ICT, Strategy & Policy
www.tno.nl
+31 88 866 50 00
info@tno.nl

TNO 2024 R11770 – 20 September 2024

Usability aspects of specialized software tools

Author(s)	Alexandr Vasenev, Nan Yang
Classification report	TNO Public
Title	TNO Public
Report text	TNO Public
Number of pages	19
Number of appendices	0
Project name	UserTools
Project number	060.61474

Contents

Executive summary

The User Tools project aimed to increase TNO-ESI knowledge on usability of software tools.

This 3-month exploratory project surveyed the literature, interviewed three TNO-ESI industry partners on their needs, and scanned the usability knowledge landscape in Dutch universities.

The Tracy (log analysis) tool served as a case to study the usability role in tool adoption.

The project brought the following insights to TNO-ESI:

- practitioners see usability of SW tools as an important topic (specifically aspects of learnability, memorability, and ease of installation);
- usability knowledge experts from TU/e (Eindhoven University of Technology) were identified;
- TNO-ESI is in an advantageous position to investigate usability of specialized software tools by building on learnings from other domains;

Interviews with TNO-ESI experts provided the following insights:

- A structured approach to usability engineering (including various activies like user testing) is missing;
- As most TNO-ESI tools build upon a conceptual (methodological) solution, TNO-ESI can explicitly consider the split of: (1) Usability of the method or methodology and (2) Tool usability per se.

The literature scan suggests that TNO-ESI should employ a mixed set of usability methods to get complementary insights into tool usability. A further investigation is needed to understand how these methods can be applied in different stages of tool evolution.

The project revealed several potential actions for TNO-ESI:

- Investigate ways to improve learnability, intuitiveness, and memorability of tools. E.g., create an AI-based assistant to answer users’ questions about tools;
- Design and conduct user testing with industry partners to help companies building a business case for tool adoption;
- Co-organize a hackathon to collect usability feedback;
- Co-supervise students to conduct usability studies of TNO-ESI tools;
- Involve implementation partners in early tool development stages. It will help to ease the transition from a prototype to an TNO-ESI tool, while ensuring continuous tool support.

The project suggests to further investigate the following research questions:

- How to assess (TNO-ESI) methodologies while focusing on users?
- How to approach usability in various stages of a tool (or methodology) evolution?

Introduction

Tasks of high-tech industry engineers, e.g., diagnostic experts, are complex. TNO-ESI provides tools and methodologies to improve user's efficiency and effectiveness. Yet, tools should be *usable* to assist engineers in absorbing these solutions.

Diverse user needs shall be effectively addressed at every step of the tool development, to ensure their transition from a Demonstrator to a Professional tool (Figure 1). Each evolution step has specifics. First, the 'Repeated use' implies support of several (but somewhat different) cases. Then, in the step of involving 'User community' new users are introduced, who were not familiar with the tool. This makes ease of use and learning a particularly important topic. Moreover, at this step earlier usability issues are clearly exposed. Afterwards, a viable business case need to be elaborated.

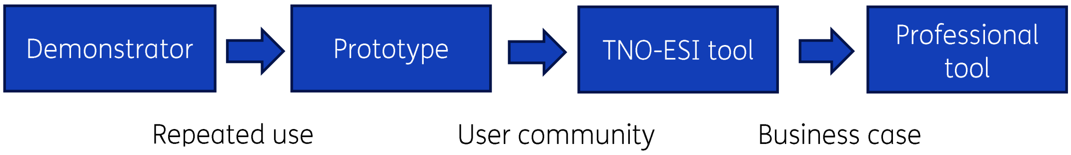

Figure 1. Tool evolution flow

Better understanding state-of-the-art usability practices will help TNO-ESI to effectively address needs of high-tech industry professionals. For instance, identifying relevant design and testing guidelines will help to consistently account for users' concerns and feedback. Having engineers, acting as users, in the tool development process will provide insights if they can effectively use the tools and identify what needs to be improved.

Usability is a recognized quality of SW and complex systems (see, e.g., ISO 9126 or 25010). It is addressed via specialized methodologies. In SW engineering, for instance, eye-tracking, prototyping, and analysis of screen recordings can be used. To ensure the solutions are designed for efficiency and are user-friendly, their validation should include:

- Quantitative metrics and objective data to comprehensively assess their impact, usability, and efficiency;
- Analysis of user experiences through structured experiments or usability testing.

This report focuses on the following research questions:

RQ1. What are needs of TNO-ESI industrial partners w.r.t. usability of software tools?
RQ2. What knowledge and facilities are available at TNO-ESI academic partners?

Several other questions, e.g., "How to effectively assess TNO-ESI solutions (tools and methodologies) while focusing on users?" and "How to deal with usability in different stages on the TNO-ESI tool maturity framework?" are suggested for future studies.

Study methodology

Figure 2 shows the project methodology. It included two major steps: Literature scan and Interviews that lead to several categories of findings. We took a software log analysis tool (called Tracy) as a use case to learn industry needs.

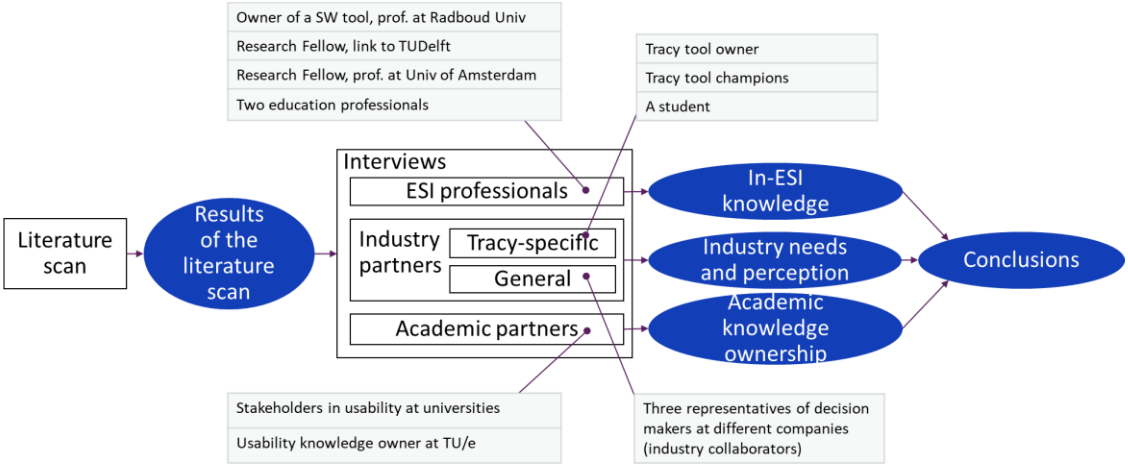
Owner of a SW tool, prof. at Radboud Univ
Research Fellow, link to TUDelft
Research Fellow, prof. at Univ of Amsterdam
Two education professionals
Tracy tool owner
Tracy tool champions
A student
Literature scan
Results of the literature scan
Interviews
ESI professionals
Industry partners
Tracy-specific
General
Academic partners
In-ESI knowledge
Industry needs and perception
Academic knowledge ownership
Conclusions
Stakeholders in usability at universities
Usability knowledge owner at TU/e
Three representatives of decision makers at different companies (industry collaborators)

Figure 2. Methodology steps

We selected interviewees from several groups:

- *Applied TNO-ESI research professionals*: three senior research fellows and two education professionals. This group helped to ensure the applied nature of the usability disussions, while linking industrial and academic domains. One of the interviewees is owner of an advanced software tool used in several companies; each of the research fellows is afficiated with a Dutch university. Each of the interviewees has significant experience with applied research projects in the high-tech industry domain.
- *Industry partners* group included:
  - *Tracy-specific:* Professionals related to a case study of two tool champions from different companies, and a student involved in conducting a research task at one of the companies. The aim was to deepen our understanding of practical views on usability of software tools;
  - *General:* Three representatives of decision makers from different companies (to investigate tool adoption topics relevant to companies);
- *Academic partners of TNO-ESI* (as elaborated in a section below).

The findings were grouped in several categories:

- Literature scan results;
- Outcomes of interviews with TNO-ESI professionals;
- (High-tech) industry needs and perception; and
- Academic knowledge experts (in Dutch universities).

The report structure follows these categories one-by-one and then provides overall conclusions.

Literature scan results

This section provides an overview of key findings from the literature scan. It aims to inform practitioners on usability definitions, methods, and practices in literature.

Usability definition

Relevant definitions were identified as follows. First, a subset of articles was identified via knowledge databases (Scopus and Google Scholar) using prases 'usability definitions' and 'usability software tools'. Then, the identified articles were studied for references to seminar papers on usability and standards, as shown below.

Usability definitions vary. This study adopted the following definition:

> "Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" (ISO9241-11, 1998).

This definition is widely recognized, including in NIST 800-63-3 (Grassi et al., 2017).

Usability aspects (sometimes called components) vary in different sources. This report doesn't adopt a particular structure of usability aspects. The usability aspects listed in the three examples below can be used to discuss particular topics with practitioners, construct questionnaires, or usability test protocols.

Example 1: Nielsen (Nielsen, 2012) identifies:

- Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the (interface) design?
- Efficiency: Once users have learned the design, how quickly can they perform tasks?
- Memorability: When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- Satisfaction: How pleasant is it to use the design?

Example 2: ISO 9126 from 2001 "Software engineering — Product quality" lists the following aspects:

- Understandability: The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use;
- Learnability: The capability of the software product to enable the user to learn its application;
- Operability: The capability of the software product to enable the user to operate and control it;
- Attractiveness: The capability of the software product to be attractive to the user;
- Usability compliance: The capability of the software product to adhere to standards, conventions, style guides or regulations relating to usability.

Example 3: In 2011, ISO 25010 “Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models” updated those aspects as follows:

- Added: "user error protection" and "accessibility" (use by people with a wide range of characteristics);
- Renamed "understandability" to "appropriateness recognizability" (a user can recognize the system is appropriate for their needs), and "attractiveness" to "user interface aesthetics".

Usability Validation Methods

To evaluate the usability of software tools, a range of methods can be employed to gather insights into how effectively a tool meets user needs (Nielsen Norman Group, 2024). By searching the keywords 'usability validation' and 'usability evaluation' on Google, we identified the relevant literature described below.

Table 1 summarizes a few methods that are characterized based on the extent of user involvement and the type of data they generate, whether qualitative or quantitative, attitudinal or behavioural. **User involvement** refers to how actively users participate in the evaluation process. **Qualitative data** typically consists of insights, opinions, or descriptive feedback from users, offering rich, in-depth information about their experiences. On the other hand, **quantitative data** includes measurable, numerical outcomes, such as task completion times or error rates, providing statistical evidence of usability. Additionally, the methods differ in capturing **attitudinal data**, which reflects users' feelings and perceptions, versus **behavioural data**, which focuses on users' actions and interactions with the software.

Table 1: Examples of existing validation methods

Methods	Characteristics
Paper prototyping	Light/no user involvement, Qualitative
Checklist validation	Light/no user involvement, Qualitative
Heuristic evaluation	Light/no user involvement, Qualitative
Cognitive walkthrough	Light/no user involvement, Qualitative
Interview	Heavy user involvement, Qualitative, Attitudinal
Questionnaire	Heavy user involvement, Quantitative, Attitudinal
Testing (task-based)	Heavy user involvement, Quantitative, Behavioural
Testing (think-aloud)	Heavy user involvement, Qualitative, Behavioural
User logging	Heavy user involvement, Quantitative, Behavioural

The first four methods—Paper Prototyping, Checklist Validation, Heuristic Evaluation, and Cognitive Walkthrough—are valuable for refining designs and identifying potential usability problems early, without heavily involving end users. Among these, Heuristic Evaluation and Cognitive Walkthrough specifically rely on usability expert analysis.

Paper prototyping involves creating low-fidelity sketches or models of the software interface. This method is primarily qualitative and is typically used early in the design process. It allows designers to explore ideas and get feedback on basic concepts before any coding begins. This method does not involve users directly in the testing phase but can provide valuable insights for refining the design before involving users.

Checklist validation is a method where a predefined list of usability criteria, user requirements or best practices is used to evaluate the software. This method is qualitative and can be performed without direct user involvement. It helps ensure that basic usability principles are adhered to in the design and can be used to catch potential issues before conducting user-based evaluations. Some examples of such checklist can be found in (UX Collective, 2016).

In **heuristic evaluation**, usability experts review the software based on established heuristics or guidelines, such as the well-known usability heuristics (Nielsen, 2024). This qualitative method involves experts rather than end users and is effective in identifying usability problems early in the design process. It provides a usability expert perspective on the usability of the interface, allowing for iterative improvements before user testing.

Similar to heuristic evaluation, a **cognitive walkthrough** involves experts who simulate a user's thought process when interacting with the software. This method is qualitative and focuses on understanding how a new user might learn to use the system. Like heuristic evaluation, it does not require direct user involvement but instead relies on expert judgment to predict potential usability issues. It is a task-based usability-inspection method that involves a crossfunctional team of reviewers walking through each step of a task flow and answering a set of prescribed questions. This method is particularly useful for identifying those aspects of the interface that could be challenging to new users.

The other methods listed in Table 1 directly involve users and provide both qualitative and quantitative data, depending on the approach. These methods are crucial for validating the software's usability from the user's perspective, ensuring the tool effectively meets their needs in real-world scenarios.

Interviews involve direct interaction with users to gather qualitative and attitudinal data about their experiences, needs, and perceptions. This method is user-centric and provides deep insights into how users feel about the software and what they expect from it.

Questionnaires are used to collect quantitative data from a large group of users. These can include both closed and open-ended questions to gauge user attitudes, preferences, and satisfaction levels. The quantitative nature of questionnaires allows for statistical analysis of user feedback.

Task-based testing involves observing users as they perform specific tasks with the software. This method is quantitative and focuses on measuring user performance, such as task completion time, error rates, and success rates. It provides objective data on how users interact with the software in a real-world context. This approach is typically structured as a controlled experiment where users are divided into two groups: one using the traditional tool and the other using the new tool to complete predefined tasks. Such a comparative study provides valuable insights for tool developers, enabling them to assess how effectively their software performs relative to existing tools.

In **think-aloud testing**, users are asked to verbalize their thoughts while interacting with the software. This qualitative method provides insights into the user's cognitive processes and uncovers usability issues that might not be evident from purely observational methods. It involves real users and focuses on understanding their experience and reasoning.

User logging involves capturing data on user interactions with the software over time. This quantitative method tracks behaviors such as navigation patterns, feature usage, and session duration. It provides a comprehensive view of how users interact with the software in their natural environment, offering valuable insights for ongoing usability improvements.

Best Practices and Challenges in Usability Evaluation for Tool Adoption

As software tools become increasingly complex, usability has emerged as a critical factor in determining their success and adoption. Users are more likely to adopt tools that are intuitive, efficient, and easy to use. To achieve this, designers and developers must prioritize usability throughout the design process, employing systematic approaches to evaluate and refine the user experience. This section introduces literature related to the challenges, and best practices in usability evaluation. The literature was identified by searching keywords "usability practice and challlenges" on Google Scholar.

Addressing usability in the context of Model-Driven Engineering (MDE) has been recognized as a common challenge. A study by (Whittle et al., 2013) provides valuable insights into the challenges and practices related to MDE tool adoption. The study employs an interview-based approach, gathering insights from practitioners who use MDE tools in their daily work. The findings reveal that many of the challenges related to the adoption of MDE are tool-related, ranging from usability concerns to the overall effectiveness of the tools in supporting the development process. (Weber et al., 2019) suggests that evaluating the usability of MDE tools is particularly challenging because their use often requires domain-specific knowledge, making them more complex than general-purpose tools. These studies emphasize the need for special attention to the usability of MDE tools to ensure their successful integration into the development workflow.

To create solutions that prioritize usability, designers should focus not only on traditional usability aspects but also on integrating usability principles into the design process. According to Gould (1985), there are three fundamental principles for system design. First, designers should **focus early on users and tasks**, considering users' cognitive, behavioural, and attitudinal characteristics, along with understanding the nature of the tasks users will perform. Second, usage should be **measured empirically**, with early testing using simulators or prototypes to collect data, ensuring design decisions are informed by real user interactions. Lastly, the design process should be **iterative**, repeating steps of redesign, testing, and measurement to continuously improve usability.

Research has suggested that different kinds of usability evaluation methods are complementary to each other and by applying a mixed method tool builders could discover various kind of usability issues. For example, the study conducted by (Weber et al., 2019) explores the usability of development tools through a detailed case study of the Eclipse 4diac IDE, a tool used for Model-Driven Software Engineering (MDSE). The researchers apply various usability evaluation methods to assess the tool, discovering various types of issues, such as syntactic issues that are related to user interactions and user flows, and semantic issues that hinder the mental models of users and system comprehension. Several literature review studies investigate how practitioners apply these usability evaluation methods. For example, the review by (Paz & Pow-Sang, 2016) surveys the frequency of different usability evaluation methods used in the software development process. Their analysis shows that surveys and questionnaires are the most commonly used methods (26%), followed by user testing (14%) and heuristic evaluation (13%). Interviews (10%) and think-aloud testing (10%) are also frequently employed. Other methods such as user logging are less frequently used. These studies emphasize the effectiveness and usefulness of different usability methods in identifying and addressing usability issues.

Several studies derived best practices from Open Source Software community. For example, (Terry et al., 2010) studies the perceptions and practices of usability in the open source software community. The results show that project members often overlook the significance of their ad-hoc practices in creating usable software. The most effective practices suggested by practitioners include organizing open content projects, akin to hackathons (i.e., inviting users to perform a task with the tool), and involving UX designers to enhance the user experience. These collaborative settings enable users to engage directly with the tool, offering valuable insights that inform its continuous improvement. Enabling users to easily report usability issues during software usage is also crucial, as it allows developers to quickly identify and address problems in real-time. Central to these practices is the interaction between developers and **core users** — a key group that consists of **reference users** and **bleeding-edge users**. Both types of users play an essential role in providing high-quality, respected feedback, which is often explicitly solicited by developers as they design and implement new functionality. **Reference users** are valued for their domain expertise and extensive experience with the software. These users typically operate in stable environments and use the tool to accomplish practical tasks, providing feedback on its usability, efficiency, and integration within real-world workflows. Their insights help ensure the tool meets the practical needs of its broader user base, making them a crucial part of the design process. **Bleeding-edge users**, by contrast, are highly engaged with the latest developments of the software, often using nightly builds and early versions to test new features. These users are at the forefront of exploring the tool's capabilities, often pushing it to its limits in innovative and experimental ways. Their feedback helps developers identify bugs, usability issues, and feature gaps that may not be apparent to less intensive users. The close relationships that develop between these core users and the developers foster an organic form of **participatory design**, where user input directly influences the iterative improvement of the software. This collaboration ensures that the tool evolves in a user-centered way, aligning with the real-time needs and feedback of its most engaged users. The software benefits from continuous refinement, driven by the active involvement of these core user groups.

Practical guidelines from usability experts also offer cost-effective strategies for usability engineering. For example, (Nielsen, 2020) from the Nielsen Norman Group suggests three steps for rapid iteration: 1) using simple prototypes for early testing, 2) conducting heuristic evaluations, and 3) performing sample user testing. These steps help streamline the usability process while maintaining a strong focus on improving user experience.

As takeaway messages from the literature scan, TNO-ESI could consider several actions to enhance tool usability. First, employing a mixed set of usability evaluation methods would provide more comprehensive insights, especially by exploring methods that are not frequently applied in practice but are highly recommended in the literature, such as **heuristic evaluation and usability testing**. Additionally, TNO-ESI could host **hackathons** to engage users in real-world tasks, providing valuable feedback on tool functionality. To further improve usability, a mechanism should be developed that allows users to easily report issues encountered during software usage. Finally, it is important to determine how different evaluation methods can be applied at various stages of tool evolution, ensuring continuous usability improvements throughout the development lifecycle.

Outcomes of interviews with TNO-ESI experts

This project interviewed three senior research fellows and two education professionals from TNO-ESI. It provided an overview of:

- the context in which usability is addressed at TNO-ESI;
- whether usability principles are recognized and followed; and
- insights on how to approach usability topics.

The context of developing software tools plays an important role in addressing usability. TNO-ESI mission is “Embedding cutting-edge methodologies into the Dutch high-tech systems industry in order to cope with the ever-increasing complexity of their products” (from www.esi.nl). In other words, the *domain of interest* is high-tech systems.

The task of *embedding methodologies* (that include tools, techniques, and methods) aims to address the ever-increasing product complexity. To note, TNO-ESI tools address specific cases (i.e., advanced tasks of professional high-tech system developers) and:

- require domain-specific (or even company-specific) knowledge;
- need general methodological knowledge to develop or tailor fit-for-purpose methodologies; and
- require timely and practical introduction (as part of methodologies).

During the interviews these aspects were well recognized. Interviewees higlighed that the tool usage and users’ perceptions play an important role in embedding methodologies.

Out of the three usability principles, two are already being followed: ‘Focus early on users and tasks’ and ‘design iteratively’. The approach to co-develop solutions together with industrial companies facilitated that adoption. Examples are:

- One project started with having lunch-time face-to-face workshops with end users. It resulted in refining concepts, quickly resolving issues, unclarities, and roadblocks;
- In the second project a tool was co-developed with an industrial company, while an engineer continuously tried new features;
- The third project developed a new DSL (domain specific language) and held a workshop with users to evaluate it.

To note, the principle ‘measure the usage empirically’ was less recognized in the interviews.

Importantly, the role of the tool developers was noted as being crucial for the tool evolution. The developers are expected to support tool users and adjust the tool based on their needs.

The discussions provided the following insights:

- A structured approach to usability engineering (including various activies like user testing) is missing;
- As most TNO-ESI tools build upon a conceptual (methodological) solution, TNO-ESI can explicitly consider the split of: (1) Usability of the method or methodology and (2) Tool usability per se.

Industry needs and perception of usability

This project studied industry needs and perceptions of usability for software tools by interviewing our industry partners. Further, we are interested in understanding what the role of usability plays in tool adoption. To address this angle, we interviewed tool champions (who are promoters and often core users of the tool) and decision makers (who are familiar with the general tool adoption process within companies). The interviews were conducted in a semi-structured format, with topics and questions prepared in advance. During the interviews, we asked these questions along with any necessary follow-up questions. Afterward, we analysed the meeting notes to extract the interviewees' responses, which are summarized in Tables 2 and 3.

To understand the perspectives of tool champions, we take Tracy (Tracy, 2024) as a case study. Tracy is a log analysis tool that provides filtering, matching and colouring functionalities for engineers to analyse their log files. Tracy is interesting for usability discussion because 1) it is specialized in helping engineers in high-tech industry understand complex logs, 2) the users of Tracy are representative users of ESI tools, and 3) it was first developed together with company A and later introduced to company B.

Table 2 summarizes how the tool champions of Tracy view usability. Overall, the champions are very passionate about the concepts behind Tracy, especially the visual support feature, which is not seen in other log analysis tools. Both champions consider intuitiveness and learnability to be the most important usability aspects. Specifically, they do not want to have heavy user documents but an intuitive and self-explanatory tool interface. One of the champion shares that they would like to have a “Friday-afternoon-tool” which is easy to install and try on a Friday afternoon. A “Friday-afternoon-tool” can ease the adoption by having a few core users who can regularly promote and demo the tools to other engineers.

It is also observed that user requirements differ when moving from one company/user group to another. For example, company A prefers to integrate Tracy into the Visual Studio IDE because their software engineers are using it, while company B prefers to have a standalone application because the target users in company B are functional engineers who do not use an IDE. Another key observation is that companies often seek scalable, intuitive solutions tailored to their specific, narrow use cases. To meet these needs, they develop and implementing features on top of the existing Tracy platform. For instance, in response to scalability challenges (such as the need to analyse large logs), company B involves a student to reimplement Tracy in C#. However, when companies create their own versions of Tracy with custom features, it can lead to challenges in version control and complicate usability evaluations. Moreover, as features are coupled, it becomes difficult to thoroughly evaluate the tool across different versions, as the divergence from the core system grows.

Table 2: Perspective of Tracy Champions

	Company A	Company B
Interested features	- Visual support - State pattern matching	- Visual support - Comparison between logs - Colouring makers
Users	Software engineers	Physicists (with domain knowledge but little software knowledge)
Usability definition	Intuitive and learnable	
Importance of usability	Important	Very important (because target users are not necessarily software expert)
Main barrier of adoption	Limitation on input size	
Usability feedback collection	Ad-hoc feedback meetings and workshop with demo	Feedback meetings with student projects
Other possible improvements	- Visuals are not always intuitive - Missing overview when specifying state patterns	Usability evaluation has not conducted yet and there will be a student project focusing on usability
Current activities	No activity	Two students from an applied sciences university working on - log comparison - reimplementing using C# to solve scalability issue
Adoption process	No formal process. Users' satisfaction plays a major role. It usually starts with a very few engineers and is promoted by these engineers to a larger audience.	

To gain insights into the perspectives of decision-makers, we conducted interviews with representatives from three companies. These individuals, who have been involved in multiple TNO-ESI projects, play a significant role in project evaluation and tool adoption. The goal of the interviews was to understand how they perceive usability and tool adoption from a decision-making standpoint.

Table 3 summarizes how the decision makers perceive usability. It is observed that these interviewees see usability of tools as an important topic. When it comes to usability definitions, learnability and memorability come back as very important aspects. Interestingly, consistent with the view of TNO-ESI experts, the evaluation of a methodology is considered to be a different activity from the evaluation of a tool that implements that methodology.

When it comes to tool adoption, decision-makers seek quantitative insights, such as the efficiency and effectiveness of the tools. These metrics help guide their decisions and persuade colleagues to promote the tools within the organization. Other critical factors, such as ease of installation, ongoing support, and tool stability, are also essential considerations in the adoption process.

Table 3: Perspective of decision makers

	Company A	Company B	Company C
Usability definition – important aspects	• Match user's background • Intuitive, self-explanatory, and memorable • Stable • Good visuals	• Good visuals • Easy to install	• Match user's background • For a specific goal and purpose • No excessive training • Learnable and memorable
Role of usability in tool adoption	Important. The risk of TNO-ESI tools is that it is very specialized and complex, which raises challenges in • getting a large group • training and tool support • addressing users' concerns in knowledge reuse	Important. It should be very easy to get installed otherwise people lose interests	Important. Especially when the users don't have software engineering background
Role of usability in tool development process	It gets more and more important in the tool evolution, especially when getting actual users: • Demonstrator: show the methodology is feasible for the target users • Prototype: allow target users to try the methodology with a stable and intuitive tool • ESI tool: the tool should be easy to install and learn		
Important usability measurement	• Learning curve • Users' happiness and satisfaction	• Productivity • Quantitative insights about what people use, and how they interact with the tools	• Learning curve • Productivity • User satisfaction
Tool adoption process	• Pilot some projects with students as a sidetrack • Have a go/no-go meeting, discuss similar efforts in the company and involve multiple stakeholders to align requirements	• A champion brings the idea to a SW managers meeting • Discuss if there are similar tools and efforts • Defines a business case and evaluate the gain and cost	• Both bottom-up and top-down tool adoption - depends on how wide is the impact
Other important factors in tool adoption	• Cost • Integration with current way of working and infrastructure • Developers' preference	• Relation to existing similar tools • License model and tool support • Integration with a current way of working and infrastructure • Performance	• Integration with current way of working and infrastructure • Scope of adoption in different departments and disciplines

Based on outcomes of interviews with tool champions and decision makers, the project suggest that TNO-ESI could consider:

- Investigating improving learnability, intuitiveness, and memorability of tools. This could be addressed by providing better documents and applying AI (Artificial Intelligence) to build a tool assistant which can answer developers' questions about tools.
- Designing and conducting user testing with industry partners. This helps companies to build a business case for tool adoption. By consolidating the results in scientific papers, we can present and promote our tools to academic and industry partners.

- Co-organizing a hackathon to collect usability feedback. It has been mentioned both by literature and our interviewees that inviting users to perform a task using the tool could be a fruitful way to collect user feedback.
- Supervising students to conduct usability studies of TNO-ESI tools.
- Involving implementation partners in the early phase of the development to address usability and to ease the transition from prototype to TNO-ESI tools with professional and continuous tool support.
- Studying how methodology can be assessed without having a mature tool in place.

Academic experts

To identify knowledge owners on the usability of software tools, this study surveyed the landscape of researchers in Dutch universities.

The survey process was organized as follows:

1. Due to the project focus on software tools, the project scope included the following Dutch universities that have software engineering groups:
   - Delft University of Technology (TU Delft);
   - Eindhoven University of Technology (TU/e);
   - Radboud University Nijmegen;
   - University of Amsterdam;
   - University of Twente;
   - Vrije Universiteit Amsterdam;
   - Groningen university.
2. Potential candidates were identified (typically, one or two persons per university):
   - via LinkedIn and a search engine using terms ‘usability’ and ‘software’;
   - using a snowballing method (recommendations of applied researchers);
3. We (study participants) discussed background and publications of the identified researchers and then selected whom to contact (up to two researchers per university);
4. The identified candidates were contacted via emails.
5. If a candidate expressed his/her interest in the topic, an online call followed.

To note, as the study was limited in time (3 months), which partially overlapped with the vacation season, only a limited number of contacts responded.

The study identified a particularly relevant contact for follow-up discussions and knowledge exchange – a lecturer on the topic of usability of software from TU/e. Other universities seem to focus less on the usability of professional software tools. The study also found several usability experts (e.g., on products and services) whose knowledge could be applied to the professional software domain. TNO-ESI is in position to assist such knowledge transfer from other domains.

A potential follow-up study can further explore the usability knowledge landscape (beyond Dutch universities). Also, other usability knowledge experts (not from software engineering groups) can be interviewed on their view of the software usability specifics.

Conclusions and recommendations

To answer two research questions, this study surveyed the literature, interviewed researchers and practitioners, and scanned the usability knowledge landscape in Dutch universities. The answers can be formulated as follows:

- **Research question 1**. What are needs of TNO-ESI industrial partners w.r.t. usability of software tools?
  Answer: Companies see the topic as important. They seek **scalable intuitive solutions** to their own specific cases. It can lead to simultaneously co-existing tool versions, even with overlapping capabilities. This results in difficulties in version control and usability evaluation. **Learnability, memorability, and ease of installation** are considered to be very important usability factors for tool adoption. Decision makers appreciate having **quantitative information** (e.g., the number of existing tool users and the required time to learn the tool**)** for decisions on tool adoption.

- **Research question 2**. What knowledge and facilities are available at TNO-ESI academic partners?
  Answer: Usability of **professional SW tools** is a less explored topic at Dutch universities (except TU/e). TNO-ESI is in a good position to study usability of specialized SW tools **by building on learnings from other domains**.

As a knowledge acquisition study, the project provided TNO-ESI with:

- an introduction to the (domain of) usability of professional SW tools;
- understanding how usability is perceived in industry, including tool champions and decision makers in high-tech systems industry;
- An initial overview of the software tools knowledge landscape (in Dutch universities).

Limitations of the study included:

- limited duration (3 months, partially during a vacation period);
- scoped efforts (2 part-time researchers);
- the project's approach to investigate a broader focus (investigate domain knowledge, conduct interviews, focus on a specialized tool, scan the usability knowledge landscape), rather than an in-depth investigation of a specific aspect.

The project suggests to further investigate the following research questions:

- How to assess methodologies while focusing on users?
- How to approach usability in different stages of a tool (or methodology) evolution? (see Figure 1 for the list of stages)

References

Gould, J.D., and Clayton L (1985). *Designing for usability: key principles and what designers think*. Communications of the ACM 28.3: 300-311.

Grassi, P. , Garcia, M. and Fenton, J. (2017), *Digital Identity Guidelines*, Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD, [online], https://doi.org/10.6028/NIST.SP.800-63-3 (Accessed September 20, 2024).

ISO 9241-11 (1998). *Ergonomic Requirements For Office Work With Visual Display Terminal (VDT)S – Part 11: Guidance On Usability.* International Organization for Standardization (ISO).

Nielsen Norman Group (2024). *Research methods. Research Methods Articles, Videos, Reports, and Training Courses*. Available at: Research Methods Articles, Videos, Reports, and Training Courses (nngroup.com) (Accessed September 20, 2024).

Nielsen, J. (2012). *Usability 101: Introduction to usability*. Nielsen Norman Group. https://www.nngroup.com/articles/usability-101-introduction-to-usability/ (Accessed September 20, 2024).

Nielsen, J. (2020). *Discount Usability 30 Years.* Nielsen Norman Group. https://www.youtube.com/watch?v=D5FgSAPa-Tw (Accessed September 20, 2024).

Nielsen, J. (2024). *10 Usability heuristics.* Nielsen Norman Group. https://www.nngroup.com/articles/ten-usability-heuristics/ (Accessed September 20, 2024).

Paz, F., & Pow-Sang, J. A. (2016). *A systematic mapping review of usability evaluation methods for software development process.* International Journal of Software Engineering and Its Applications, 10(1), 165-178.

Terry, M., Kay, M., & Lafreniere, B. (2010, April). *Perceptions and practices of usability in the free/open source software (FoSS) community.* In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 999-1008).

Tracy (2024). *Tracy tool.* Available at: https://esi.nl/research/output/tools/tracy (Accessed September 20th, 2024).

UX Collective (2016). *Examples of UX checklists.* Available at: https://uxdesign.cc/ux-checklists-bf2dda85787f (Accessed September 20th, 2024).

Weber, T., Zoitl, A., & Hußmann, H. (2019, September). *Usability of development tools: A case-study.* In 2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C) (pp. 228-235). IEEE.

Whittle, J., Hutchinson, J., Rouncefield, M., Burden, H., & Heldal, R. (2013). *Industrial adoption of model-driven engineering: Are the tools really the problem?* In Model-Driven Engineering Languages and Systems: 16th International Conference, MODELS 2013, Miami, FL, USA, September 29–October 4, 2013. Proceedings 16 (pp. 1-17). Springer Berlin Heidelberg.