances. In addition, all participants who received an explanation triggered by a task interruption confirmed that they understood the explanation. This indicates the clarity of the explanation provided, as all respondents were able to comprehend its content without difficulty.

6.2.4 Repetitive Action - Search for a specific product

In the observed cases of explanation triggered by repetitive action, all 30 participants received an explanation during their interaction with the prototype. However, two of them claimed not to have received any, thus only 28 responses were collected in total. Figure 6.6 and Table 6.7 provide an overview of the ratings for metrics M3, M6, M8, M12, and M17. It is obvious that the ratings are heavily weighted toward the high scores. With a mean of more than 4.6 in all aspects and a mode and median of 5, the data also has a considerably low standard deviation. Timeliness has the highest standard deviation out of the five aspects, namely 0.72.

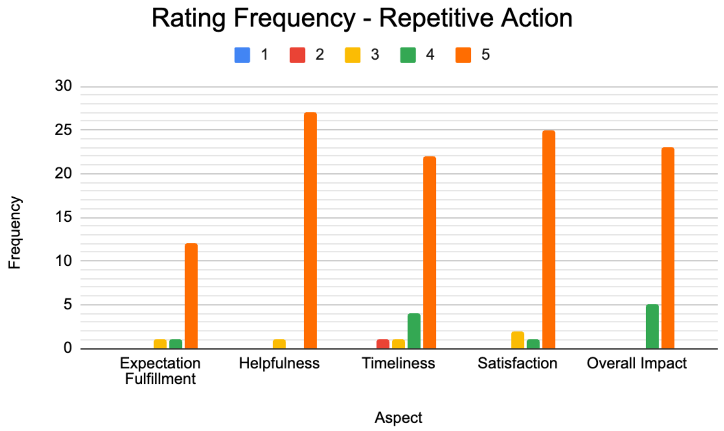


Figure 6.6: Rating distribution of the repetitive action trigger

Aspect	Mean	Median	Mode	SD
Expectation Fulfillment	4.79	5	5	0.58
Helpfulness	4.93	5	5	0.38
Timeliness	4.68	5	5	0.72
Satisfaction	4.82	5	5	0.55
Overall Impact	4.82	5	5	0.39

Table 6.7: Descriptive statistics summary of the repetitive action trigger

A summary of the further analysis carried out for the responses regarding this trigger is presented in Table 6.8. This table confirms the statement mentioned earlier that all users received an explanation, regardless of

Metric	Result
M1: Number of instances where users anticipated an explanation but did not receive one.	0
M4: Percentage of total trigger activations among users who anticipated guidance.	100%
M5: Percentage of total trigger activations among users who did not anticipate guidance.	100% (16/16)
M9: User ratings on the helpfulness among users who did not anticipate guidance. (Mean)	4.87
M10: Analysis of reasons why users did not read the given explanation.	Every user who received an explanation read it.
M11: Proportion of users who understood the explanation.	100%
M13: Ratio between successful and unsuccessful task completions after users read the explanation.	27 : 1
M14: Correlation between reading explanation and successful task completion. (chi-square test of independence, $\alpha = 0.05$)	Insufficient evidence of a statistically significant relationship (p-value=0.79).
M15: Proportion of users who apply the given guidance in the subsequent tasks.	86%

Table 6.8: Results of the further analysis in regards to the defined metrics for repetitive action trigger

whether they expected it or not. Additionally, Figure 6.7 shows the percentages of users who anticipated and who did not anticipate an explanation, with 53.3% of them did not anticipating one. However, on average, they still scored 4.87 for the **helpfulness** aspect of the given explanation. The success rate of this task is high, as only one out of 28 cases was unsuccessful among users who engaged with the provided explanation. Although all 30 participants received an explanation, only 28 participants were observed to have actually read it, as two of them claimed not to have received an explanation. In addition, all participants who read the explanation reported that they understood it.

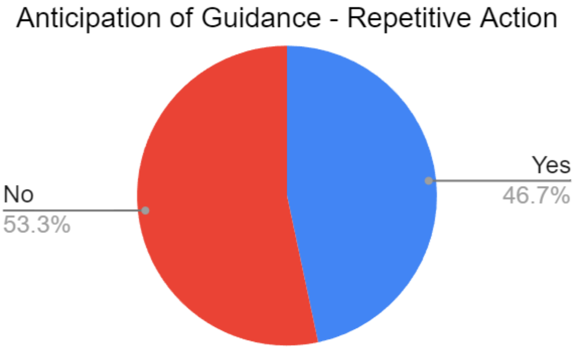


Figure 6.7: Percentage of participants anticipating and not anticipating an explanation triggered by repetitive action

Table 6.9 provides an overview of the analysis conducted to address metric M16, which measures the total time spent completing tasks by users who read explanations versus those who did not. However, due to the limited data available (only two instances of participants completing tasks without reading explanations), statistical tests could not be performed. Nevertheless, the table provides some insights. For instance, the fastest task completion time, 26 seconds, was recorded by a user who read an explanation. In contrast, the fastest completion time for a user who did not read an explanation was 49 seconds. Moreover, there is a difference of 12 seconds in maximum duration between the two cases. On average, participants who received an explanation completed tasks around 23 seconds faster than those who did not read an explanation.

Case	**Mean Duration (s)**	**Min Duration (s)**	**Max Duration (s)**
Read	41.62	26	67
Not read	64	49	79

Table 6.9: Overview of the analysis performed for the metric M16

6.3 Qualitative Data Analysis

This section contains examination of the responses gathered in relation to the metrics M2 and M7, focusing on the timing of trigger-based explanations and the need for additional guidance during task completion. Additionally, it also provides the evaluation of participants' opinions of trigger-based explanations in general. The data were examined through a qualitative analysis approach utilizing open, axial, and selective coding techniques, as mentioned previously. This approach allows us to extract insights and identify patterns in participants' responses.

6.3.1 First-time Use - Save a product to a collaborative collection

Several participants reported that no additional guidance was needed and some found the given explanation helpful. Although there were a number of participants who expected the given explanation to be provided earlier, some found the timing suitable. However, some users experienced confusion and suggested that the explanation should be provided not only earlier but also with a more intuitive interface. Table 6.10 illustrates the results of the open coding.

Code	Frequency
Shorter explanation	1
Earlier timing	6
User confusion	2
Unintuitive UI	5
Good timing	8
Provided guidance was helpful	2
No additional guidance needed	9

Table 6.10: Summary of code frequencies for responses on timing and need for additional guidance in the use case of saving a product to a collaborative collection

6.3.2 Repeated Errors - Create an account

Users expected additional detail in the explanations provided and needed better assistance from an interactive user interface. They also expressed the expectation of receiving explanations before or during the input process, not only after errors have occurred. Furthermore, their responses highlight the importance of designing a UI/UX that meets user expectations, is intuitive, and provides clear and immediate explanations. Some participants still found these aspects lacking in the implementation and commented about unusual username requirements. Another important point is that one user expressed that despite receiving an explanation, they still could not complete the task. Table 6.11 gives an overview of some points mentioned in the responses.

Code	Frequency
No additional guidance needed	8
Positive feedback	1
Clarity issue	5
Unmet condition details	2
UI improvement suggestion	10
Timing issue	25
Additional explanation at the start	12
Uncommon username requirements	3
Ineffectiveness	1

Table 6.11: Summary of code frequencies for responses on timing and need for additional guidance in the use case of creating an account

6.3.3 Task Interruption - Change the number of items in the shopping cart

A number of users reported that they did not need any additional guidance. However, some of them expected a guidance at the beginning as soon as the shopping cart is opened. There were also comments about the unusual mechanism of the use case and suggestions for improving the UI elements to make the process smoother. For example, by differentiating the design between clickable and non-clickable elements, or by providing an edit icon. One participant mentioned that it was a waste of time trying to figure out what to do. On the other hand, another participant also mentioned that the explanation was good to complement a UI with minimal information. These points are summarized in Table 6.12.

Code	Frequency
No additional guidance needed	6
Positive feedback	3
Uncommon mechanism	1
Unintuitive UI	8
Earlier timing	11

Table 6.12: Summary of code frequencies for responses on timing and need for additional guidance in the use case of changing quantity in shopping cart

6.3.4 Repetitive Action - Search for a specific product

Code	Frequency
No additional guidance needed	15
Positive comments on presentation	2
Appropriate timing	8
Chance for exploration	2
Effectiveness and efficiency	5
UI/UX improvement	5
Earlier timing	5

Table 6.13: Summary of code frequencies for responses on timing and need for additional guidance in the use case of searching for a product

Some points that were mentioned in the responses are shown in Table 6.13. Most participants mentioned that they did not need additional guidance because the guidance provided was sufficient. A few also complimented the simple and clear explanation and the interesting design. Some participants felt that the timing was good to allow the user to explore before giving a hint so as not to overwhelm and disrupt the new user experience. While some felt the explanation was given at the right time, others would have liked to have received it earlier. In addition, there was good functionality in terms of effectiveness and efficiency, as the explanation was helpful in speeding up the process or showing an unnoticed feature. One participant mentioned that it was so good that they used it again in all subsequent tasks. In terms of UI/UX, there were some suggestions for improvement to make the search icon more noticeable.

6.3.5 Trigger-based explanations in general

Table 6.14 provides an overview of some points that were mentioned in participants' opinion regarding trigger-based explanations in general. The responses revealed varied opinions. While some participants valued the explanation provided by trigger-based explanations and found them beneficial, especially for unfamiliar tasks or in scenarios where they felt lost or uncertain, others found them unnecessary, particularly when the website's design already conveyed sufficient information. Some expressed that, regardless of the explanation, the website itself should also be user-friendly. Several participants highlighted that explanations do not always fit all scenarios and have to be designed based on their context. Additionally, there were suggestions for improving the clarity and visibility of trigger-based explanations, such as highlighting important elements or providing more concise and focused guidance.

Code	**Frequency**
Earlier timing	4
Useful and helpful	14
Effectiveness and efficiency	6
Presentation and content improvement	3
Negative experience	3
Explanation complements UI	4
Contextual use only	7
UI improvement suggestions	4
Timely	2
Understandable	3

Table 6.14: Summary of code frequencies for responses on participants feedback regarding trigger-based explanations and their relevance to user experience

Chapter 7

Discussion

7.1 Answers to the Research Questions

Answers to the research questions are presented in this chapter based on the results and evaluation from the literature review as well as the user study. The defined goal-question-metrics are the guidelines for reviewing the quality and impact of the triggers.

RQ 1 What are the possible indications that users need an explanation?

There were many connections that could be drawn from the literature review and the workshop. First, user inactivity can be an indication of a need for explanation. This could indicate that users need additional guidance in a decision-making situation or are unsure about making certain inputs, for example, because of privacy concerns or because they are wondering about the outcomes of their inputs. Inactivity can also lead to another possible indication, which is the unusual duration of time taken to complete a particular process. In addition, in a situation where users are confronted with unexpected system behavior, possible indications of the need for explanation may include navigating back to previous pages or steps.

When users do not know how to use the system or perform certain functions, potential explanation triggers could be unusual mouse movements, task interruptions, or repetitive actions. These activities may indicate that users need guidance and more information to finish the task. Post-software updates or first-time use are also possible triggers, as users may need help understanding how to use the system or certain new features. Changes after software updates might cause confusion, especially for users who were familiar with the feature before the update. Finally, errors or repeated errors were also discussed as a signal of a need for explanation. Users may look for information about the causes and solutions of the errors.

Additionally, based on findings from previous studies, it was also found that some users prefer to receive guidance only when they ask for it. An example of this case also came up during the workshop, which was contacting a chatbot for customer support. This is a valid and clear sign of the user's need for explanation, however, this thesis focuses more on identifying the user's action to provide an explanation before the user actively asks for it.

RQ 2 What effects do trigger-based explanations have on the overall user experience?

RQ 2.1 How do users evaluate the impact of trigger-based explanations on their user experience?

RQ 2.2 How do trigger-based explanations influence task completion outcomes?

First-Time Use

Based on the collected results, the general impression of the first-time use trigger seemed to be positive. It was activated by all users who expected an explanation, and the explanations provided were found to be helpful. In fact, even among users who did not anticipate it, the average helpfulness rating was notably high, at 4.5. In addition, users expressed that they did not feel the need for additional guidance, suggesting that the triggered explanation adequately supported them throughout the task. Although there were some users who expected the explanation to be triggered earlier, most of the users felt that the timing was suitable. This is also supported by the timeliness ratings, where more than 50% of users who received an explanation gave a rating of 4 or 5. These factors indicate a high suitability of the trigger.

Furthermore, users found the timing to be effective, as was reflected in the high ratings for helpfulness and overall impact on the user experience. While no statistically significant relationship was evident between reading the explanation and task completion success, all users who read the triggered explanation were able to complete their tasks successfully. This suggests that at least reading the explanation ensured task completion. Although this study did not explicitly measure whether the explanation triggered by first-time use accelerated task completion, the high success rate indirectly suggests efficiency. Finally, user satisfaction was noticeably high, as reflected in median and mode ratings of 5 across several aspects such as fulfillment of expectations, helpfulness, timeliness, satisfaction, and overall impact on the user experience.

Repeated Errors

Users generally felt that receiving explanations after repeated errors was somewhat suitable. However, they expressed displeasure with the timing of this trigger, which they felt was inappropriate. Many users mentioned that the explanation should have been provided before or during the input process, or at least after the first failed attempt. This expectation likely stems from common practice in similar scenarios, where input requirements are typically displayed in advance. Users expected to receive an explanation immediately after the first error in order to quickly understand the problem and its solution. Therefore, it is important to carefully evaluate situations to determine whether it is more reasonable to provide an explanation after a single error, after multiple errors, or even before the input process. The timing in such cases can be determined, for example, by how many users are familiar with the information. If many users do not have the needed information to even start the operation, it would be beneficial to trigger the explanation before the operation. On the other hand, triggering an explanation at the beginning could be disruptive for users who are familiar with the process. In such cases, it may be more appropriate to trigger an explanation after certain error thresholds or unexpected user behavior that indicates that the users may not have the necessary information. Moreover, the results also show that while some users found the explanation helpful, others felt that their expectations were not being fulfilled.

In relation to effectiveness and efficiency, the results show that they are considerably low. One-fourth of the users could not complete the task even after receiving an explanation. This might be influenced by the clarity problem that the participants had, as some of them mentioned that they needed more detailed information. Also, some of them explicitly mentioned that the input conditions were unusual and not easy to understand. Although the uncommon conditions were intentionally designed, this also indicates that the content or wording of the explanation in the prototype was not very well-written. One way to improve the effectiveness of this trigger is to provide a very detailed explanation of why the errors occurred and how to fix them. For example, in the study, instead of simply stating that the given username did not meet the requirements and showing the requirement list, it might have been more understandable and helpful to explicitly state which exact requirement was not met and why the given username did not meet it.

Furthermore, despite the timing issue, the satisfaction aspect was moderately positive. However, the responses were fairly distributed across all scores, from one to five. Overall, the results show that most users agreed that timing and lack of detailed information were the biggest problems in completing the task. However, these were very likely influenced by the prototype to some extent, which did not mimic the common and well-known use case of creating a username. Repeated errors are, for example, more appropriate for common functionalities that the software expects users to know how to perform. The feedback in this study may indicate that the use case implemented in the prototype did not simulate this scenario very well, although repeated errors might be more useful in other different scenarios.

Task Interruption

Based on the ratings and explicit comments from the users, the task interruption trigger was perceived to be minimally suitable. For this specific task, an explanation was only provided if the user tried to remove all items from the shopping cart. That means that for those users who anticipated an explanation but did not intuitively remove all items from the shopping cart, the trigger was not directly activated. As a result, they did not receive an immediate explanation and spent considerable time trying to figure out how to complete the task. This suggests that the trigger for this case was not optimal, as it was not activated to fulfill the explanation needs of the users.

However, despite the low rating for timeliness and some unmet expectations, participants still found the explanation helpful. Interestingly, some users did not expect an explanation during the task completion, and the helpfulness rating among them was exceptionally high, with an average of 4.83, very close to the highest rating. This implies that there was a strong need for explanation or guidance from the users to be able to accomplish the task. Unfortunately, the trigger was not activated at the right moment.

The timing issue of the trigger also affected the other aspects of satisfaction, effectiveness, and efficiency. Although the satisfaction ratings show that users are satisfied with the explanation, the low timing ratings, which average less than 3, suggest that users were not satisfied with the trigger. Since the majority of the responses expected the explanation to be shown earlier, these responses imply that the trigger was not effective and efficient in delivering the explanation to users when they needed it. However, even though the quality of this trigger was minimal in the study, this trigger may still be suitable and useful in cases where users instinctively remove all items from the shopping cart or interrupt a particular task because they are not sure how to proceed.

Repetitive Action

The results show that 53.3% of the users did not expect an explanation, but all 30 users received one. Interestingly, the average helpfulness rating of the users who did not expect an explanation was significantly high, at 4.87. This means that even though the users did not feel the need to receive an explanation, the explanation still provided useful information. The perception of high suitability is also supported by high user ratings for expectation fulfillment, with low variability in responses. Moreover, there was also positive feedback regarding effectiveness and efficiency. Users mentioned that the explanation showed a solution to their problem, helped speed up the process, and helped point out an unnoticed feature. The results also show that, on average, users who read the explanation were able to complete the task faster. Another sign of effectiveness and efficiency is the fact that 86% of users used the guidance in the subsequent tasks, indicating that the guidance had a positive impact during the first application. In addition, users were very satisfied with this trigger. This is evidenced by the fact that the average ratings for meeting expectations, helpfulness, timeliness, satisfaction, and impact on the overall user experience were higher than 4.6, with most of the ratings for all of these aspects also scoring 5.

Overall

Among the implemented triggers in this study, repetitive actions emerged as the most highly rated trigger. This trigger facilitated users in understanding certain operations or in executing them more efficiently. Moreover, the timely explanation also came as a nice surprise to those who did not anticipate it. Additionally, the first-time use trigger also positively influenced the user experience and most users reported that the triggered explanation was delivered timely.

Finally, the timing of the explanations triggered by repeated errors and task interruption in this study was perceived as highly inappropriate by users, leading to its ineffectiveness. In both use cases, many users reported that the explanation came too late, indicating that their needs for an explanation were not fulfilled at the right time. Another factor contributing to the ineffectiveness of the repeated error trigger is that some users felt that the explanation did not provide detailed and comprehensive information. However, one of the possible factors for the rather low ratings of these triggers is that they might not be implemented or simulated in the appropriate use cases in this study. If they were implemented in the appropriate scenario, they might be able to better meet the user's need for explanation and have a better impact on the user experience.

7.2 Limitations and Threats to Validity

This section outlines the limitations and threats to validity of this Research. These potential threats are divided into four different categories: construct validity, internal validity, inferential validity, and external validity, as proposed by Wohlin et al. [23].

Limitation in Literature Review

Despite efforts to review a wide range of papers, the results may not include all existing papers relevant to this topic. However, the results of the literature study were more or less connected to the results of the brainstorming session in the workshop, where triggers were defined more concretely. Nevertheless, the literature review did provide some insights to answer the research questions and lay a foundation for the concept of triggers.

Construct Validity

The intentional design of the prototype with low usability and user-friendliness may not fully represent real-world scenarios. This approach could also induce confusion and negative emotional states in participants, potentially affecting their perception of the explanations provided. However, it was necessary to do this in order to elicit these specific user behaviors for research purposes.

Internal Validity

The differences between the online and in-person environments during the study may have influenced user behavior and responses differently. To minimize this threat, participants were advised to sit in a quiet room in both environments to create a similar environment. In addition, the use of screen recording during user interactions could potentially affect user performance due to the awareness of being observed, although not directly. To make participants feel less observed, participants' interactions were not directly monitored in both the online and in-person studies. During the online study, participants were informed that I would not be looking at the screen during their interaction with the prototype.

Conclusion Validity

Because there was not enough variability in the cases, statistical tests might lack the power to detect significant relationships. Consequently, even when statistical tests were conducted, the lack of significance in the results could be due to this low variability. Additionally, in some cases, statistical tests were not carried out for this reason. The mean and other measures were presented to help provide insights and aid in the interpretation of the results.

External Validity

The relatively small sample size and specific demographics of the participants may not be representative of the larger population. Moreover, the inclusion of participants known to me may also introduce the potential for social desirability bias, despite efforts to encourage them to share honest feedback and to remind them that the focus of the study was on the results and not on their individual task performance. Furthermore, while the study aims to explore triggers for explanations in software systems in general, the user study was conducted in the context of an online shopping website scenario. Thus, all tasks were centered around this specific scenario. However, the selected triggers were conceptualized within a broader context as described in Section 4.1, rather than being specifically tailored to the online shopping website context. Hence, the triggers were designed to be as universally applicable as possible across different scenarios and software systems.

Chapter 8

Conclusion and Future Works

8.1 Conclusion

The main objective of this thesis was to develop and evaluate a concept and implementation of trigger-based explanations within software systems. The focus of the research was to evaluate whether the implemented triggers effectively address the user's need for explanation in the appropriate context and at the appropriate time, and how it impacts the user experience. To achieve this goal, various triggers were implemented and assessed in a user study. The study began by defining research questions and establishing goal-question-metrics to guide the research and serve as benchmarks for evaluation.

A literature review was then conducted to identify existing studies relevant to the topic. However, it was found that existing research, while addressing different types of explanation needs, did not provide concrete signals of potential explanation needs during user interaction with software systems. To address this lack of information, a workshop was carried out to elicit and investigate more specific signals of explanation needs. Through this workshop, several concrete indicators of explanation needs were identified, such as user confusion revealed by unusual behavior, periods of inactivity, or returning to previous steps in the user interaction flow.

Following the collaborative workshop, four triggers were selected and conceptualized: first-time use, repeated errors, task interruption, and repetitive action. These triggers were identified as potential indicators of explanation needs during user interaction with software systems, based on the insights gained from the workshop discussions. Subsequently, a prototype incorporating these selected triggers was implemented. The triggers in this prototype were then evaluated through a user study. The results of this user study revealed diverse feedback from the participants, highlighting the varying preferences and needs of users, in line with prior research findings [4, 10, 19, 20] and the outcomes of the workshop.

The evaluation assessed various aspects such as expectation fulfillment, helpfulness, satisfaction, and task completion success rate. However, their timeliness was emphasized as the most critical aspect in determining the performance of the triggers. Timeliness in this context refers to the promptness with which the triggered explanation was provided when the need for explanation arose. The results showed that out of the four implemented triggers, the repeated action trigger received the highest ratings, followed by the first-time use trigger. Most participants agreed that the explanations triggered by these two signs were delivered promptly. In addition, they were perceived as suitable, effective, and efficient, which had a positive impact on user satisfaction and experience in the scenarios provided. On the contrary, explanations triggered by repeated errors and task interruptions were considered not optimal to meet the explanation needs of the majority of participants.

However, several potential factors may have influenced the evaluations negatively to some extent. First, the use cases in the prototype were intentionally designed with low usability and user-friendliness in order to elicit specific user behaviors for research purposes. For this reason, the user interface was intentionally designed to be non-intuitive compared to typical software designs. Another factor to consider is that the triggers may be implemented or simulated in inappropriate use cases during this study. If implemented in appropriate scenarios, it is possible that they would have better met the users' need for explanation and positively impacted the user experience.

Based on the results and these potential influencing factors, it can be concluded that when designing and implementing trigger-based explanations, careful consideration must be given to which trigger is appropriate for which specific use cases, as the effectiveness of each trigger also depends on the scenario. Furthermore, while some participants acknowledged the usefulness of trigger-based explanations in general, many expressed frustration with the unintuitive user interface of the prototype. This suggests that while trigger-based explanations can have a positive impact on the user experience, they alone cannot mitigate usability issues. Therefore, in order to optimize the user experience, it is essential to ensure intuitive UI design and usability from the start, with trigger-based explanations serving as a complementary tool for specific scenarios.

8.2 Future Works

This section explores potential directions or improvements for future research and development in the field of trigger-based explanations, building upon the insights gained from this study.

Exploration of Other Triggers

Some triggers, although discussed in the workshop, were not explored within this study's scope due to time and resource constraints. Future research could examine these unexamined triggers or even investigate completely new triggers not covered in this work. Such explorations would expand the investigation of the wide variety of potential triggers and their appropriateness in various scenarios.

More Contextual and User Diversity

Including a wider range of user groups in future evaluations may be beneficial since the need for explanations varies significantly across users and contexts. Future work could focus on comparing how different demographics respond to the triggers or how the triggers perform across various types of software systems, for example, gaming platforms, streaming platforms, communication tools, etc. Investigating these differences can reveal whether certain triggers are context-specific or if they have a more universal application.

Comparative Study

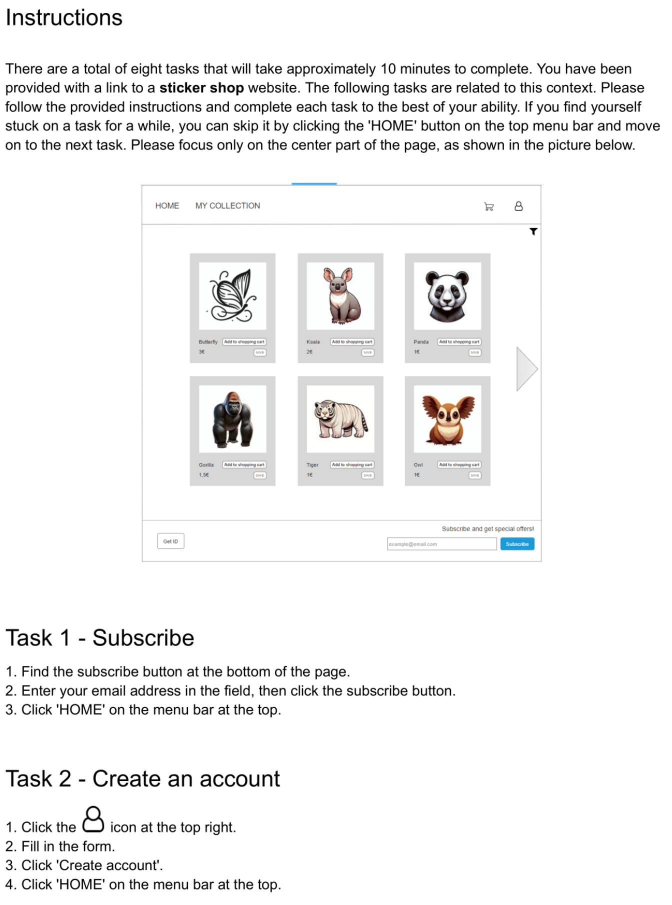
This study only involved a single prototype incorporating trigger-based explanations. An alternative research approach could involve comparing two distinct user groups: one interacting with a prototype that provides explanations after certain behaviors, and another where such behaviors do not trigger explanations. This comparative study could analyze both situations and investigate how much trigger-based explanations contribute to enhancing software quality and the user experience.

Real-World Applications

Another focus of future work could be to integrate trigger-based explanation systems into real software products used by real users. The advantage of this approach is not only to collect authentic feedback but also that with a larger number of participants, usability and user-friendliness might not necessarily have to be decreased in order to induce certain behaviors. Furthermore, including brief feedback surveys for users who receive the triggered explanations could provide invaluable insights for continuous refinement.

Appendix A

Task Instructions



Instructions

There are a total of eight tasks that will take approximately 10 minutes to complete. You have been provided with a link to a **sticker shop** website. The following tasks are related to this context. Please follow the provided instructions and complete each task to the best of your ability. If you find yourself stuck on a task for a while, you can skip it by clicking the 'HOME' button on the top menu bar and move on to the next task. Please focus only on the center part of the page, as shown in the picture below.

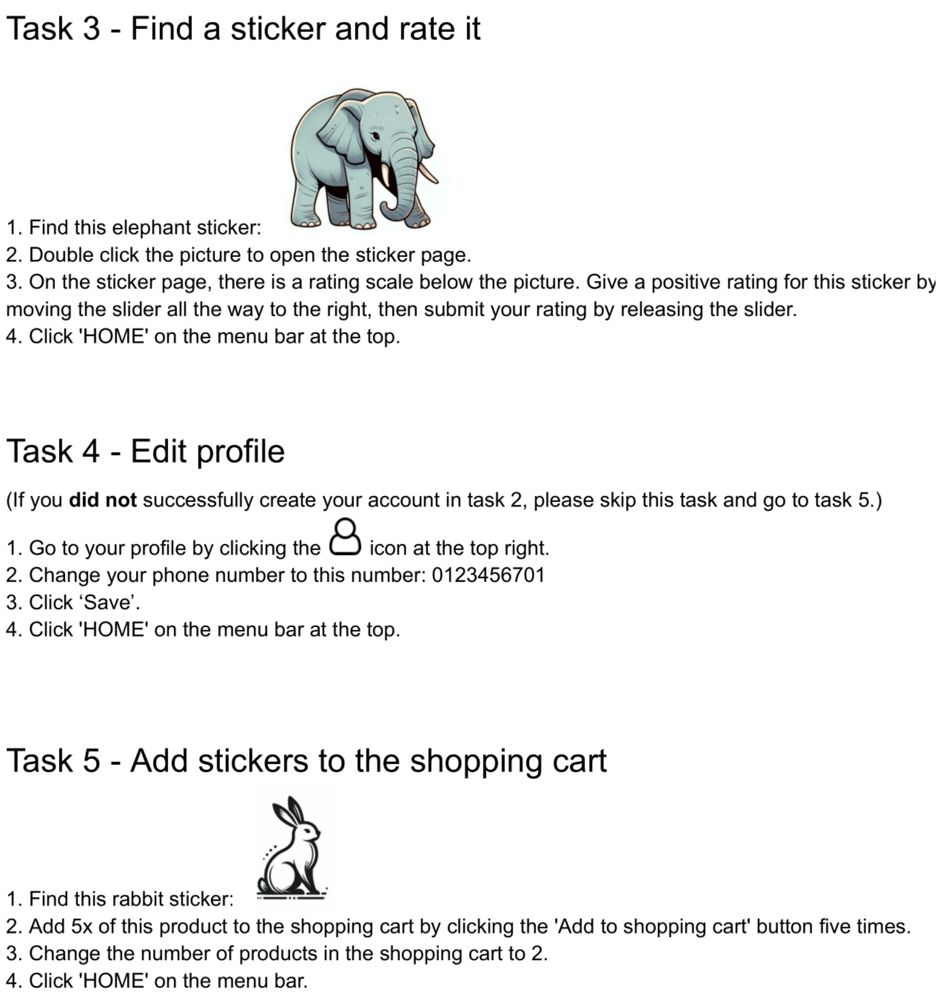
Task 1 - Subscribe

1. Find the subscribe button at the bottom of the page.
2. Enter your email address in the field, then click the subscribe button.
3. Click 'HOME' on the menu bar at the top.

Task 2 - Create an account

1. Click the 👤 icon at the top right.
2. Fill in the form.
3. Click 'Create account'.
4. Click 'HOME' on the menu bar at the top.

Figure A.1: First page of the task instructions



Task 3 - Find a sticker and rate it

1. Find this elephant sticker:
2. Double click the picture to open the sticker page.
3. On the sticker page, there is a rating scale below the picture. Give a positive rating for this sticker by moving the slider all the way to the right, then submit your rating by releasing the slider.
4. Click 'HOME' on the menu bar at the top.

Task 4 - Edit profile

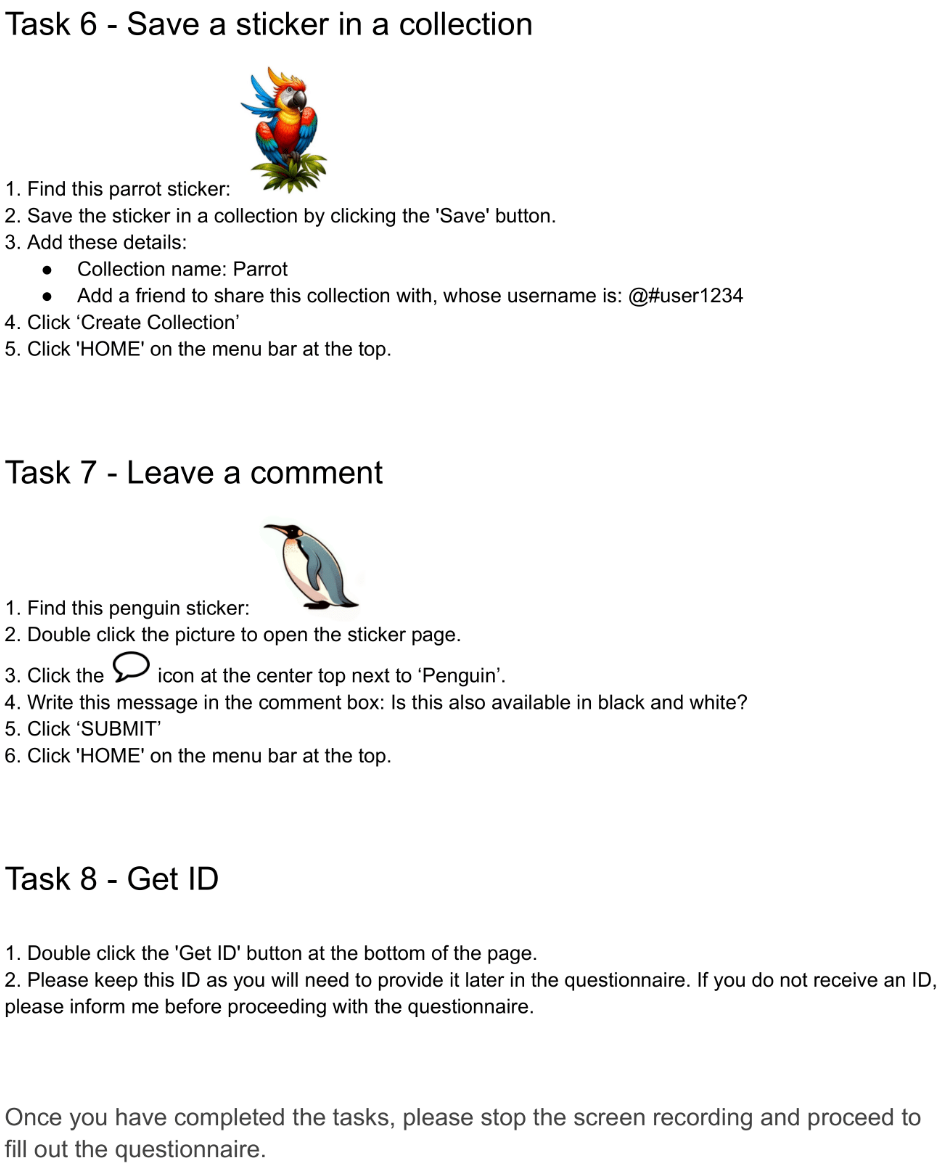
(If you **did not** successfully create your account in task 2, please skip this task and go to task 5.)

1. Go to your profile by clicking the 👤 icon at the top right.
2. Change your phone number to this number: 0123456701
3. Click ‘Save’.
4. Click 'HOME' on the menu bar at the top.

Task 5 - Add stickers to the shopping cart

1. Find this rabbit sticker:
2. Add 5x of this product to the shopping cart by clicking the 'Add to shopping cart' button five times.
3. Change the number of products in the shopping cart to 2.
4. Click 'HOME' on the menu bar.

Figure A.2: Second page of the task instructions



Task 6 - Save a sticker in a collection

1. Find this parrot sticker:
2. Save the sticker in a collection by clicking the 'Save' button.
3. Add these details:
    - Collection name: Parrot
    - Add a friend to share this collection with, whose username is: @#user1234
4. Click ‘Create Collection’
5. Click 'HOME' on the menu bar at the top.

Task 7 - Leave a comment

1. Find this penguin sticker:
2. Double click the picture to open the sticker page.
3. Click the 💬 icon at the center top next to ‘Penguin’.
4. Write this message in the comment box: Is this also available in black and white?
5. Click ‘SUBMIT’
6. Click 'HOME' on the menu bar at the top.

Task 8 - Get ID

1. Double click the 'Get ID' button at the bottom of the page.
2. Please keep this ID as you will need to provide it later in the questionnaire. If you do not receive an ID, please inform me before proceeding with the questionnaire.

Once you have completed the tasks, please stop the screen recording and proceed to fill out the questionnaire.

Figure A.3: Third page of the task instructions

Appendix B

Questionnaire

Questions	**Answer Options**
What is your age	Under 18, 18 - 24, 25 - 34, 35 - 44, 45 - 54, 55 or older
What is your gender?	Male, Female, Diverse, Prefer not to say
How would you rate your proficiency with technology and software?	Novice, Beginner, Intermediate, Advanced, Expert
How often do you use online shopping websites?	Daily, Several times a week, Once a week, Several times a month, Once a month, Rarely, Never

Table B.1: Demographics questions in the user study

Questions	**Answer Options**
Was there a moment during the task where you anticipated receiving guidance?	Yes, No
On a scale of 1 to 5, to what extent does this explanation meet your expectations for guidance during the task completion?	1,2,3,4,5
Did you receive the explanation during the task?	Yes, No, Not sure
Did you read the explanation during the task completion? (This question was only presented if the participant answered Yes in the previous question)	Yes, No
Why did you not read the explanation? (This quesiton was only presented if the participant answered Yes in the previous question)	Disturbing, Overwhelming, Misclick, Reflexively closed or ignored it, Other
Did you understand the explanation during the task completion?	Yes, No, Not sure
On a scale of 1 to 5, to what extent did you find the explanation helpful?	1,2,3,4,5
In your perspective, was the timing of the explanation suitable during your task completion?	1,2,3,4,5
Please elaborate on your response regarding the timing of the explanation. For example, if you rated the timing as not very suitable, could you provide suggestions or insights on when you believe the explanation should have been presented during your task completion?	
On a scale of 1 to 5, how satisfied were you with the provided explanation?	1,2,3,4,5
How would you characterize the impact of the provided explanation on your overall experience while completing the task?	1,2,3,4,5
Were there specific moments during the task completion where you felt an additional explanation would have been helpful? If so, when?	

Table B.2: Questions in the feedback survey of the user study

List of Figures

2.1 Representation of Goal-Question-Metric Framework 6

3.1 Workflow diagram of the brainstorming session. 17

4.1 Homepage of the website prototype featuring navigation options and product page 35
4.2 Collection page from the website prototype, featuring a pop-up introducing the collaborative collection feature 36
4.3 Profile page of the website prototype displaying an error message for an invalid username and the corresponding list of username requirements 37
4.4 Shopping cart view in the website prototype, displaying a confirmation dialog and informational message about how to change item quantities 38
4.5 Triggered explanation appears following the detection of repetitive actions, providing information about the search feature 39
4.6 Utilization of the search bar, showcasing stickers filtered by keywords in their names 39

6.1 Rating distribution of the first-time use trigger 49
6.2 Percentage of participants understanding and not understanding the given explanation 50
6.3 Rating distribution of the repeated error trigger 51
6.4 Percentage of participants understanding, not understanding, and not sure about the given explanation 53
6.5 Rating distribution of the task interruption trigger 54
6.6 Rating distribution of the repetitive action trigger 56
6.7 Percentage of participants anticipating and not anticipating an explanation triggered by repetitive action 57

A.1 First page of the task instructions 75
A.2 Second page of the task instructions 76
A.3 Third page of the task instructions 77

List of Tables

3.1 Overview of the findings source from the literature review . . 14

4.1 An overview of the first-time use trigger concept in this thesis 28
4.2 An overview of the repeated errors trigger concept in this thesis 29
4.3 An overview of the task interruption trigger concept in this thesis 30
4.4 An overview of the repetitive action trigger concept in this thesis 31
4.5 Use case demonstrating the first-time use trigger in a scenario of saving a sticker in a collaborative collection 35
4.6 Use case demonstrating the repeated errors trigger in a scenario of creating an account 36
4.7 Use case demonstrating the task interruption trigger in a scenario of changing the number of items in the shopping cart 37
4.8 Use case demonstrating the repetitive action trigger in a scenario of searching for a specific product 38

5.1 Levels of technical proficiency and definitions, provided as answer options in the questionnaire. 44

6.1 Descriptive statistics summary of the first-time use trigger . . 49
6.2 Results of the further analysis in regards to the defined metrics for first-time use trigger 51
6.3 Descriptive statistics summary of the repeated error trigger . 52
6.4 Results of the further analysis in regards to the defined metrics for repeated error trigger 53
6.5 Descriptive statistics summary of the task interruption trigger 55
6.6 Results of the further analysis in regards to the defined metrics for task interruption trigger 55
6.7 Descriptive statistics summary of the repetitive action trigger 56
6.8 Results of the further analysis in regards to the defined metrics for repetitive action trigger 57
6.9 Overview of the analysis performed for the metric M16 58

6.10 Summary of code frequencies for responses on timing and need for additional guidance in the use case of saving a product to a collaborative collection 59
6.11 Summary of code frequencies for responses on timing and need for additional guidance in the use case of creating an account 59
6.12 Summary of code frequencies for responses on timing and need for additional guidance in the use case of changing quantity in shopping cart 60
6.13 Summary of code frequencies for responses on timing and need for additional guidance in the use case of searching for a product 60
6.14 Summary of code frequencies for responses on participants feedback regarding trigger-based explanations and their relevance to user experience 61

B.1 Demographics questions in the user study 79
B.2 Questions in the feedback survey of the user study 80

Bibliography

[1] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[2] 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. IEEE, 2021.

[3] V. R. B. G. Caldiera and H. D. Rombach. The goal question metric approach. *Encyclopedia of software engineering*, pages 528–532, 1994.

[4] 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.

[5] L. Chazette, W. Brunotte, and T. Speith. Explainable software systems: from requirements analysis to system evaluation. *Requirements Engineering*, 27(4):457–487, 2022.

[6] L. Chazette, O. Karras, and K. Schneider. Do end-users want explanations? analyzing the role of explainability as an emerging aspect of non-functional requirements. In *2019 IEEE 27th international requirements engineering conference (RE)*, pages 223–233. IEEE, 2019.

[7] L. Chazette, V. Klös, F. Herzog, and K. Schneider. Requirements on explanations: a quality framework for explainability. In *2022 IEEE 30th International Requirements Engineering Conference (RE)*, pages 140–152. IEEE, 2022.

[8] L. Chazette and K. Schneider. Explainability as a non-functional requirement: challenges and recommendations. *Requirements Engineering*, 25(4):493–514, 2020.

[9] H. Deters, J. Droste, M. Fechner, and J. Klünder. Explanations on demand-a technique for eliciting the actual need for explanations. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 345–351. IEEE, 2023.

[10] H. Deters, J. Droste, and K. Schneider. A means to what end? evaluating the explainability of software systems using goal-oriented heuristics. In *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering*, pages 329–338, 2023.

[11] J. Droste, H. Deters, R. Fuchs, and K. Schneider. Peeking outside the black-box: Ai explainability requirements beyond interpretability. In *REFSQ Workshops*, 2024.

[12] J. Droste, H. Deters, M. Obaidi, and K. Schneider. Explanations in everyday software systems: Towards a taxonomy for explainability needs. In *2024 IEEE 32nd international requirements engineering conference (RE)*. IEEE, 2024.

[13] J. Droste, H. Deters, J. Puglisi, and J. Klünder. Designing end-user personas for explainability requirements using mixed methods research. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 129–135. IEEE, 2023.

[14] H. Koziolek. Goal, question, metric. *Dependability Metrics: Advanced Lectures*, pages 39–42, 2008.

[15] O. Kuzmin, D. Naumov, and D. Zakharov. The application of relational interactive logic to build logical triggers in event-driven systems. In *IOP Conference Series: Materials Science and Engineering*, volume 919, page 052012. IOP Publishing, 2020.

[16] Y. Liu, A. Hamou-Lhadj, J. Li, and Q. Lu. Observability and explainability for software systems decision making. *IEEE Software*, 41(1):45–49, 2023.

[17] A. Rosenfeld and A. Richardson. Explainability in human–agent systems. *Autonomous agents and multi-agent systems*, 33:673–705, 2019.

[18] M. Sadeghi, V. Klös, and A. Vogelsang. Cases for explainable software systems: Characteristics and examples. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 181–187. IEEE, 2021.

[19] K. Sokol and P. Flach. One explanation does not fit all: The promise of interactive explanations for machine learning transparency. *KI-Künstliche Intelligenz*, 34(2):235–250, 2020.

[20] N. Tintarev and J. Masthoff. Effective explanations of recommendations: user-centered design. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 153–156, 2007.

[21] M. Unterbusch, M. Sadeghi, J. Fischbach, M. Obaidi, and A. Vogelsang. Explanation needs in app reviews: Taxonomy and automated detection. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 102–111. IEEE, 2023.

[22] A. Vogelsang. Explainable software systems. *it-Information Technology*, 61(4):193–196, 2019.

[23] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in software engineering*. Springer Science & Business Media, 2012.

[24] H. Wu and S.-O. Leung. Can likert scales be treated as interval scales?—a simulation study. *Journal of social service research*, 43(4):527–532, 2017.