ulm university universität
uulm

Universität Ulm | 89069 Ulm | Germany

Faculty of Engineering, Computer Science and Psychology
Databases and Information Systems Department

Measuring Learnability in Human-Computer Interaction

Master's thesis at Universität Ulm

Submitted by:
Manuela Unsöld
manuela.unsoeld@uni-ulm.de

Reviewer:
Prof. Dr. Manfred Reichert
Dr. Rüdiger Pryss

Supervisor:
Johannes Schobel

2018

Version from September 3, 2018

© 2018 Manuela Unsöld

This work is licensed under the Creative Commons. Attribution-NonCommercial-ShareAlike 3.0 License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/3.0/de/ or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Composition: PDF-LaTeX 2_ε

Abstract

It is well accepted that learnability is a crucial attribute of usability that should be considered in almost every software system. A good learnability leads within a short time and with minimal effort to a high level of proficiency of the user. Therefore, expensive training time of complex systems is reduced. However, there is only few consensus on how to define and evaluate learnability. In addition, gathering detailed information on learnability is quite difficult. In todays books on usability evaluation, learnability gets only few attention, research publications are spread to several other fields and the term *learnability* is also used in other context.

The objective of this thesis is to give an structured overview of learnability and methods for evaluation and additionally assist in the evaluator's individual choice of an appropriate method. First of all, several definitions of learnability are discussed. For a deeper understanding psychological background knowledge is provided. Afterwards, methods to asses learnability are presented. This comprises nine methods that seem particularly appropriate to measure learnability. As this methods are very diverse, a framework based on *analytical hierarchy process* is provided. This framework aims to classify presented methods with respect to certain criteria and assess practitioners in selecting an appropriate method to measure learnability.

Contents

1 Introduction **1**

1.1 Problem statement 2

1.2 Objective 3

1.3 Structure of the thesis 4

2 Fundamentals **5**

2.1 Learnability 5

2.2 Learning and Memory 10

2.2.1 Defining Learning 10

2.2.2 Human Memory 12

2.2.2.1 Multi-Store Model 12

2.2.2.2 Levels-Of-Processing Theory 14

2.2.2.3 Working Memory Model 15

2.2.2.4 Theories of Long-Term Memory 15

2.2.3 Expertise 17

2.2.3.1 Stages of Skill Acquisition 17

2.2.3.2 Learning Curves 18

2.2.3.3 Impact of Expertise on Chunking 23

2.2.3.4 Mental Models 23

3 Existing Methods for Measuring Learnability **25**

3.1 Overview 25

3.2 Testing Methods 29

3.2.1 Mental Model Interviews 29

3.2.2 Question-Suggestion Protocol 30

3.2.3 Performance Based Measurements 34

3.2.3.1 Performance Measurement Based on Learning Curves . 34

3.2.3.2 Analysing Trials-to-Criterion by Means of Range Statistic 38

3.3 Analytics . . . 41
3.3.1 Analysis of Log-files . . . 41
3.3.1.1 Learnability Evaluation based on Chunk Detection . . . 42
3.3.1.2 Petri Net Based Approach . . . 45
3.4 Inquiry Methods . . . 48
3.4.1 Questionnaires . . . 48
3.4.2 Diaries . . . 50
3.5 Formal-Analytical Methods . . . 52
3.5.1 Attributes Models . . . 52
3.5.1.1 A Learnability Attributes Model . . . 53
3.6 Inspection Methods . . . 57
3.6.1 Cognitive Walkthroughs . . . 58
3.7 Discussion . . . 62

4 AHP **71**
4.1 AHP Method . . . 71
4.2 Related Work . . . 74
4.3 AHP for Selecting Methods to Measure Learnability . . . 74
4.3.1 Problem Hierarchy . . . 75
4.3.2 Examples for Ratings of Criteria . . . 78
4.3.3 Ratings of Alternatives . . . 82
4.4 Discussion . . . 85

5 Conclusions **87**

A Appendix **101**

1

Introduction

The importance of an excellent user experience (UX) in human-computer interaction (HCI) is well known [1, 2]. For systems where users can freely choose between several alternatives (such as websites or mobile applications), a good user experience is a matter of survival as users leave if usage is too difficult and intransparent. But also in workplace, a well-designed system is crucial as it strongly influences the employees productivity [3].

Note that the aim of UX is not only to provide positive emotions, such as enjoyment, the all-encompassing fulfilment of desires and emotional attachment to the product. The core of UX is usability and utility (see Figure 1.1). Therefore, the interface with its offered functionality must be suitable for the user's tasks and allows users to effectively and efficiently achieving their goals. "In the best cases, the interface almost disappears, enabling users to concentrate on their work, exploration, or pleasure" [2]. Therefore, usability is not a *'nice to have'* exclusively influencing the user's satisfaction. It extremely affects the user's productivity and error mades. In critical environments, such as in air traffic, nuclear reactors and clinical care, a good usability might even be life saving. There are several famous examples where unintentional errors led to serious consequences. Just recently, an employee of the Hawaiian Emergency Management Agency (HEMA) had caused panic in Hawaii after he accidentally sent an emergency alarm warning of an incoming ballistic missile. The system suffered under a very obvious usability problem [5, 6].

One important attribute of usability is learnability [2, 7, 8], which generally can be described with how easy it is to learn the system. Some researchers even refer to

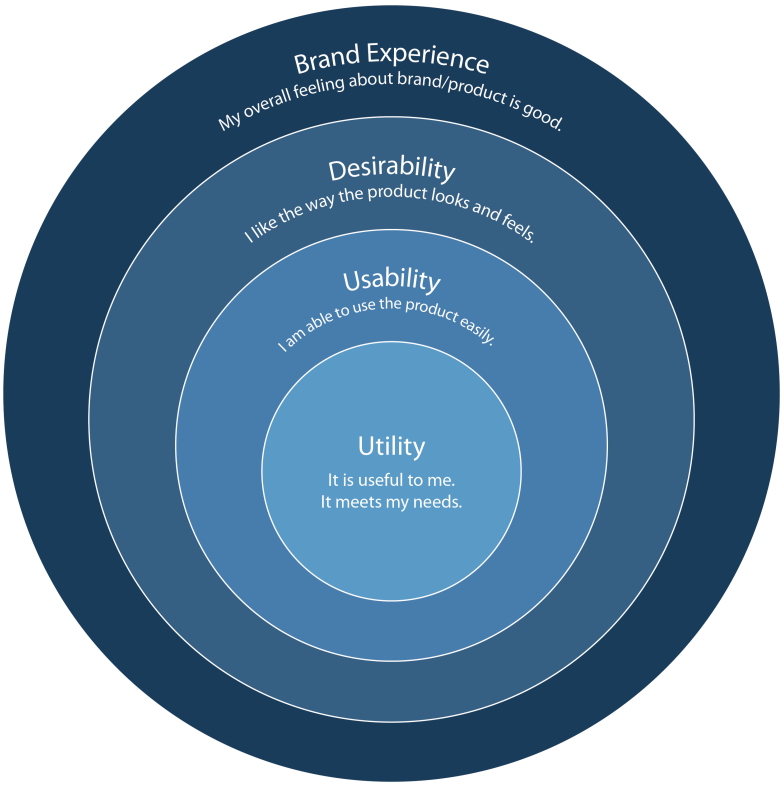


Figure 1.1: Aspects of user experience (adapted from [1, 4])

learnability as the most fundamental attribute [8] and highly recommend to involve explicit evaluation of learnability when evaluating usability [9].

Especially in industrial and commercial systems, where productivity and costs are crucial and training time is expensive, learnability is important. However, also in most other systems ease of learning is considerable. One example are social media applications, where users try a competitive supplier if they cannot succeed quickly [2].

1.1 Problem statement

Although researchers highly recommend to involve explicit evaluation of learnability, only few advise can be found in up-to-date known textbooks (e.g., [1, 2]). In research publications, there is only few consensus on how to evaluate learnability and even on

how to define learnability in first place [9]. Surprisingly, [8] supposed that learnability might one of the easiest aspects of usability to measure.

Over the last 40 years, learnability is of interest in HCI [9]. Since then, several definitions and evaluation methods have been conducted. Surprisingly, only one publication [9] could be found that performed an extensive literature research on existing approaches.

This situation is aggravated by the fact that practitioners looking for information on learnability are faced with the challenge of finding suitable literature. In todays books about usability evaluation methods, such as [1, 10], learnability gets only few attention. In research, the term *learnability* is applied to multitudinous other fields, such as artificial intelligence [9, 11], language and notation learning [12, 13], instructional technology [14] and psychological fundamental research [15]. Therefore, a relative lengthy literature search might be necessary to identify the relevant publications spread across different research fields.

Considering this information, it is not surprising that learnability is rarely explicit measured in practice [16].

1.2 Objective

Therefore, the overall goal of this thesis is to give a structured overview of learnability with its meaning and existing approaches to measure learnability.

In detail, the first sub-goal of this thesis is to provide an overview of the term *learnability* with its definition and the underlying process of learning from a psychological perspective.

The second sub-goal is to present and discuss existing approaches to measure learnability. As many different methods are applicable, all with their own strength and weaknesses, the third sub-goal is to give assistance in finding the most appropriate method for oneself. It is based on a decision process, the *analytical hierarchy process* (AHP), where personal preferences are utilized to propose the best fitting alternative to measure learnability.

1.3 Structure of the thesis

The following chapter, Chapter 2, starts with defining learnability and continues with a review of human learning from a psychological perspective in order to facilitate a comprehensive understanding of the term *learnability*.

Afterwards, in Chapter 3, several existing approaches are presented in detail after providing an overview. Finally, the existing methods are discussed.

The assistance in finding the most appropriate method regarding individual requirements and preferences is provided in Chapter 4. At the beginning of this chapter, the fundamental process, AHP, is explained. Next, related work is presented. Afterwards, AHP is applied for finding the most appropriate method. First, the general problem hierarchy is presented. Then examples are given on how the criteria of the hierarchy could be rated regarding different scenarios. Then, the presented approaches in this thesis to measure learnability are rated consistent with AHP. The chapter concludes with a discussion on the appliance on AHP to assist in finding the most appropriate method.

This thesis finish with a conclusion in Chapter 5.

2

Fundamentals

In order to understand how learnability can be measured, it is important to first comprehend what learnability actually means. Therefore, definitions of learnability are discussed first. Additionally, to fully comprehend learnability and the possibilities of measuring learnability, it is essential to understand basics of human learning processes. Hence, learning from a psychological point of view is presented afterwards.

2.1 Learnability

Although learnability is standardized by the *International Organization for Standardization* (ISO), there seems to be disagreement on how to define learnability, as many other popular definitions exist. This impression is reinforced by [9], which reviewed all articles mentioning the term *learnability* and published in the ACM conference series on *Human Factors in Computing Systems* (CHI) and *ACM Transactions on Computer-Human Interaction* (TOCHI). This led to a collection of 88 papers from the years 1982 to 2008. Entire 45 article used learnability without any definition. The remaining articles used various definitions, that [9] arranged in eight categories. For example, the definitions range from "easy to learn" to "change in performance over time" and the "[a]bilty to remember skills over time" [9]. In the following, only some of the definitions are presented, trying to cover as many different approaches as possible.

The first step is the standardization by ISO. Maybe one reason why it is not widely used when it comes to learnability is that the standard series by ISO are not accessible without charges and also within the standard series more than one definition is provided for

learnability, such as the definitions in ISO 9241-110:2006 and ISO/IEC 25010:20. In both standards learnability is regarded as a sub-characteristic of usability[1].

The first one, ISO 9241-110:2006, describes dialogue principles (see Figure 2.1), which are general goals that should be achieved in interactive systems to optimize usability. These comprise seven principles including learnability, which is referred to as *suitable for learning* in this standard. It is important to note that the individual principles are not independent and can overlap semantically [17]. For instance, *Conformity with users expactions* may affect learnability. Therefore, it is quite challenging to define and measure each principle individually. Some principles are also competing, so one has to weigh which principle is more important.

Figure 2.1: Dialogue principles of ISO 9241-110:2006 (own representation, based on [17])

[1] The term *Usability* will not be explained any further in this thesis. For clarification or further interest on this topic, reference is made to [8].

In this standard Learnability is simply described with:

> "A dialogue is suitable for learning when it supports and guides the user in learning to use the system" [17].

However, ISO/IEC 25010:2011 provides a more detailed definition:

> "[Learnability is] the degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use" [18].

The definition gets more concrete about what is meant by *the user can use the system*. He uses the "system with effectiveness, efficiency, freedom from risk and satisfaction" [18]. Additionally, the definition emphasizes that the specified user for whom the system is intended, specified goals of learning and the specified context of use need to be taken into account.

There are several other definitions that also emphasize the significance of characteristics of the users for learnability. One example is the following definition, which is one of the earliest definitions (from 1980), that could be found:

> "[T]he system should be easy to learn by the class of users for whom it is intended" [19].

However, this definition, as well as the definition by ISO 9241-110:2006, leaves unclear what is meant by *easy to learn* or *learn to use*.

One quite popular definition, that gets more concrete about what is meant by *easy to learn*, is by [8]:

> "Ease of learning refers to the novice user's experience on the initial part of the learning curve[,] [...] allow[ing] users to reach a reasonable level of usage proficiency within a short time".

For one thing, [8] refers exclusively to novice users focusing on their initial learning experience. Furthermore, [8] relativizes the relationship between efficiency and learnability (as given in the definition of ISO/IEC 25010:2011, for example). [8] states that

systems designed exclusively for high learnability will lead to an increase in efficiency, but the efficiency will maybe remain below the maximum possible value. Other way around it is the same: A system that will lead to high efficiency focussing on expert users will probably not have minimal learnability. This interdependency of learnability and efficiency will be described more detailed in Chapter 2.2.3.2. Therefore, [8] defines learnability as an aspect of usability, in addition to efficiency, which is defined as a separate aspect of usability (see Figure 2.2). Furthermore, he defines learnability not with achieving efficiency, but with "reach[ing] a reasonable level of usage proficiency within short time". However, [8] stays unclear on how to estimate a reasonable level of usage proficiency. Moreover, nothing is said about the transition from a reasonable to an expert performance.

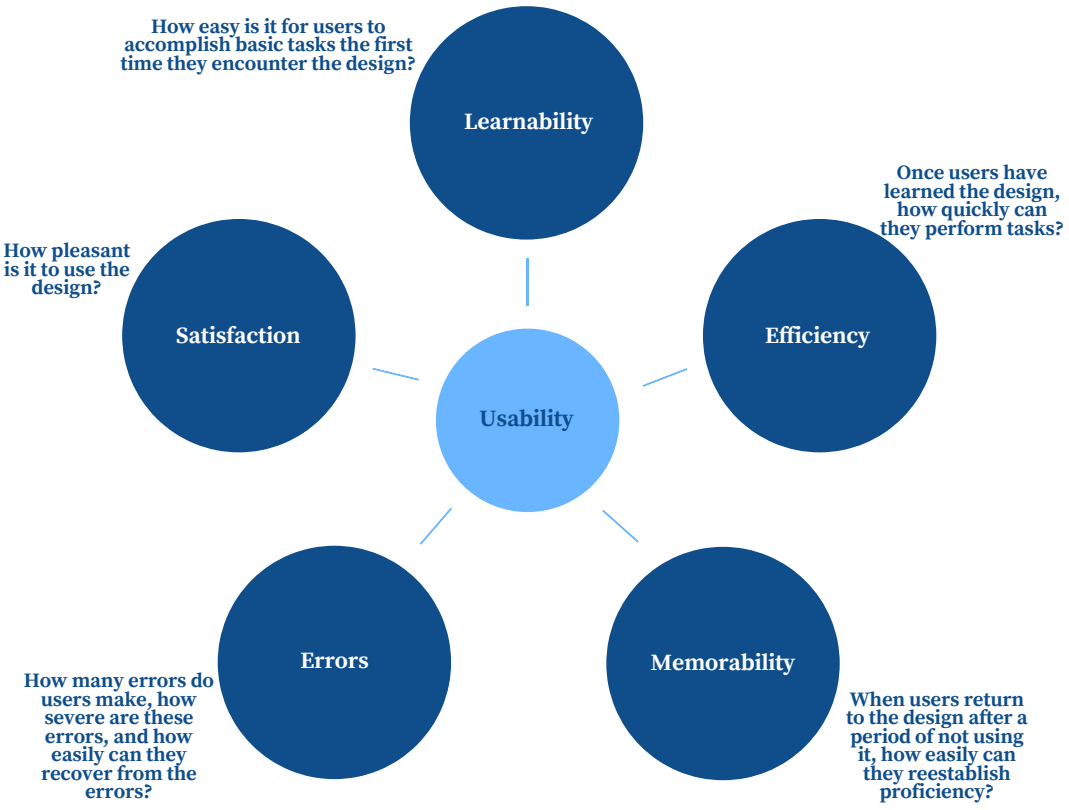


Figure 2.2: Usability attributes by [8] (own representation)

Similarly, [20] define learnability also with reaching a certain level of proficiency:

> "[T]he effort required for a typical user to be able to perform a set of tasks using an interactive system with a predefined level of proficiency".

As with [8], the focus is on initial learning, but at this definition *effort* is seen as an indicator for learnability and not *time*.

[21] takes *time* as well as *satisfaction* into account and asserts that the goal is an efficient and error-free interaction:

> "[T]he word *learnability* signifies how quickly and comfortably a new user can begin efficient and error-free interaction with the system, particularly when he or she is starting to use the system".

As seen in the last examples, most definitions focus on initial learning, but there are also definitions which explicit include long term learning such as:

> "[T]he ease with which new users can begin effective interaction and achieve maximal performance" [22].

Another example is by [23], which not only includes mastery of the basics, but also of the "advanced system functions".

There are many more definitions [9]. Now, how should learnability defined? How is learnability defined for this thesis? Which is 'the best' definition? I think every definition has its right to exist, each covering different aspects of learnability. Therefore, instead of choosing one of the existing definitions for this thesis or providing an own definition, I would like to give a small summary of important aspects that define learnability, based on the definitions found.

First, learnability has something to do with *learning to use the system*. This involves the ability to get some work done [8]. The user learn mainly how to use the basic functions, but also advanced functions [23]. The result of *learning to use the system* is a change in performance which can be observed over time [8]. This change results in a more efficient, effective and error-free usage [18, 21]. Critical for learnability is to which degree a change can be observed [8, 18], how quick this change takes place [8, 21], how much effort is required [20] and how satisfied the user is [18, 21]. Essential for the assessment

of learnability is always the consideration of the specified user and his concrete context of use [8, 18].

Although there are discussions about the demarcation of learnability and usability, mostly learnability is seen as an attribute of usability.

2.2 Learning and Memory

In the last chapter various definitions for learnability were presented. Often, learnability is simply described with how easy it is to learn the system. But, what does *easy to learn* really mean? What is learning in general?

Answering these questions is essential to fully understand learnability and especially for the understanding of possibilities to measure learnability. Therefore the next chapter takes a deeper look of the psychological understanding of human learning and memory.

There is a lot of research in this area. Presenting all aspects and theories of learning and memory would be beyond the scope of this thesis, therefore, only basic theories that have or may have implications for the understanding of how learnability could be measured, will be introduced. This chapter tries to answer, amongst other things, the following central questions:

- How do humans learn?
- What happens when skill is growing?

2.2.1 Defining Learning

In our daily life the term *learning* is used with the most matter of course in a vast variety of topics, such as learning as a child how to speak, learning how to interact with the environment, riding a bicycle and learning chemistry. Regarding this variety of usage, it is not surprising that *learning* is a huge field in psychological research as well as in educational science. Therefore, no generally valid definition could be found. There are various of definitions existing, written from different point of views. In the following some

definitions are cited, so the reader can get a better understanding what exactly is meant by the term *learning*.

> "[L]earning [...] [is] the process by which changes in behavior arise as a result of experience interacting with the world" [24].

This definition emphasizes on the goal of learning: a change in behaviour. The process of learning itself, however, is not explained, besides the mentioning of the term *experience*.

This definition is criticized by [25] being to simple as not every experience will necessarily result in a change of behaviour, which [25] would suggest as learning. According to [25], experience that arises by storaging information in the brain is the result of learning. Whether this experience will lead to a change in behaviour does not matter. Likewise, [26] refers to a definition where learning is defined as a relatively permanent change in behavioural disposition, and not necessarily in behaviour.

The following definition describes learning from a more insight view:

> "Acquiring knowledge and skills and having them readily available from memory so you can make sense of future problems and opportunities" [27].

An important aspect of this definition is the mentioning of what someone can learn: knowledge and skills. Furthermore, according to this definition, learning is the process of the acquisition of knowledge and skills. But knowledge and skill can only be considered as learned, if there is the possibility of retrieving this knowledge and skill form memory in order to use it for further problems and opportunities. Therefore, memory seems to play an important role in terms of learning.

> "Learning and memory are intimately, perhaps inextricably, intertwined. The term *learning* emphasizes the acquisition of information, whereas the term *memory* emphasizes its retention, but both are facets of a single system for storing information about our experiences. You cannot remember an experience unless you first create a record of it (learning), and you cannot learn from this experience unless you retain this record (memory)" [25].

This definition indicates how closely interdependent learning and memory are. Learning includes memory and "memory depends on learning" [25]. Therefore, the next chapter focuses on human memory.

2.2.2 Human Memory

For a better understanding on how learning works, the comprehension of human memory is essential. Thus, a short overview of functionality of memory is provided. As a summary of all aspects of the human memory would be clearly beyond the scope of this thesis, only essential theories that are important for answering the central question of this thesis, how to measure learnability, are presented.

In general, learning and memory include the following three stages: encoding, storage and retrieval. The first stage, encoding, occurs during the presentation of information and is responsible for the transfer of information, which can be visual, auditory, semantic, a taste or smell, into a code that can be stored in memory. The result of the encoding stage is the second stage: storage in memory system (the brain). The third stage, retrieval, describes the process of recovering stored information on demand [28].

Many theories exist trying to explain the functionality and structure of memory. However, most psychologists share the opinion that the memory system can be discriminate in (at least) short-term memory and long-term memory. Both are types of memory, which differs in capacity and how long information can be stored [28].

2.2.2.1 Multi-Store Model

The probably best-known model is the multi-store model, presented in [29], as it had an enormous influence on psychology [10]. Nowadays, only few researchers still accept this model in detail, nevertheless, the basic idea of the distinction of memory in different components as described below is still widely hypothesized and its concept is the basis for some modern theories [30]. Therefore, the basic concepts are described in the following. Additionally, Figure 2.3 presents the model visually.

According to the model, human memory splits into three structural components: the sensory register, the short-term store and the long-term store. First of all, incoming sensor information enters the sensory register, which is characterized through the model by a high capacity but very low duration keeping. The sensory register can be subdivided in components for the different senses. Only few information, those who get attention, get transferred to the next component of memory, the short-term store. This selective function protects humans from stimulus satiation. Other information is lost, besides few information that can be kept as long as desired through a process called *rehearsal*. In simple terms it means the repetition of information over and over again [10].

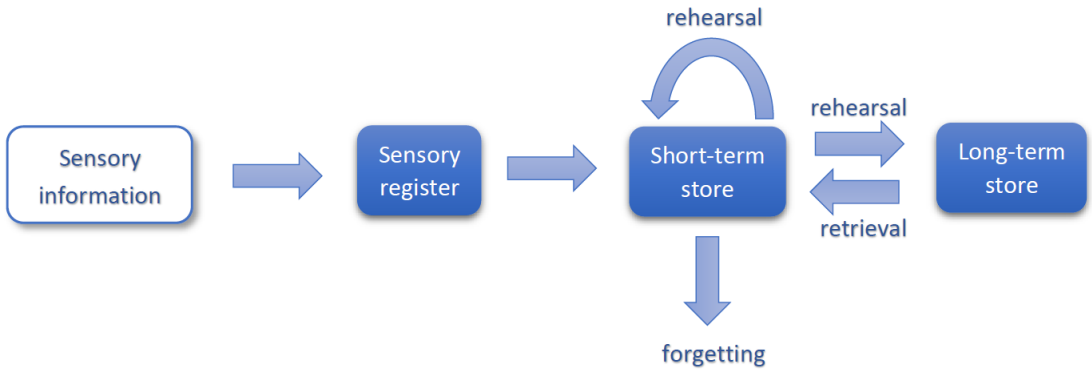


Figure 2.3: Multi-store model (adapted from [28, 30])

The short-term store gets selected input both from sensory register and from long-term store, thus new information can be compared to existing ones from long-term store. This enables a meaningful structure of the new incoming information. Therefore, the short-term store is described as the subject's working memory. It has a limited capacity and keeping duration of information. The duration is approximately between 15 and 30 seconds [10, 30]. Furthermore, the capacity is limited to approximately seven units, so-called chunks [31]. Through this chunking, individual information units can be combined in greater units of meaning and, therefore, the capacity of short-term store can be increased [10]. For instance, four single numbers can be combined to one date. More detailed information on chunking is provided in Chapter 2.2.3.3.

Finally, information can be transferred to the long-term store, which is characterized through a fairly endless capacity and an unlimited duration keeping. However, from time to time, humans forget information. Actually the information is still in long-term store, but the subject has a lack of access facilities [10]. The long-term store can be considered as a huge library with books instead of information. The book is still somewhere in a shelve, but was probably not arranged systematically, resulting in no incident to regain the information, or the book was not used for quite a while. Therefore, a meaningful structure and integration of new information is essential for knowledge retrieving [26]. In this model learning means the retention of processed information in long-term store [10].

The multi-store model has its strength, like the separation of memory in two systems, the short-term and long-term memory, with different capacity and keeping duration of information. There is evidence that these assumptions are correct [28]. Nevertheless, the multi-store model is criticized for being too simple about the structure of short-term and long-term memory. [29] assumed one single system for each, but as we see in the following chapters, other theories hypothesize several stores for short-term and long-term memory [28]. Furthermore, the multi-store model is criticized for the assumption that information get transferred from short-term to long-term store by rehearsal as in daily life people store many new informations without spending much time on active rehearsal [28].

Therefore, some alternative or complementary theories are presented in the following chapters.

2.2.2.2 Levels-Of-Processing Theory

On crucial disagreement of different theories is the assumption of how information transfers from short-term to long-term store. Whereas the multi-store model [29] assume that the probability of getting information transferred to long-term store increases with the amount of rehearsal, [32] assume that the depth of processing is crucial. According to them, rehearsal does not or only poorly improves memory, as long as the information is not repeated in a deep meaningful way [30] – independent of how long it is repeated [10].

2.2.2.3 Working Memory Model

Two decades after the publication of the multi-store model, [33] proposed a new theory of working memory as the short-term store of the multi-store model is far too simple. [33] assumes that the working memory consists of two independent systems for auditory information (phonological loop) and visual information (visuospatial sketchpad) as well as a central executive controlling them [10].

2.2.2.4 Theories of Long-Term Memory

The multi-store model [29] hypothesize only one single store for long-term memory. But considering the diversity of information that need to be stored, several researchers assume multiple stores for long-term memory [28].

[34] assumes two types of knowledge, declarative and procedural knowledge, which are interacting with each other. The declarative knowledge, corresponds to factual knowledge, like *Berlin is the capital of Germany*. A characteristic of declarative knowledge is that it is consciously accessible. [34] describes it as “things that we are aware we know and can usually describe to others”. Furthermore the knowledge is represented in chunks [34].

However, procedural knowledge is organized in so-called production rules. One example for a production rule is if you want to turn right with your vehicle, you must signal your intention. Another example is if you want to add two numbers you must first of all add the last digit of each number and then the previous digit of each number including the calculating transfer and so on until you have the sum of both numbers. In contrast to declarative knowledge, a person is not aware of its procedural knowledge. He or she is able to do something like riding a bicycle but can not verbalize how to do it [34]. Therefore, some theorists refer to declarative knowledge as explicit memory, whereas procedural knowledge is called implicit memory [24].

[35, 36] further subdivide declarative knowledge in episodic and semantic memory. Episodic memory covers things that someone remember, whereas things someone knows belongs to semantic memory. Therefore, autobiographical events, such as how

the day of your graduation ceremony was, what you were wearing on that day and how you were feeling, is episodic memory according to [35, 36]. Information about the context of the event is also included: where and when the ceremony was held and, therefore, where and when the event was stored in memory. Furthermore episodic memory is characterized by an acquisition in a single exposure, the event itself. Unlike episodic memory, semantic memory composes of things we know, such as facts like the president's name of the United States. The memory is not necessarily attached to a context: Someone knows the president's name, but has no clue where he knows it from or since when he knows it [24].

Figure 2.4 visualize this division of long-term memory into different types of knowledge.

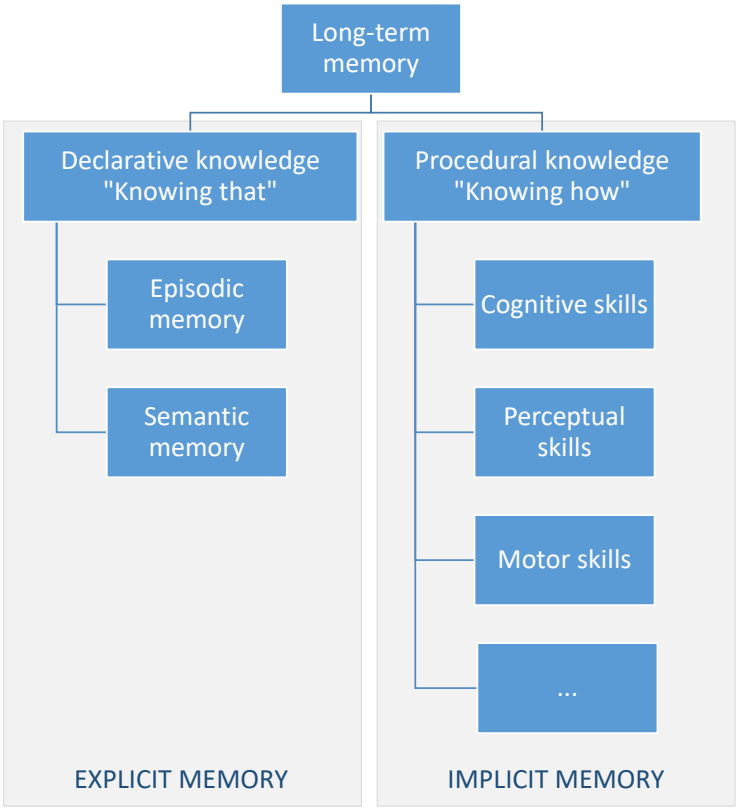


Figure 2.4: Hypothesized structure of long-term memory (adapted from [28])

Due to research on brain damaged patients (e.g., [37, 38]), it seems to be widely accepted that long-term memory is differentiated between declarative (with episodic and semantic memory) and procedural knowledge. However, recently an increasing number of theorist argue that the distinction between declarative and procedural knowledge is oversimplified [39]. One reason for this assumption is that there are tasks intend to address one type of knowledge (declarative or procedural), but in reality also the other type of knowledge gets involved. For example, in order to address the declarative knowledge, people get confronted with hints expected to recall their personal memory corresponding with the given hint. As the person is supposed to actively recall her or his memory, it is seen as a memory task testing the declarative respectively explicit memory. However, most of the memories produced are not explicit, but rather spontaneously and unintended [39].

2.2.3 Expertise

Earlier, a short overview regarding some basic theories on how the human memory works were provided. The reader should have by now some idea of the hypothesized functionality and structure of memory, such as how new memory is stored and of which components memory might be composed of.

As discussed in Chapter 2.2.2, theorist differentiate between skills (procedural knowledge) and knowledge (declarative knowledge). In the context of learning, skill and knowledge are inspected individually, since the acquisition of skill and knowledge also differs. Since good learnability is aimed at proficient system usage, the focus is on procedural knowledge. In the following, therefore, the main focus is on skill acquisition. Mainly the following question will be answered: *What happens when skill is growing?*

2.2.3.1 Stages of Skill Acquisition

According to [30], skill acquisition can be divided into three phases: cognitive, associative and autonomous stage.

In the first phase, the cognitive stage, a declarative encoding of the skill is developed, which consists of facts significant to the skill. During the first performance of the skill, learners normally rehearse theses facts. For instance, when learning how to use the gear lever in a car, first of all the location of the individual gears is memorized [30].

In the second phase, called associative stage, a procedure for executing the skill is produced. The learner "relays less on actively recalled memories", as he begins to use stereotyped actions. Furthermore, mistakes in initial understanding are discovered and removed by degrees [30].

With the last phase, the autonomous stage, the procedure becomes increasingly automated and fast. Also, fewer processing resources are required. Therefore, resources can also be spent on other tasks. The driver could be engaged in a conversation during driving even with no memory for the traffic he has driven through [30]. By this time "it may be impossible to verbalize in any detail the specific movements being performed, and performance may have become much less dependent on verbalizable memories for events and facts” [24].

In summary, "[t]he degree on which participants rely on declarative versus procedural knowledge changes dramatically as expertise develops". This "process by which people switch from explicit use of declarative knowledge [over] to direct application of procedural knowledge, which enables them do things such as riding a bike without thinking about it", is called proceduralization [30].

2.2.3.2 Learning Curves

The previous chapter explained why the performance of skill becomes more efficient and faster as it develops. Surprisingly, the amount of time required to conduct a skill decreases in a regular and predictable manner, independent of the skill [24, 40]. Thus, this relationship of practice and performance can be described mathematically. However, there is a dispute about the best function to describe it.

Widely accepted is the so-called power law of practice introduced by [41], which describes the relationship between response time and number of practice trials in a power function

[24, 30, 40]. [41] compared the power with an exponential and a hyperbolic function. However, their results pointed towards the power function. Likewise, other theorists reject the exponential function in favour of the power law of practice, e.g. [40]. A detailed discussion about the cognitive causes pointing towards a power function is given by [40].

However, other researchers are pleading for an exponential function (e.g., [42]), a sigmoid curve (e.g., [43]) or a mixture of power and exponential function [42].

Power Law of Practice

As already mentioned the power law of practice, which was introduced by [41], is widely accepted and quite a *gold standard* [42]. Generally, the power law of practice describes the relationship between performance and amount of practice, whereby performance can be measured by any variable that decrease with practice, such as response time, execution time or amount of errors [41]. However, [41] focused their research primarily on time measurements.

Typically, the power the law of practice refers to skill acquisition including cognitive as well as perceptual-motor skills, but also knowledge acquisition can be described with a power function. Therefore, sometimes it is also referred to as the power law of learning [24, 30].

The power law of practice can be mathematically described as followed (Equation 2.1) with T as the performance time, P is the amount of practice, a the speed on the first trial and b the slope of the function [30, 40]. The amount of practice, P, is typically measured in trails, which can be one execution of a task [41].

$$T = aP^{-b} \tag{2.1}$$

Assuming this function, there would be no limit in performance. After enough practice, the task could be executed in arbitrarily small time. In reality, however, there are many situations where the performance speed is unable to fall below a certain level. In addition, there is another issue this function is oversimplifying: It assumes that the first trial measured is the beginning of learning. Due to this two assumptions the power

function was further developed to Equation 2.2 to observe prior learning as well as to introduce a lower limit of performance speed [40].

$$T = c + a(P + d)^{-b} \tag{2.2}$$

c is the asymptotic level for performance speed and *d* is the estimated amount of practice trails that occurred before the first measured trail [40].

Another way of analysing the performance time in relation to the amount of practice is to transform the power law in a linear function by using log-log transformation [30, 44].

$$lnT = ln(a) - b \cdot lnP \tag{2.3}$$

Visualising the measured data, a linear function in log-log coordinates should be seen if the relationship of time and practice in normal coordinates fit to the power function (see Figure 2.5) [30].

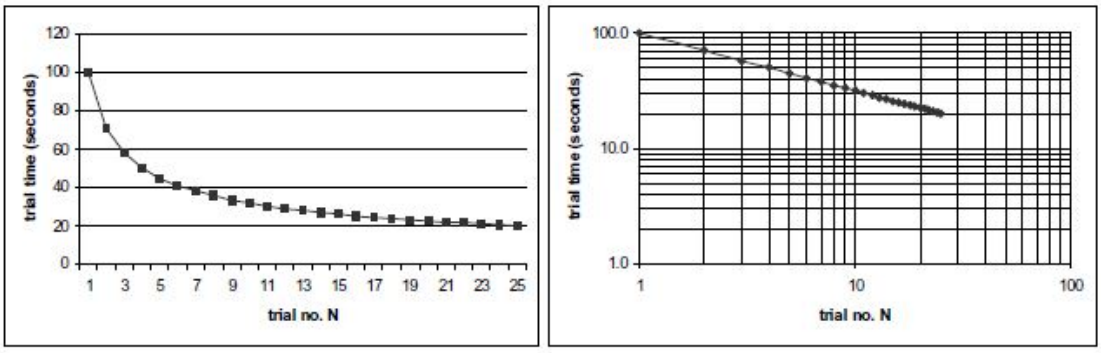


Figure 2.5: Visualisation of measured data. In the left plot the typical curve of the power be seen. In the right plot the data is presented in log-log coordinates [44]

As already mentioned, the power law of practice could be observed in many cognitive tasks. Also in HCI, researchers observed such a relationship between performance and practice [40, 45, 46]. For instance, [45] analysed mean time on task of 12 participants over 20 trials using an e-commerce data management tool to update product information. Learning curves were analysed for two different update tasks. Just one learning curve

fits to a power function. Nevertheless, they find the power law of learning useful in helping to analyse tool efficiency.

Research on evidence for the power law of practice has conducted mainly on average data. For instance, [41], who plead for the power law, used data averaged over subjects, conditions, or practice blocks for all tasks they had examined, except for one [42]. However, it was assumed that the power law of practice also holds for individual data [41, 42], even though it is known that the curve of individual data composing average curves do not need to be the same as the curve of the average data [42]. This is one main reason for emerging discussion about the correctness of the power law.

Exponential Function

[42] is one of the persons that criticize the evidence for a power law being faulty due to the fact that it is based in averaged data. He analysed datasets of 475 subjects in 24 experiments and came to the result that an exponential function fits better in all unaveraged data sets. The exponential function is presented in Equation 2.4. Instead of *b*, the slope of the function is called α, besides that the naming of the parameters is analogous to the power function.

$$T = c + ae^{-\alpha \cdot P} \tag{2.4}$$

Learning Curves by Nielsen

The previous learning curves attempt to provide mathematical functions that fit to learning process in most of cognitive skills, including human-computer-interfaces. [8], in turn, concentrates only on learning curves for human-computer-interfaces. He provides only a general shape and not an exact mathematical function. Furthermore, he presents two learning curves depending on the context of use of the system.

[8] differentiate between systems focusing on novice users and systems focusing on expert users. Systems with focus on novice users have usually a high demand on ease of learning. Therefore, such systems have a strong increase in usage proficiency and

efficiency for the first part of the learning curve. The user can reach within short time a suitable level or proficiency [8]. The resulting shape of the curve, which is presented in Figure 2.6, reminds of an exponential or power function.

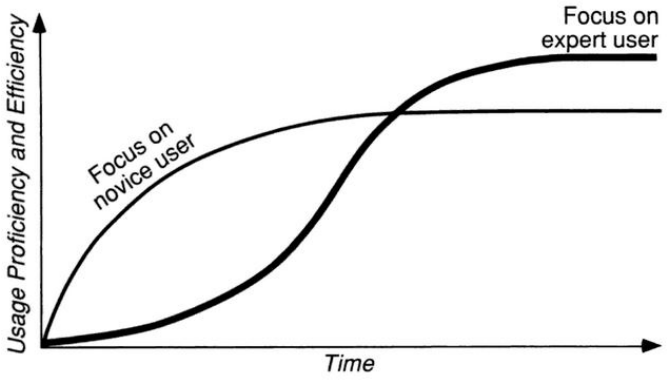


Figure 2.6: Learning curves by Nielsen [8]

Systems developed for expert users have usually a high demand on efficiency. Thus, the system may be hard to learn at the beginning [8]. An example is the usage of shortcuts, which can be very helpful for experts to get work done faster, but are difficult to memorize at first. Regarding the learning curve of such a system, the user makes only small progress at first, but then proficiency is rapidly increasing and outstrips the possible level of proficiency and efficiency of systems for novice users [8]. The learning curve has the shape of an sigmoid curve, which is also presented in Figure 2.6.

[8] differentiated between two extreme hypothetical systems, one only focussing on novice users and one only on experts. In practice, it is seldom necessary to decide whether a system is either easy to learn or allows reaching high efficiency. Often it is possible to develop a system that is easy to learn in the beginning and yet achieves a high level of proficiency. This can be reached by proving an interaction style easy to use at the beginning and than give the user the possibility to switch to a more efficient interaction style, for instance. The shape of the learning curve would be the same as the learning curve for systems only for novice users at the first part, but it raises up at the level of a system for expert users [8].

Summary

Independently of how a learning curve can mathematically described (for example as a power or an exponential function), it has the following characteristics in common:

- If something is easy to learn, the learning curve has a steep decrease in performance time and, therefore, a steep increase in proficiency at first.
- Then the decrease respectively increase lessens and slowly gets closer to a level of performance time which can not be undercut.
- The learning curve follows a predictable pattern.
- The exact curve varies for different tasks and subjects, but the tendency stays the same.

2.2.3.3 Impact of Expertise on Chunking

Chapter 2.2.2 mentioned the concept of chunking, which sorts information to greater units, so-called chunks, and hold them in short-term memory. In this type of memory only a certain amount of chunks, around seven chunks, can be hold [26]. As the research from, for example [47, 48, 49], shows, experts form larger as well as more complex chunks than novices [49]. This means that more information can be stored within one chunk and, therefore, much more information can be hold in memory. [20, 50] showed in their studies with 24 and 28 participants that this phenomenon can also be observed in the context of HCI. Furthermore, [20] observed that the chunks size gets more regular.

2.2.3.4 Mental Models

Another aspect that is important when considering learning progress, especially with regard to HCI, are *mental models* [1, 51]. With respect to HCI, a mental model is defined by [51] as:

> "Knowledge that the user has about how a system works, its component parts, the processes, their interrelations, and how one component influences another".

In general, the purpose of mental models is to help people to learn and understand complex situations [51]. Note that mental models are based on user's believes, built on previous experience, knowledge and current observations, rather than on facts. They are incomplete and change over time, for example, if new experience and knowledge are gained [52, 53]. Hence, when expertise is growing, the mental model changes. This thesis is supported by several researchers, who observed in the field of HCI a discrepancy between mental models of novice and expert users, e.g. [51, 54]. The mental model of expert users were significantly closer to the intended model of the system [54].

3

Existing Methods for Measuring Learnability

The last chapter dealt with learnability and learning in general, whereas this chapter evaluates possibilities to measure learnability in human-computer interaction.

Although learnability is rarely explicit measured in practice [16], despite its widely recognized importance in research [9], several methods to measure learnability could be found that seem to be either appropriate or promising. This includes methods that have either proved to be valuable in an evaluation or are still under development, but appear promising enough to be worth mentioning.

3.1 Overview

In general, methods for usability can be subdivided into two categories: empirical and analytical methods. In empirical methods, a system is assessed by studying the actual users (respectively representatives of actual users), whereas analytical methods are performed without user involvement. Analytical methods are either conducted by experts, who put themselves in the position of a user, or are based on models [10].

A characteristic of evaluation methods is the time and purpose of their execution: One distinguishes between formative and summative evaluations. Formative evaluations are performed during the development process with the aim of detecting problems and correcting them afterwards. In contrast, summative evaluations pursue the goal to evaluate the overall quality, for example, to decide between two alternatives. Simply

said, summative evaluations try to answer *Which one is better?* and/or *How good is it?*, whereas formative evaluations try to answer the question *Why is it bad?* [10, 55].

The collected data can be either objective or subjective and quantitative or qualitative [55]:

- **Objective data**: Can be directly measured or observed.
- **Subjective data**: Opinions, usually expressed by the user. But also methods that strongly relay on the expertise of the evaluator produces subjective data [22].
- **Quantitative data**: Numerical data, such as scores.
- **Qualitative data**: Non-numerical data, such as lists of issues.

These two characteristics occur in every combination. For instance, survey data is subjective and quantitative, whereas *performance measurements* are objective and quantitative. Data, which is qualitative as well as objective, for example, is a record of sequence of steps taken by a user. Noticed feelings during an *observation* are subjective and qualitative.

Important to notice is that these characteristics are referring to the collected data and not to the method itself. For instance, a *questionnaire* results in subjective data, the method itself is usually highly objective [10].

Some researchers additionally distinguish between attitudinal and behavioural dimensions [56]. The purpose of attitudinal research is the measurement or understanding of the user's opinion, whereas the purpose of the second one focusses on the behaviour of the user [57].

These categorization seems also be applicable for learnability, since learnability is widely seen as an aspect of usability and found methods can be classified according to these characteristics.

Before discussing the methods to assess learnability in detail, Figure 3.1 gives an overview of all the methods that will be presented in this thesis. There is no uniformly accepted classification of evaluation methods in HCI, so an own categorization has been made based on [1, 10, 58].

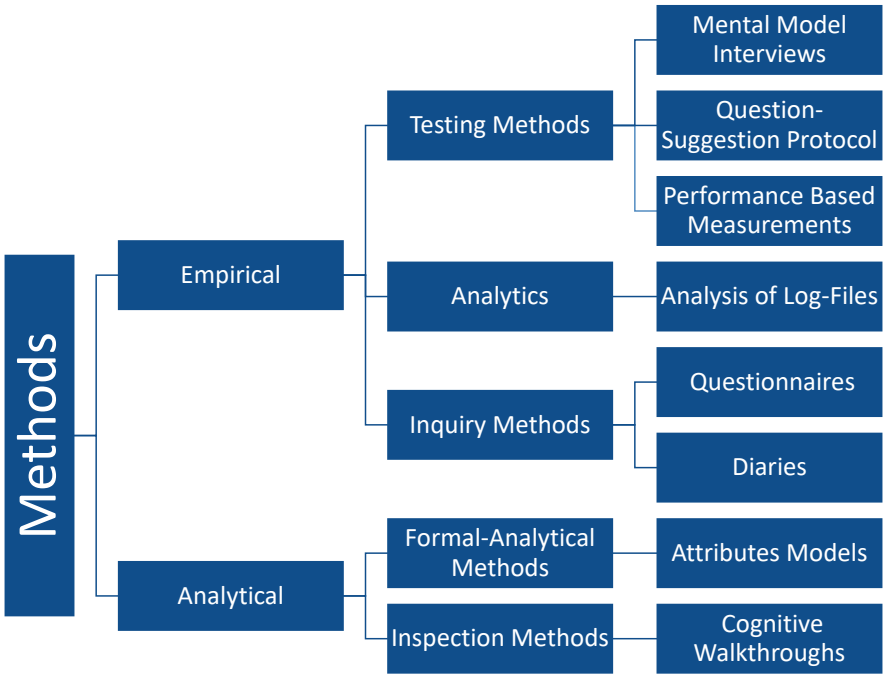


Figure 3.1: Overview of methods to assess learnability (general classification inspired by [1, 10, 58])

In this classification empirical methods are further differentiated in:

- Testing Methods: These refer to usability testing, which is described by [59] with "three key components: representative participants, representative tasks, and representative environments, with participants' activities monitored by one or more observers" [59]. Typically these methods are conducted in a usability laboratory, but also remote-usability-tests, respectively remote-learnability-tests, and field studies are possible [1].

- Analytics: User behaviour is analysed via tracking, such as logging of events or web-analytics-data [1]. The advantage of these methods is that there is no need to involve a moderator or observer. Participants can freely use the system in their natural environment. Note that these methods can also be used within testing methods [1].

- Inquiry Methods: The focus of these methods is on getting an overall subjective impression (such as preferences and opinions) of the user [58]. *Diaries*, one inquiry method, are normally exclusively performed within longitudinal studies during natural system usage, whereas *questionnaires* are often conducted within a testing method.

Analytical methods are further subdivided into:

- Formal-Analytical Methods: User interfaces are analysed and described using established formalisms. The process takes place without the involvement of users or user representatives [10].

- Inspection Methods: Experts go through the application identifying learnability issues based on either tasks or principles [1]. The application does not need to be implemented yet. However, it rely solely on the judgement of the evaluator [58].

During the research, the focus was set on methods that are exclusively tailored to learnability. Nearly all methods found particularly for learnability are presented in this thesis. One exception, for example, is the research by [60], who hypothesised that learnability can be assessed by observing brainwave patterns with electroencephalography. Although a dependency could be observed, study results did not fully comply with their hypothesis. Therefore, further research is necessary in this field.

In addition, since most researchers agree that learnability is an aspect of usability, typical usability methods were examined to evaluate if they are particularly well-suited for assessing learnability. The methods that appear suitable are presented in the following. It is important to note that other methods, which are not mentioned, may also be appropriate, such as *observations* or *thinking-aloud protocols*. However, during research, other methods have emerged that seem more appropriate. These methods are *questionnaires*, which include parts solely for learnability, and *diaries* and *cognitive walkthroughs*, which are explicit recommended for assessing learnability [10, 61, 62].

In the following, methods are presented in detail and discussed afterwards. If a very specific method is presented, I will also refer to enhancement and related work.

3.2 Testing Methods

This chapter presents testing methods suitable for learnability measurement.

3.2.1 Mental Model Interviews

One option to evaluate learnability with user involvement in very early stages of software development, when not even a prototype exists, are *mental model interviews*. As discussed in Chapter 2.2.3.4, mental models of expert users are significantly closer to the intended model of the system than mental models of novices. According to [54], systematic deviations between the user's mental model and the system model can indicate usability issues. Especially the comparison of novice and system models can highlight potential learnability difficulties [54].

Therefore, *mental model interviews* seem valuable as they can be used to uncover potential learnability issues very early in design when changes can be easily implemented. However, only one publication [63] could be found using *mental model interviews* for assessing learnability without evaluating the method itself. Nevertheless, *mental model interviews* are applied in other areas such as usability (e.g., [64]) and play a relatively large role in fundamental research in HCI. Therefore, the approach by [63] is presented below.

Presentation of Method

Mental model interviews are generally conducted with the purpose to gain informations about the user's mental model. [63] used these interviews to get an impression of the user's metal model even before users interact with the system. In a 45-minute interview (per participant), the participant was shown an interface. For individual elements, such as icons, the participant was asked questions, such as *"Which icons seem familiar to you? What do you think the other icons represent?"* and *"What do you expect the items that you see to be?"*. Afterwards, participants were asked how they would perform certain basic tasks. As [63] provided a clickable prototype, participants were allowed

to try the suggestion they had proposed. In terms of failure, the correct operation was shown.

[63] audio recorded the interview to analyse comments in detail afterwards. Special attention was paid to situations where the participant's mental model did not map to the system structure.

[63] judges *mental model interviews* a good opportunity to identify either where system functions should be changed towards users' expectations or where functionality should be more obvious to the users.

Related Work and Enhancement

As already mentioned, this publication [63] was the only one found that explicit uses the users' mental models in the context of learnability. In addition, no other publication of the author could be found that further evaluated or enhanced the approach.

However, investigating the users' mental models is a common method in HCI. There are different purposes and approaches on doing so. In addition to a system evaluation, mental models can generally be used to analyse opinions and desires of users to refine personas and scenarios, or to gain a better understanding of customers, for example, in sales and customer service. Besides interview techniques, mental models may also be gained, for example, through *diaries* or *field observations*. However, the deepest understanding of user's mental model is usually obtained in interviews [65].

One example is presented in [64]. The authors conducted interviews to gain qualitative insights into how novice and expert users perceive and respond to different computer security warnings. Based on the resulting mental models, the authors provide general advise on how to communicate security information.

3.2.2 Question-Suggestion Protocol

The *question-suggestion protocol* is a method specially designed to analyse learnability. It was developed by [9] after reviewing existing approaches. It is based on the *question-asking protocol* [66], which was presented in 1986 as an alternative to the *thinking-aloud*

protocol. The main idea of the *question-asking protocol* [66] is that instead of letting the participant constantly talk about his or her thinking, while using the system, a tutor sits next to the participant whom the participant can ask if something is unclear. Only concrete questions should be ask and not vague ones like *What should I do next?*. The tutor should not bring the participant to ask any specific questions. Furthermore, when answering a question, the tutor "should not give the participant any more information than what is really needed to solve the current problem". [66] argues that asking questions is far more natural to the participants than constantly talking about thinkings as in the *thinking-aloud protocol*.

The *question-asking* as well as the *thinking-aloud protocol* seem to be especially appropriate to evaluate learnability as both protocols provide insights to the cognitive processes of the participants [9]. However, these two methods were designed to evaluate usability and, unfortunately, it stays unclear whether these approaches are proper to evaluate learnability. [9] agreed with this assessment. Nevertheless, [9] considered the *question-asking protocol* very promising and underestimated. Therefore, he designed the *question-suggestion protocol* as an adaptation of the *question-asking protocol* allowing the evaluation of initial as well as extended learning.

Presentation of Method

The *question-suggestion protocol* is similar to the *question-asking protocol*, but augmented with the possibility of the tutor to suggest something to the participant [9].

As a remember: the *question-asking protocol* does not allow the tutor to give any more information than necessary to solve the current problem. This rule prevents the participant to be able to solve other task without asking questions, which he otherwise could not have. Next to inviting the participant to explain his or her behaviour when the behaviour of the participant appears illogical, the tutor is only allowed to take the initiative if the participant should be led "to use a [specific] [...] function that would otherwise remain unknown to him/her" [66].

[9] criticized this approach as it only focusses on initial learning. [9] thought that "[t]o truly understand extended learnability, we must also understand what causes users to not just acquire new abilities, but what causes them to improve their usage behaviors by finding more efficient strategies". Therefore, at the *question-suggestion protocol* the tutor is encouraged to propose better strategies of usage to the participant. [9] compared this situation to a scenario where a colleague is sitting next to the participant and notices that a certain behaviour can be improved. This scenario has proven to be a conventional way for users to learn. Due to this "suggestion", it is possible to evaluate initial as well es extended learning as it helps participants to make progress and, therefore, learnability issues that emerge later at the learning curve can be detected. In addition, the suggestion allow the participant to further edit a task, which, in turn, reveal a greater number of learnability issues [9].

As [9] found no studies comparing the *question-asking protocol* or a similar approach with the *thinking-aloud protocol*, he conducted a study with ten participants using AutoCAD (a software for technical drawings and 3D constructions) comparing his *question-suggestion protocol* with the *thinking-aloud protocol*. As AutoCAD is a quite complex software, the participants had domain knowledge as well as experience between 2 months and 5 years in AutoCAD. In addition to the tutor, which was an AutoCAD expert, there was also an experimenter, who was a HCI expert and ensured that the rules of the protocols were met. The two protocols were used as a within subject variable, counterbalanced in their order. Each participant had to perform four tasks, two tasks per protocol [9].

The study concentrated mainly on the number of detected learnability issues. As hypothesized, with the *question-suggestion protocol* more learnability issues could be found than with *thinking-aloud*. For the *question-suggestion protocol* an average of 7.55 learnability issues were reported, whereas with *thinking-aloud* only 2.8 issues were reported. Besides the significant effect for the protocol, a significant effect for the level of experience could be observed. But no significant dependence between protocol and experience could be observed. The results are presented in Figure 3.2 [9].

Additionally, categories of learnability issues were analysed (see Figure 3.3). In the *location* category the *thinking-aloud protocol* found a higher proportion of issues compared

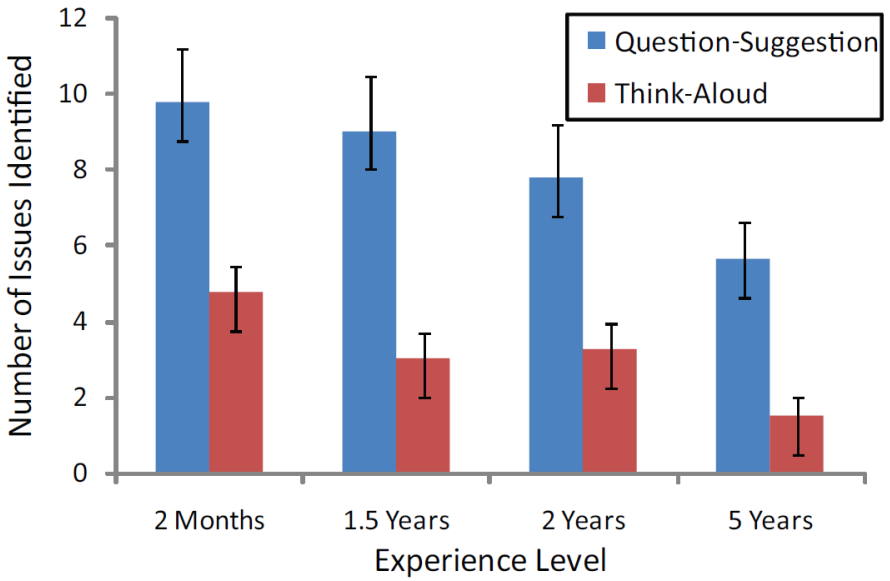


Figure 3.2: Comparison of the number of learnability issues averaged over all tasks identified by the *question-suggestion* and the *thinking-aloud protocol*. Results are grouped by the level of experience of the participants [9]

to the *question-suggestion protocol*. In all the other categories, the *question-suggestion protocol* founded a higher proportion of issues. Issues concerning transition were found exclusively by *question-suggestion protocol*.

Even though the *question-suggestion protocol* showed clear benefits, this protocol is not intended to replace the *thinking-aloud protocol*, as the latter have its own strength, such as the possibility to observe "how well users can recover from errors, and how long it takes them to figure things out on their own" [9].

Related Work and Enhancement

The paper [9] was cited over 200 times in 2018 according to Google Scholar. However, only 15 of these 200 papers pay attention to the *question-suggestion protocol*. None of them, however, evaluated or enhanced the *question-suggestion protocol*. Some of the papers, such as [67], applied the protocol in practice.

Category	Example Learnability Issue Observed
Task Flow	Did not know how to set up a text style. Did not know how to evenly space 6 objects.
Awareness	Was not aware of the *divide* command. Was not aware of the *mirror* command.
Locating	Could not find *layer manager* icon. Could not find *leader* toolbar.
Understanding	Couldn't figure out how to use the *extend* command. Couldn't figure out how to use the *align* command.
Transition	Did not use *match properties* tool and instead created a new hatch. Didn't think to use *mirror* command, and instead manually created a section of a plan.

Figure 3.3: Categories of observed learnability issues [9]

3.2.3 Performance Based Measurements

As discussed in Chapter 2.1, several authors define learnability through reaching a certain level of efficiency. Besides the time required to reach that level, error-free usage is explicit mentioned. Therefore, it seems obvious to measure learnability via efficiency with performance metrics, such as execution time or errors made.

Indeed it is proposed to asses learnability via efficiency by researchers (e.g., [8]). Due to the simplicity of this method, [8] supposed that learnability is one of the easiest usability attributes to measure. In practice, this procedure is widespread (e.g., [9, 46, 68, 69]). The basis for analysing the measured performance data is often the power law of practice (discussed in Chapter 2.2.3.2).

First, approaches assessing learnability via performance metrics over all participants are presented. Afterwards an outstanding method is presented, as it only analyses the worst and the best performing participant.

3.2.3.1 Performance Measurement Based on Learning Curves

According to [70], nearly every performance metric over time can be used. Indeed a wide range of proposed metrics can be found in literature [9]. However, the best-known

metrics are aimed at "efficiency, such as time on task, errors, number of steps, or task[s] [successfully executed] [...] per minute" [70]. But note that there is a possible trade-off between speed and error made as systems that are extremely low in the likelihood of failure may suffer in performance speed [2].

After choosing a metric, the measurement interval must be specified, as the metric should be observed over time. Ideally, it is based on the usage behaviour of the target users. However, it may be the case that the system is only used every few weeks, months or even years. A study that takes so long is usually not practical. [70] suggests the following options:

1. Several trials within one session
2. Several trials within one session but with pauses in between
3. Several trials "over multiple sessions, with at least one day in between"

[70] defines the term *trial* as each instance of capturing data. Within a trial, the participant has to conduct one or several tasks. From the first option to the third one, effort for the conductor and participants increases. Likewise, the study gets more realistic as memory losses are taken into account. In General, [70] recommends at least three or four trials.

When visualizing the measured metric for each trial, a learning curve should be observable (e.g., Figure 3.4). The shape of the curve can be compared with the shape of an *ideal learning curve*, such as a power function. For instance, if the task is to solve a problem with a user interface, like ordering a product via an online store, deviations from the ideal curve may indicate issues with the interface [45]. Another way to use the knowledge about the power law of learning is the prediction of future performance. Having the data of the first trials, the unknown variables from Equation 2.1 a and b can be calculated and, therefore, future performance may be predicted [45]. This second opportunity is interesting if someone want to know when the user will reach maximum performance.

Although the idea of performance measurement is based on learning curves, practitioners are seldom trying to compare the outcome with a mathematical function, such as the power function. Instead, the curves of different systems, alternatives or study conditions

are compared. For example, [68] assessed the effect of different training conditions to learnability by measuring task completion time in two sessions with one week in between. The authors evaluated a deformable smartphone case that acts as an input device through bend gestures. The results are presented in Figure 3.4. Additionally, memorability was evaluated separately by comparing the performance of Trial 3 of Session 1 to Trial 1 of Session 2.

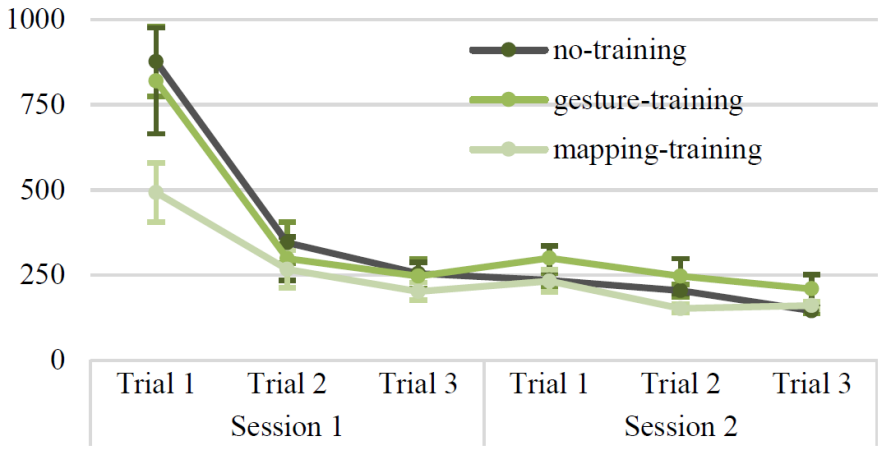


Figure 3.4: Completion time in seconds for different training conditions [68]

[69] conducted their study of learnability of a complex business application in three sessions on one day. According to [69], memorability is an aspect of learnability. Therefore, in order to take memory loss into account, participants had breaks together between the sessions outside the testing room where they were encouraged to talk about other subjects. Additionally, a "distraction task" was given before the final session [69]. They also measured task completion time as a performance indicator. Results were visualized per participant as well as per task. For analysing the learnability per task, [69] calculated the improvement from Round 1 to Round 2 in percentage, averaged over all participants. Thereby, it is clearly recognizable within which task learnability issues exist. The results for each task are visualized in Figure 3.5.

Another example is described in [46], where visit duration of websites are used as an indicator for learnability. One example of their results are presented in Figure 3.6.

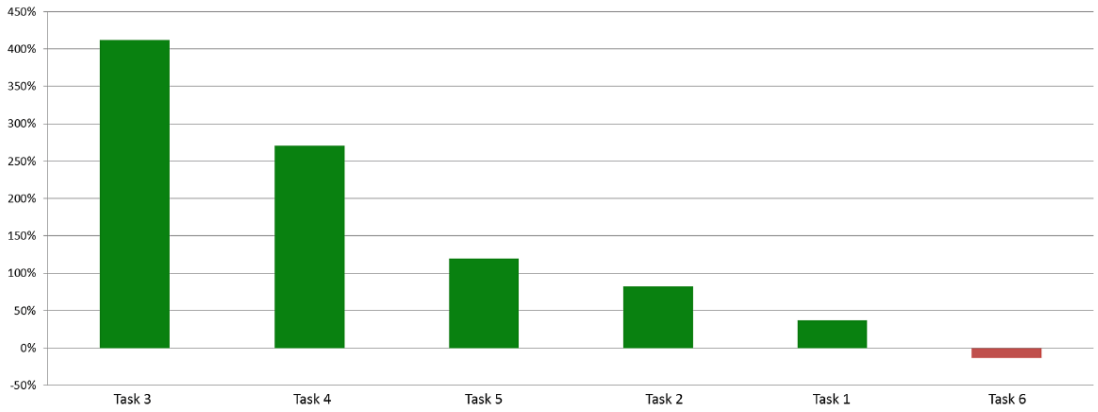


Figure 3.5: Percentage improvement in completion time for each task [69]

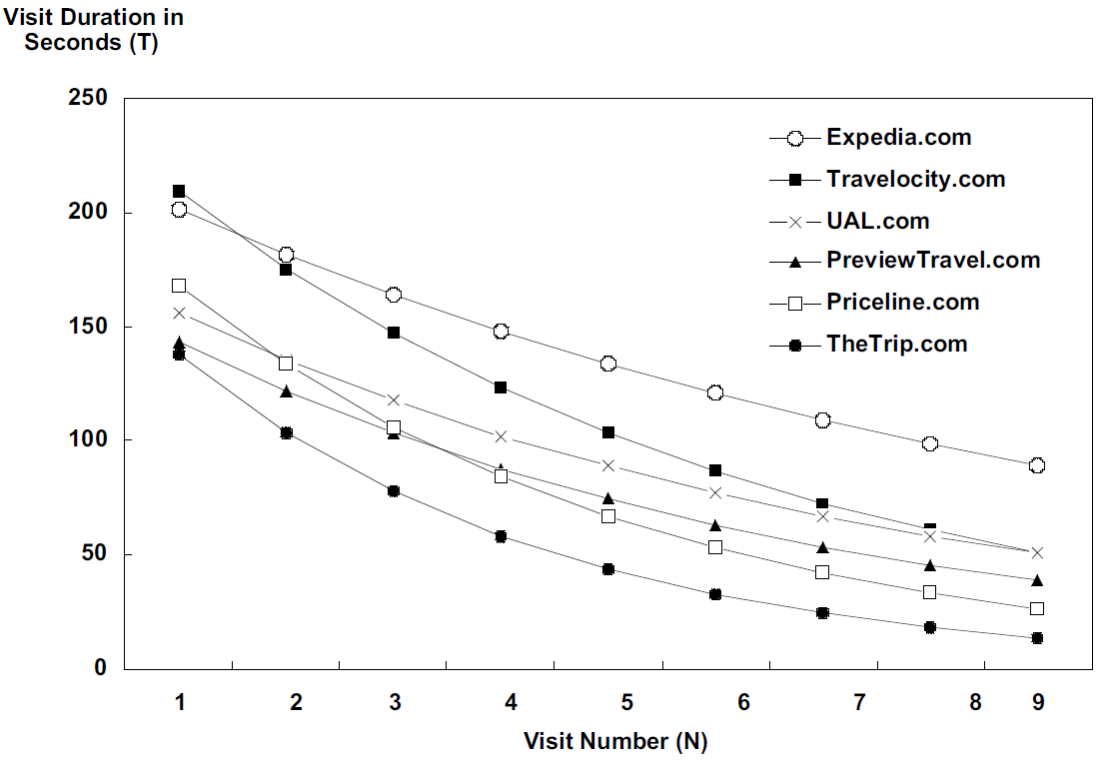


Figure 3.6: Visit duration observed over number of visits of various travel websites [46]

One outstanding approach is discussed in [7], where a score is calculated based on performance measurements for better comparable results. Overall learnability is defined by $\bar{c}$ as an average over all sessions (see Equation 3.1).

$$\bar{c} = \frac{1}{N} \sum_{j=1}^{N} c_j \tag{3.1}$$

c_j, in turn, is based on "the total number of tests within a session" (*n*), the efficiency for a task (e_i) and the completion time of the task (t_i) (see Equation 3.2).

$$c = \frac{n \sum_{i=1}^{n} e_i ln(t_i) - \sum_{i=1}^{n} e_i \sum_{i=1}^{n} ln(t_i)}{n \sum_{i=1}^{n} ln(t_i)^2 - (\sum_{i=1}^{n} ln(t_i))^2} \tag{3.2}$$

Note that the approach can only be applied if a standard learning curve can be observed. [7] assume a learning curve of novice users by Nielson [8] (see Chapter 2.2.3.2), which they formalized as a logarithmic function. The authors successfully validated their method within a study with 101 participants, which had to perform ten different tasks on an wrist watch that tracks sport activities.

3.2.3.2 Analysing Trials-to-Criterion by Means of Range Statistic

Another approach of quantifying learnability via performance measurements is from [71], using the amount of trials participants need to reach a predefined criterion. Two non-standard characteristics of their metric are the fast track evaluation of learnability and the preservation of the variability of different individuals [71].

Presentation of Method

To quantify learnability, [71] proposed to evaluate the number of trails needed to reach a defined criterion either during early practice or after a while when trying to re-achieve the criterion. Measurements during early practice evaluate initial learning whereas re-achievement focusses on the ability to retain information.

To estimate the degree of learnability, only the best and worst performing participant is analysed. Their performance is added and the inverse of the mid-range is calculated,

which is denoted with $\bar{i}$. This calculation is presented in Equation 3.3, with *B* as the number of trials-to-criterion of the best performing participant and *W* as the number of trials-to-criterion of the worst performing participant.

$$\bar{i} = 2/(B + W) \tag{3.3}$$

As already mentioned, [71] had the intention to preserve the variability of performance of the participants instead of restricting them by manipulation or control. Restriction of real-world variables is a common practice in standard hypothesis testing. In software evaluation, many variables would have to be taken into account, like the experience, motivation, intelligence and alertness of the participants. However, in real-life these factors significantly influence learning and performance. Therefore, [71] could not see the point in restricting these factors during evaluation. According to the authors, the goal should be that the software is "so good that the cognitive work (human-system integration, etc.) will be measurably superior despite the daunting variability of the world" [72]. Hence, a mid-range, in this case the inverse of the mid-range (cf., Equation 3.3), is used to preserve the performance span.

The result of the inverse mid-range $\bar{i}$ is an absolute value between 0 and 1. According to [71], this enables an evaluation of new technology without the need to compare it to a reference system, like a legacy system. However in this context, the interpretation is rather challenging as it can be seen as a conjoint measurement scale evaluating the appropriateness of the defined criterion as well as learnability. If $\bar{i}$ is pretty close to 1, the best as well as the worst performing participants reached the criterion within few trials. This indicates that "[e]ither the cognitive work is trivial or the criterion was set too low" [72]. The other way round, if $\bar{i}$ is close to 0, the cognitive work might be "very difficult or the criterion was set too high" [72]. Therefore, "the i scale can serve as a tool for fine-tuning the criterion, or guiding the selection of the learning trials cases (or problem tasks) of an appropriate degree of difficulty" [72].

If one assumes the criterion is adequate, $\bar{i}$ can be interpreted as a scale for learnability. A high value of $\bar{i}$ indicates a good learnability, whereas a low value may be an evidence for required improvement. The threshold of interpreting learnability as good or worse is

domain-specific. The same also applies to the interpretation of cognitive work and the criterion in the first place. One possible interpretation of $\bar{i}$ scale is provided by [71], which is mostly based on their own experience in laboratory. Their approach is presented in Table 3.1. Nevertheless the authors supposed that measurement the of $\bar{i}$ could be widely applied.

$\bar{i}$ Scale Ranges		Values of (B, W) $\bar{i}$	Desired Discrimination
↑ Range of Trivial Cognitive Work		(1,1) 1.00 (1,2) 0.66	$\bar{i}$ between 1.00 and 0.66 suggests that the cognitive work may be trivial or that the performance criterion needs to be raised.
↓ Range of Non-trivial Cognitive Work		(1,3) 0.50	Edge of the range. Criterion may still be set too low.
	↓ Range of (re)learnability	(1,4) 0.40 (2,3) 0.40 (2,4) 0.33 (2,5) 0.29 (1,6) 0.29 (3,5) 0.25 (3,6) 0.22 (4,6) 0.20	Fine discriminability is desired.
	↓ Range of Stretch	(2,9) 0.18 (3,8) 0.18 (5,7) 0.16 (4,9) 0.15 ↓	Finest discriminabilty is desired.

Table 3.1: Illustrative interpretation of the $\bar{i}$ scale by [71, 72]

Quite low values (cf., Table 3.1, values below 0.20) could have different causes. It indicates that "the cognitive work might be extremely difficult", a low learnability, a too high defined criterion or a combination of the latter. If the cognitive work is extremely difficult, like during aviation trainings on a flight simulator, where trainees need hours of practice before reaching a certain criterion, finest differentiation is desired as variations in second decimal place of $\bar{i}$ might be meaningful [72].

As far as it can be assumed from [71, 72] the method has not been evaluated within an experiment or study with participants. The authors emphasize that their solution is not

a closed-end one. "Rather, it is a first step or a prospectus". So far their research was rather of mathematical interest as their main focus is on providing a more suitable statistical analysis for non-Gaussian distributions in human-computer interaction evaluation software-supported cognitive work [71, 72].

Related Work and Enhancement

Several publications were found in which learnability was evaluated via trials-to-criterion. For example, [73] assessed learnability of two alternative designs of a control panel in aviation with a criterion of two consecutive accurate executions of a scenario. However, the measurement was only applied and not critically scrutinized. Additionally, no score, such as $\bar{i}$, was calculated.

No publication could be found applying the metric presented in [72] or evaluating it towards the measurement of learnability.

3.3 Analytics

Another type of methods to assess learnability are analytics. The term *analytics* is almost exclusively used in the context of web usage, estimating for example event flows to follow the users' navigation paths or metrics, such as page visits and download rates [1].

Likewise, it is possible to automatically estimate such data of non-web-based applications. With the help of log-files, events can be tracked during natural system usage. Therefore, such automated tracing in non-web-based applications are also classified to the analytics methods in this thesis.

3.3.1 Analysis of Log-files

During research two different methods of learnability assessment could be found that are explicit developed to analyse log-files. Thereby, participants can stay in their natural environment. There is no need for any laboratory studies or need for presence

of evaluators or instructors, such as in *observations*, where participants are in their natural environment, but getting observed, which may have an effect on the participant's behaviour [74].

3.3.1.1 Learnability Evaluation based on Chunk Detection

As discussed in Chapter 2.2.3, the size of each chunk increases and becomes more regular with expertise. Expertise, in turn, is a result of learning. Therefore, it seems reasonable to use chunk size and its variation as an indicator for learnability.

[20, 50] introduced a method to detect chunks for evaluating learnability in HCI.

Presentation of Method

[20, 50] developed a chunking detection algorithm, which can be used to measure learnability. In an experiment with 24 participants they validated "the use of chunk size as an indicator of learnability". The participants were divided into two groups, one with assisted learning and the other one with limited tutoring. To control learning strategies, all participants received at least basic tutoring of problem solving strategies. The experiment was conducted over twelve sessions with nine tasks to solve per session [20].

Previously, the algorithm itself was validated in an experiment with 28 participants, resulting in a significant number of detected chunks [50].

The algorithm for chunking detection is based on user actions. While using the system to be evaluated, all user actions need to get logged (automatically) and then get analysed by the algorithm.

First, in order to explain the functioning of the algorithm, the users' behaviour while interacting with a computer interface is demonstrated. Given a huge task that has to be conducted, like ordering a product, the user typically subdivides it into smaller, cognitively manageable, self-contained tasks. To solve these smaller tasks, users typically act according to a cycle with two phases:

- **Acquisition phase**: During this phase the user thinks of the goal of this task and how to reach it. He derives a strategy to reach that goal and is mentally planning how to execute this strategy. Typically, there is no physically interaction with the system in this phase.

- **Execution phase**: Then the user executes the plan by physically interacting with the system. A burst of activity can be observed. As soon as the goal is reached (or failed to reach it), the cycle is repeated.

During the execution phase, the plan is stored in short-term memory and can be referred to as a chunk. Regarding an interaction log, these cycles can be recognized. There are typically sequences of activity, occurring in execution phases, interrupted by pauses, corresponding to acquisition phase. In Figure 3.7 user actions are symbolized by vertical stripes. Groups of actions, separated by a quite long pause, can be classified as chunks (see Figure 3.7, second timeline).

Figure 3.7: Classification of user actions to chunks [20]

The algorithm described in [20, 50] detect theses chunk boundaries and count the size of each chunk. For identifying these boundaries [20, 50] used a variation of the keystroke-level model [75], but emphasized that also other predictive models may be applicable. In detail the algorithm proceeds following steps: For each user event estimate the predicted execution time with the predictive model of the previous user event. Then compare this predicted execution time with the actual execution time of this event. If the actual time between two events is clearly longer than estimated, a chunk boundary is assumed. [50]

predict the execution time with this equation:

$$t(E_i, E_{i+1}) = t_K + t_P + t_H + t_R + t_S \tag{3.4}$$

- t_K is the time needed to press a key or mouse button. It depends on factors like user's typing skill. As an average, 400ms can be used.
- t_P is the time needed "to move the pointer from current position to the target position." This value is estimated by Fitts' law [76].
- t_H is the time to switch from mouse to keyboard or vice-versa, a value that needs to be included when using desktop systems. [50] use 400ms as an approximation.
- t_R is the response time of the system. It can be estimate by logged events, such as key press and the screen change with timestamps.
- t_S is the time the user needs to locate information on the screen after the response of the system. It can be directly measured through eye-tracking. But as eye-tracking was not easily accessible at the time the model was developed, this variable was left out.

The predicted execution time $t(E_i, E_{i+1})$ is then compared with the actual execution time T_{i+1} - T_i [50]:

$$T_{i+1} - T_i > t(E_i, E_{i+1}) + \varepsilon \tag{3.5}$$

Additionally to the predicted execution time, a tolerance factor (ε) is added to compensate imprecision in timing and slight variations in the parameters. One could say that it indicates how long the user can pause between two actions within execution phase, without detecting this pause as a chunk boundary. This tolerance factor is a positive number and can either be constant or individually calculated for each chunk by analysing previous chunk behaviour. Although [50] considered the second option more powerful, they used a constant value in their experiment and recommend a value between 200 and 800 ms.

Regarding Equation 3.5, a chunk boundary is recorded if the actual execution time is higher than the predicted time plus the tolerance factor. After executing the algorithm, learnability can be evaluated by analysing the variance of chunk size over time [20].

[20] promoted their method to evaluate learnability as a *discount method* because its cost is minimal. Their algorithm is running in an external program in background, while the user interacts with the system to be evaluated. According to [20], the external program only needs little configuration to setup and ensures confidentiality of user data. The result of the algorithm is numerical data, which can be easily plotted and compared. Therefore, it seems like no integration and no adjustment of the system to evaluate is necessary, if this system delivers an appropriate interaction log.

Related Work and Enhancement

Although change of chunk size with expertise is a well accepted phenomenon, no other research on measuring learnability of a system based on chunking could be found. Though the paper was published over 20 years ago, it got only few attention and no enhancement of further evaluation of the method could be found.

3.3.1.2 Petri Net Based Approach

An other approach to measure learnability of interactive systems is based on the deviation from the expected way of executing certain tasks. The deviation is quantified in so-called fitness values, which indicate how much the observed way of interacting with the system adheres to the intended way. The hypothesis is that the rate of fitness values measured in repeated executions of the system over time indicates the learnability of the system. This method was presented by [77]. The goal was to develop a highly objective method to automatically quantify "(extended) *learnability* of interactive systems during their daily use" [78].

Presentation of Method

According to [77], "[a] highly learnable system should allow a user to know *how to perform correctly* any relevant task of the system after having executed it a few times in the past." Therefore, [77] propose to measure learnability by comparing the intended way of executing the system to the observed behaviour of real users over time.

To allow this comparison, the user's behaviour needs to be recorded. [77] uses automated interaction logging by the system to assess itself, which allows to evaluate learnability during the daily use of the system instead of using it in a controlled lab environment.

In order to describe the expected way of executing the system, [77] developed interaction models with one model per relevant task. The models were realized with petri nets [79]. The transitions of a petri net represent the user actions, like button clicked or text entered, required to achieve a certain task. Only one token is used, initially marking the first place, which represents the start. Figure 3.8 provides an example with *a*, *b*, *c*, *d*, and *e* as user actions.

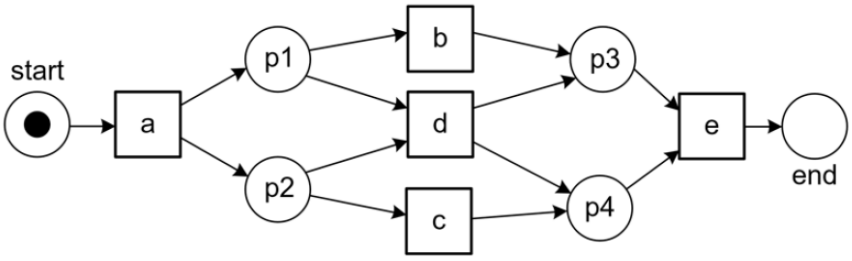


Figure 3.8: Petri net used to represent an interaction model [77]

To enable a comparison of the interaction model with the actual behaviour of the user, the actions in the user log must be able to be mapped to the transitions in the interaction model and vice versa. "[O]nly those fragments of a user log (called traces) that are related to the execution of the relevant task" gets extracted. As a task can be executed several times, multiple traces can be extracted from the user log.

Now having an interaction model and several traces from the user log for each relevant task, the deviation of intended and actual behaviour can be identified. To this end, the extracted traces get replayed over the interaction model: For each step an algorithm checks pair-by-pair if the action in the trace corresponds to the transition of the interaction model. If the action does not corresponds to the transition or vice versa, a deviation is recognized. These deviations can be caused by either mistakes made by the user or a different task execution strategy. To estimate the severity of a deviation the authors applied a cost function that allows favouring one type of deviation over another. However, the exact proceeding is not explained. The outcome are fitness values between 0 and 1, indicating "the extent to which the traces of a user log can be associated with valid execution paths specified by the interaction model."

The hypothesis described in [77] is that the rate of fitness values measured in repeated executions of the system over time indicates the learnability of the system. They suppose that their approach leads to an increased precise measurement of learnability. Furthermore, the authors argue that the strength of their approach imply an evaluation in real user settings, the possibility to weight different deviations and the opportunity to represent different strategies of use, like for novices and experts, through diverse interaction models for one task. One disadvantage is the necessity of suitable user logs.

The presented approach was still in progress when the paper [77] was published, therefore, the hypothesis was not either proven or refuted.

Related Work and Enhancement

Recently, the authors of [77] published a paper [78] where the approach is discussed in more detail and, in addition, results of a conducted experiment are presented. They performed a longitudinal study over four weeks with 23 participants. The participants got homework with growing complexity for each week. After each completed homework, the minimal optimal solution to complete the homework was presented. From homework to homework an increase in the fitness values could be observed. However, it is not mentioned whether the increase was significant. They double-check and confirm the validity of their results by a focus group involving six of the participants in a controlled

environment. The participants were ask to replicate the homework. Although [78] discussed some limitations of the approach, such as the requirement of a structured interaction log or that some aspects can not be modelled by petri nets, they argue that the automated quantification of learnability through petri nets is relevant for task-based software.

In general, research based on petri nets in the context of HCI exists [80, 81]. For example, petri nets have indeed be demonstrated to be convenient for modelling human-computer dialogues [82]. However, only one similar approach [83], published over 20 years ago, could be found. [83] formalized the user's behaviour through a petri net based on a log-file and estimated, based on the petri net, different parameters, such as *behavioural complexity*, *system complexity* and *cognitive structure*. In a user study, [83] showed that behavioural complexity, which is the complexity of the observed behaviour, correlates negatively with learning. Additionally, [83] observed that task solving time further decreases after minimal behavioural complexity is reached.

3.4 Inquiry Methods

Two appropriate inquiry methods were found to assess learnability: *questionnaires* and *diaries*.

3.4.1 Questionnaires

Questionnaires are a popular instrument in HCI, but also in other fields of research. The strength of *questionnaires* is the possibility to quickly and easily get an overview of the users' perception of the system. With low effort a large number of users, who are geographically dispersed, can be questioned [84]. As the user's effort and satisfaction plays a role for learnability (see Chapter 2.1), *questionnaires* may be a cheap method to evaluate learnability.

However, not a single *questionnaire* exclusive for evaluating learnability could be found. There are, in turn, many *questionnaires* for assessing usability, which also include questions about learnability. Some of them even have an own sub-scale for learnability.

Sub-scales have the advantage that the construct that is measured, in this case, learnability, definitely has a reasonable reliability and content variability [85]. Table 3.2 shows some of the most popular *questionnaires* that have a sub-scale for learnability, including information about whether a licence implying fees is required, the number of learnability questions and a citation for the separate sub-scale for learnability.

Questionnaire	Licence fees	Number of learnability questions	Separate sub-scale
Isometrics Usability Inventory (IUI) [85]	No	8 [86, 85]	Yes [85]
ISONORM 9241/10 [87]	No	3 (short version), 5 (long version) [88]	Yes [87]
Purdue Usability Testing Questionnaire (PUTQ) [89]	No	7 [89]	Yes [89]
Questionnaire for User Interaction Satisfaction (QUIS) [90]	Yes	?	Yes [91]
Software Usability Measurement Inventory (SUMI) [92]	Yes	?	Yes [92]
System Usability Scale (SUS) [93]	No	2 [94]	Yes [94]

Table 3.2: Popular usability questionnaires with a sub-scale for learnability

All questionnaires in the table were specially designed with a sub-scale for learnability, except for the *System Usability Scale (SUS)*. Although [93] had not provided a sub-scale for learnability, [94] found out that two learnability-related items can be combined to a sub-scale, scoring learnability with reasonable reliability and high correlation with the overall *SUS* score.

There exist other *questionnaires*, such as the *Post-Study System Usability Questionnaire (PSSUQ)* [95] that may be appropriate to measure learnability as they include questions like *"It was easy to learn to use this system"* and *"I believe I could become productive quickly using this system"*. However, no sub-scale for learnability is provided. Therefore, in order to obtain a reliable and valid result, further research on which elements best measure learnability is required.

For the *Questionnaire for User Interaction Satisfaction (QUIS)* and the *Software Usability Measurement Inventory (SUMI)* it was not possible to find a reliable indication of how many questions the sub-scale for learnability has.

All presented *questionnaires* in Table 3.2 collect quantitative data, except the *Isometrics Usability Inventory (IUI)*, which provide two versions: The second version (called $IsoMetrics^L$) contains the same items as the first version, but additionally provides a second rating for each item. This additional rating asks for the importance of that item and offers the participant the opportunity to freely write down examples that illustrate the previous rating [85]. An example of an item in $IsoMetrics^L$ is given in Figure 3.9.

		Pre-dominantly disagree		So - so		Pre-dominantly agree	No opinion
L.3	The explanations provided help me understand the software so that I become more and more skilled at using it.	**1**	**2**	**3**	**4**	**5**	
		Un-important		So - so		Important	No opinion
	Please rate the importance of the above item in terms of supporting your general impression of the software?	**1**	**2**	**3**	**4**	**5**	
	Can you give a concrete example where you cannot agree with the above statement?						

Figure 3.9: Third item of the learnability sub-scale of $IsoMetrics^L$ [86]

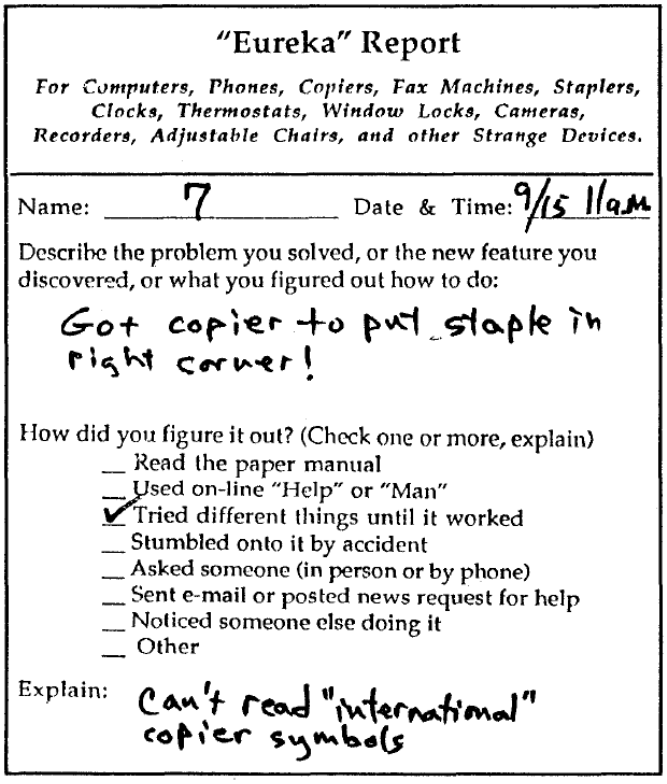
3.4.2 Diaries

For usability, diary studies are well established [96, 97] as they offer the possibility to assess a system during the daily work of the users in their natural environment without relying on log-files. Nevertheless, only one work could be found that developed a *diary*

focussing on learning process [61]. This happens to be one of the first published *diary* approach in HCI research [74]. The main concept is presented in the following chapter.

Presentation of Method

[61] proposed a diary study consisting of a daily activity log, reports about learning progress (see Figure 3.10) and interviews.



"Eureka" Report

For Computers, Phones, Copiers, Fax Machines, Staplers, Clocks, Thermostats, Window Locks, Cameras, Recorders, Adjustable Chairs, and other Strange Devices.

Name: 7 Date & Time: 9/15 11am

Describe the problem you solved, or the new feature you discovered, or what you figured out how to do:

Got copier to put staple in right corner!

How did you figure it out? (Check one or more, explain)

__ Read the paper manual
__ Used on-line "Help" or "Man"
✓ Tried different things until it worked
__ Stumbled onto it by accident
__ Asked someone (in person or by phone)
__ Sent e-mail or posted news request for help
__ Noticed someone else doing it
__ Other

Explain: Can't read "international" copier symbols

Figure 3.10: Reports participants are supposed to fill out whenever they make progress or fail [61]

In the daily log the participant was supposed to briefly describe her/his activity in half-hour intervals. After each day a short interview was planned in which the researcher met with the participant. This enabled the researcher to determine, if the activity log and reports have been filled out with adequate accuracy to reduce biased data. Further, the researcher had the opportunity to detect learning episodes throughout the discussion.

At the end of the one-week study, [61] conducted an one-hour interview covering the participant's learning experience with the system. Essential for learnability were the reports about the learning progress. The participant had to fill a report whenever he or she had learned something, solved a problem or unsuccessfully tried to solve a problem (cf., Figure 3.10).

[61] conducted the study with ten participants. According to [61], "the sampling was too small and inhomogeneous to support strong projections to a larger population". Nevertheless, the results were promising.

Related Work and Enhancement

Although the sampling of the study was insufficient, [61] was published over 20 years ago and received some attention and usability diaries are well established, no evaluation or enhancement towards learnability could be found. In general, however, there are several diary methods in HCI research. For example, [74] gave an overview of some diary methods.

3.5 Formal-Analytical Methods

Formal-analytical methods include all approaches that analyse and describe a user interface based on established formalisms without the involvement of users or user representatives [10].

3.5.1 Attributes Models

Some researchers, such as [98, 99], propose to evaluate learnability by breaking down the term into smaller ones and evaluate these low level terms.

During this thesis, one approach [99] could be found that solely deals with learnability. Therefore, the next section will discuss this approach more closely. Other approaches, which partly deal with learnability, will be introduced afterwards.

3.5.1.1 A Learnability Attributes Model

According to [99], learnability is a quite complex concept. Therefore, an evaluation and hence, improvement of learnability postulates an understanding of factors influencing learnability. Furthermore, [99] claims that there is a need for objective measurements, which are, on the one hand, reproducible and not prone to interplay of various factors like characteristics of the user, the environment or sample size. On the other hand, the measurement should not only assess learnability, but also identify weak areas of the interaction system. [99] did not found any existing methods with this characteristics. For this reason they developed their own method: a model of learnability, which is based on quantification of lower level attributes.

The model breaks down learnability into six main characteristics, such as *Interface Understandability* and *Task Match* (cf., 3.11). These characteristics are further subdivided hierarchically up to seven level. So there are eight level in total. Figure 3.11 presents the top three levels of the model. The attributes of the lowest level are quantified by metrics, which does not require the involvement of users. Instead, the system needs to be analysed.

For instance, *Interface Understandability* is further subdivided in *Global Organization Scheme* and *Representational Adequacy*. The former is further subdivided, in addition to an other sub-characteristic, in *Information Grouping Cohesivness*, which, in turn, is further subdivided. One of these lowest level attributes is *Information Grouping Visual Cohesivness* (cf., Figure 3.12). In order to estimate the *Information Grouping Visual Cohesivness*, all semantically cohesive groups (elements that are optically grouped by e.g. colours, spacing or similar means) need to be counted and divided by the total number of identified groups (this includes also, for example, semantic groups). [99] claims that the closer the value is to 1, the better is the result.

The results of all metrics can be converted to percentage. Since the values have a consistently unit of measure, they can be transferred to higher levels by calculating the average. Thus, one single learnability score can be calculated. By virtue of this procedure, one not only has a value that predicts overall learnability, but also a value

1. **Interface Understandability**
   - 1.1. Global Organization Scheme
     - 1.1.1. Interface Hierarchy
     - 1.1.2. Information Grouping Cohesiveness
   - 1.2. Representational Adequacy
     - 1.2.1. Interface Textual Contents Appropriateness
     - 1.2.2. Interface Graphical Contents Appropriateness
2. **Feedback Suitability**
   - 2.1. Navigability Feedback Completeness
     - 2.1.1.Orientation Suitability
   - 2.2. Task Progress Feedback Appropriateness
     - 2.2.1.Textual Feedback Indicator Appropriateness
     - 2.2.2.Graphical Feedback Indicator Appropriateness
     - 2.2.3.Graphical Feedback Availability
     - 2.2.4.Graphical Feedback Sufficiency
   - **2.3. Task Response Appropriateness**
     - 2.3.1. Object Selection Feedback Appropriateness
     - 2.3.2. Object Movement Feedback Appropriateness
     - 2.3.3.Object Modification Appropriateness
     - 2.3.4.Object Inspection Appropriateness
3. **Predictability**
   - 3.1. Consistency
     - 3.1.1. Internal Consistency
     - 3.1.2. External Consistency (Familiarity)
   - 3.2. Predictive Anchor Information Suitability
     - 3.2.1. Predictive Textual Anchors Suitability
     - 3.2.2. Predictive Graphical Anchors Suitability
   - 3.3. Synthesizability
     - 3.3.1. Descriptive Cues Availability
     - 3.3.2. Visual preview Availability
4. **Task Match**
   - 4.1. Dialogue Content Appropriateness
     - 4.1.1. Dialogue Content Completeness
     - 4.1.2. Economy of Dialogue Contents
   - 4.2. Task Contextual Appropriateness
     - 4.2.1. Contextual Contents Availability
     - 4.2.2. Task economy
5. **System Guidance Appropriateness**
   - 5.1. System Documents Appropriateness
   - 5.2. Help Appropriateness
     - 5.2.1. Help Availability
     - 5.2.2. Help Content Relevancy
     - 5.2.3. Help Content Understandability
     - 5.2.4. Help Content Adequacy
     - 5.2.5. Help Content Helpfulness
     - 5.2.6. Help Content Organization
   - 5.3. System Message Appropriateness
     - 5.3.1. Error Message Appropriateness
     - 5.3.2. Warning Message Appropriateness
   - 5.4. Memory Aids Appropriateness
     - 5.4.1. Iconic Caption Appropriateness
     - 5.4.2. Iconic Supplementary Labels Availability
     - 5.4.3. Pre-filled values/Defaults Availability
     - 5.4.4.??
6. **Operational Momentum**
   - 6.1. Operational Sequence
     - 6.1.1.Screen Sequence Appropriateness
     - 6.1.2.Step Sequence Appropriateness
   - 6.2. Economy of Operation
     - 6.2.1.Operational Logic Suitability
     - 6.2.2.Short cuts Appropriateness

Figure 3.11: Top three levels of the *learnability attributes model* [99]

for each attribute at every level that provides the opportunity to identify weak areas. According to [99], special attention should be paid on scores below 70 %.

As mentioned earlier, all attributes of the lowest level are quantified by predictive metrics, making a total of over 200 metrics defined by the model.

To prove the adequacy of the model, [99] conducted a study. First of all, they predict learnability of two different radio WebApps through applying their model. However, they only concentrated on the two (out of six) characteristics *Interface Understandability* and *Task Match*. Figure 3.12 shows the results, with one column for each WebApp (*DB* and *XM*). Very weak areas are highlighted, which are elements with a score below 70 %. Afterwards, they questioned 33 participants in an online survey with four questions

Characteristic/Sub Characteristics	DB %	XM %
1 Interface Understandability	89.70	86.14
1.1 Global Organization Scheme	85.83	86.88
1.1.1 Interface Hierarchy	75.00	75.00
1.1.1.1 Features Visibility	75.00	75.00
1.1.1.1.1 Main Features comprehensiveness	75.00	75.00
1.1.1.1.2 Main Features Completeness	75.00	75.00
1.1.1.2 Features sequential logic appropriateness	100.00	100.00
1.1.2 Information Grouping Cohesiveness	96.67	98.77
1.1.2.1 Information Grouping Visual Cohesiveness	94.00	96.30
1.1.2.2 Information Grouping Distinctiveness	96.00	100.00
1.1.2.3 Information Grouping Semantic Cohesiveness	100.00	100.00
1.2 Representational Adequacy	90.67	85.95
1.2.1 Interface Textual Contents Appropriateness	94.14	86.05
1.2.1.1 Textual Contents Clarity	91.95	91.45
1.2.1.1.1 Textual Contents Representational Clarity	99.67	100.00
1.2.1.1.2 Textual Contents Conceptual Clarity	84.23	82.89
1.2.1.2 Textual Content Distinctiveness	96.34	80.65
1.2.1.2.1 Representational Distinctiveness	95.11	63.75
1.2.1.2.2 Conceptual Distinctiveness	97.56	97.56
1.2.2 Interface Graphical Contents Appropriateness	89.51	85.92
1.2.2.1 Iconic Labeling appropriateness	95.63	89.66
1.2.2.1.2 Iconic Distinctiveness	93.75	86.06
1.2.2.2 Interface Graphics Appropriateness	96.88	91.02
1.2.2.2.1 Graphics Clarity	100.00	92.19
1.2.2.2.2 Graphics Distinctiveness	93.75	89.84
1.2.2.3 Interface Animation Appropriateness	76.04	77.08
1.2.2.3.1 Animation Clarity	66.67	75.00
1.2.2.3.2 Animations Distinctiveness	85.42	79.17
4 Task Match	75	70.39
4.1 Dialogue Content Appropriateness	75	65.79
4.1.1 Dialogue Content Completeness	75	100.00
4.1.2 Economy of Dialogue Contents	75	31.58
4.2 Task Contextual Appropriateness	75	75.00
4.2.1 Contextual Contents Availability	50	75.00
4.2.2 Task Economy	100	75.00

Figure 3.12: Results of the evaluation of *Interface Understandability* and *Task Match* [99]

focusing on these two characteristics (one question was formulated by [99] itself, the others were derived from the *Software Usability Measurement Inventory (SUMI)* [92] and *Isometrics Usability Inventory (IUI)* [85]), to compare the results with the outcome of the model. The results matched well, as participants preferred the same App as the attributes model for both characteristics. [99] points to the advantage of using the model to have the ability to identify weak areas, thereby enabling targeted improvement of the system.

Related Work and Enhancement

Since the paper was published in 2012, it was only cited nine times (according to Google Scholar). Most of them mention the paper only briefly, it is mainly used to define learnability and its attributes and not as a measurement of learnability. [100] criticize that the approach focuses more on the definition of attributes of a learnable system than on the process of measurement of learnability. Nonetheless, this method was presented in this thesis because it has a different approach than the other methods and could help to select appropriate metrics for measuring learnability.

During the research similar approaches could be found, which, however, have the main focus on usability (e.g., [98, 101, 102, 103, 104]). Since they treat learnability as a component of usability, these models also contain attributes for learnability. One model, called *Quality in Use Integrated Measurement (QUIM)*, has a relatively high degree of development.

Over years members of the Concordia University in Montreal have initially developed and enhanced the model [98, 105, 106]. The last release gets quite a lot of attention, as it was cited over 570 times in 2018 according to Google Scholar. *QUIM* is mainly based on the standard series by ISO.

Like the previous presented model, *QUIM* is hierarchically structured. It includes ten factors, which are decomposed into 26 measurable criteria, which, in turn, are divided into 127 specific metrics. Some criteria can be measured by more than one metric. One of these factors is learnability, which is decomposed into the following criteria [98]:

- Minimal action
- Minimal memory load
- User guidance
- Consistency
- Self-descriptiveness
- Simplicity
- Familiarity

As the name of the model already suggests, the main purpose of the model is the definition and measurement of *quality in use*, which is defined as the quality, while the system is being used. Therefore, in contrast to the previous presented model that contains exclusively predictive metrics, *QUIM* contains also metrics that must be computed with user involvement. Hence, empirical studies are necessary, such as *log-file based analysis*, *video observations* or *surveys*. These metrics are, for example, "the percentage of a task completed" or "the time spent dealing with program errors" [98].

An important aspect of the model is that it is dynamic. It is intended as a conceptual framework serving consistent definitions and guidance in planning for usability measurements. An individual measurement plan can be created depending on aspects such as the class of users for whom the system to evaluate is intended for or the context of use. Furthermore, the model was developed for both novice and expert evaluators [98].

One real-life example of *QUIM* is reported by [107], who created a *questionnaire* based on the proposed factors and criteria of *QUIM*.

3.6 Inspection Methods

Considering the possibilities for expert-based evaluation of usability, two popular methods exist: the *heuristic evaluation* (HE) and the *cognitive walkthrough* (CW).

During a *heuristic evaluation*, experts assess the user interface with regard to its conformity with well-known principles, such as DIN EN ISO 9241 and established usability heuristics [1].

In contrast, a *cognitive walkthrough* is a task-oriented method in which evaluators put themselves in the position of the user and perform typical user tasks [1]. The focus of *CWs* is on the cognitive activities of the user [108], concentrating on the evaluation of learnability [10, 109, 110]. Indeed, [111] have shown that by evaluating the usability of a healthcare information system with both *HE* and *CW*, the problems concerning learnability detected with the help of *CW* were significantly higher.

Although CW is a traditional usability methods, it is explicit recommended by researchers (e.g., [10, 62]) for evaluating learnability. Therefore, in the following the *cognitive walkthrough* is presented.

3.6.1 Cognitive Walkthroughs

First, a short summary of the *CW* method is given. Afterwords, several variations of the original *CW* method are mentioned.

Presentation of Method

"A cognitive walkthrough evaluates the ease with which a typical user can successfully perform a task using a given interface" [109]. The focus is on a task that the user must learn by exploring. That is, for example, by using hints provided by the system, rather than "knowing how to use the system" [109].

The advantage of using *CWs* is that the method can be applied in the early design process, as early system suggestions in the form of written system descriptions or mock-ups are sufficient [10].

During a *CW*, the evaluators put themselves in the position of the user and perform typical user tasks [1]. They evaluate the actions and responses of the system according to the goals and knowledge of a typical user. Therefore, differences between user's

expectations and reality can be detected. The focus of *CWs* is "on the cognitive activities of users, especially on their goals and knowledge when performing a specific task" [112].

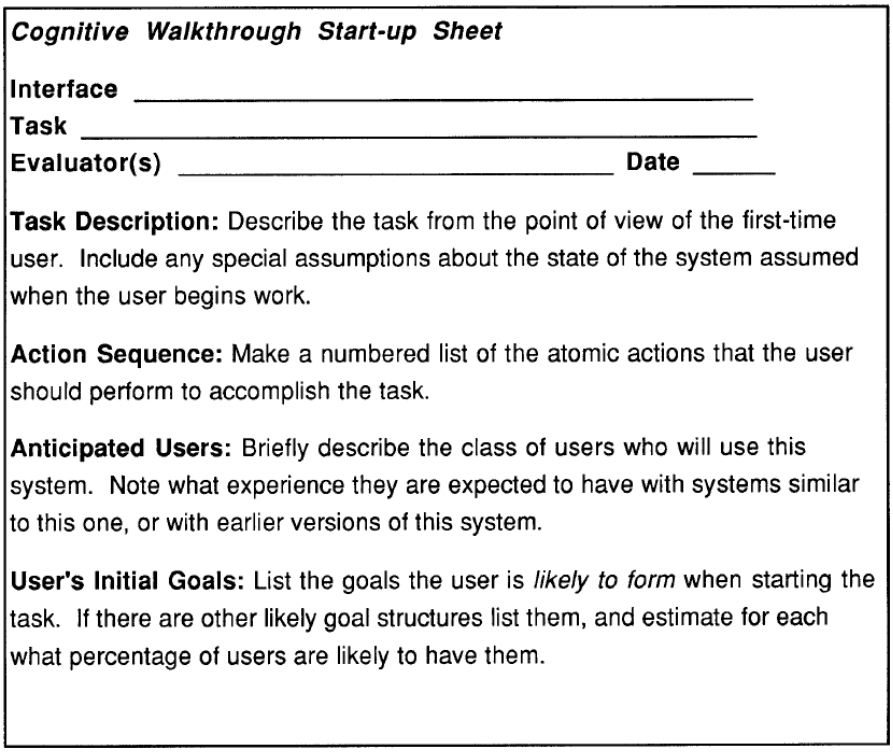
A *CW* consists of two phases: preparation and evaluation [10]. During the preparation phase, the evaluators collect information about the users for whom the system to be evaluated is intended by creating user profiles. Moreover, a set of typical user tasks must be selected, as the system is evaluated in great detail based on specific individual tasks rather than as a whole. It is seldom possible to analyse all tasks that can be conducted with a system. Therefore, the selection of tasks has a significant influence on the evaluation results. The selected tasks should be central for daily work and frequently executed in the users' routine. For each task the evaluators describe in detail how users will likely understand and evaluate this task. One could also say the evaluators try to predict the user's mental model. Thereafter, all necessary actions to accomplish the task are defined and described in detail [10].

Figure 3.13 gives an overview of the results recorded in the preparation phase.

During the second phase, the evaluation phase, all actions of each task are worked through in the previously defined order. Each action is assessed by the evaluators in terms of the background and the knowledge of the user. In doing so the evaluators must answer the following 4 questions [112]:

1. Will the user try to achieve the right effect?
2. Will the user notice that the correct action is available?
3. Will the user associate the correct action with the effect the user is trying to achieve?
4. If the correct action is performed, will the user see progress is being made towards solving the task?

All points that could hinder exploratory learning are considered and documented in detail [10]. In this phase the evaluators should evaluate only and not already try to discuss possible solutions. The search for solutions is done after the *CW* has been performed.



Cognitive Walkthrough Start-up Sheet

Interface _____

Task _____

Evaluator(s) _____ **Date** _____

Task Description: Describe the task from the point of view of the first-time user. Include any special assumptions about the state of the system assumed when the user begins work.

Action Sequence: Make a numbered list of the atomic actions that the user should perform to accomplish the task.

Anticipated Users: Briefly describe the class of users who will use this system. Note what experience they are expected to have with systems similar to this one, or with earlier versions of this system.

User's Initial Goals: List the goals the user is *likely to form* when starting the task. If there are other likely goal structures list them, and estimate for each what percentage of users are likely to have them.

Figure 3.13: Proposed form to record results of the preparation phase of a *CW* [109]

Related Work and Enhancement

The *CW* is a well-accepted method explained in many todays books about usability evaluation methods, such as [1, 10]. However, many researcher criticize *CWs* as being too tedious [112]. Therefore, many extensions of the method exists. [112], for example, reviewed eleven different extensions. There are variants where end users are involved or where the questions that evaluators need to answer are either extended, reduced or completely revised.

One enhancement is the *streamlined cognitive walkthrough* described in [113]. The author of [113] thinks that a *CW* is hard to apply in large software development companies. He discusses three reasons that hinder the effectiveness of a *CW* [113]:

- **Time pressure**: When developing a software product, involved parties, such as managers, developers and designers, are often under a huge time pressure,

and, therefore, want to use their time wisely. Following the proposed procedure by [108, 109], a lot of obvious observations have to be written down. Likewise, answering all four questions for each step is considered very time consuming, and especially for very obvious problems, not effective.

- **Lengthy design discussions**: When a problem is identified by a group of designers, [113] often observes discussions on how to solve this issue instead of using the time on evaluation.

- **Design Defensiveness**: Designers and specification writers tend to defend their work, as their team have already put much effort in their work and, in the short term, identified problems lead to more work for persons that may be already under time pressure.

To overcome these problems, [113] propose the *streamlined cognitive walkthrough*, which is divided into five phases. The first phase is similar to the preparation phase of the classic *CW*. In the second phase the evaluators are getting prepared: The goal of the walkthrough is described as well as an instruction how the walkthrough is conducted, what the evaluators should do and what they should avoid (e.g., lengthy design discussions or design defensiveness). Also certain roles may be assigned to evaluators. In any case, a usability specialist should be empowered as a session leader. The third phase is similar to the evaluation phase of the original *CW*. However, instead of four questions, only the following two questions are answered [113]:

1. "Can you tell a credible story that the user will know what to do?"

2. "If the user does the step correctly, and <describe system response>, is there a credible story to explain that they knew they did the right thing?"

In the next phase only critical information is recorded and get fixed in the last phase [113].

In using the *streamlined cognitive walkthrough*, some compromises have to be made. Perhaps the biggest one is that the causes of a usability problem are not as well understood as in the *CW*, as the more detailed questions of the *CW* will help to understand

the problem. However, [113] recommends the *streamlined cognitive walkthroughs* in large software development companies, as he sees this method as more practical [113].

3.7 Discussion

In the last chapters various methods for assessing learnability have been presented. As shown in Figure 3.1, these include a wide variety of evaluation methods. As diverse as the definitions for learnability are, so diverse are the methods that were found. As well as the disagreement over the definition, there seems to be disagreement about how to measure the learnability. This statement is supported by several publications, such as [9].

The presented methods cover different goals (see Figure 3.14). The *petri net based approach*, *performance measurements* and *chunk detection* quantify the behaviour of the users, whereas *questionnaires* assess attitudes of the users. The main purpose of *diaries* is the collection of attitudinal data, but behaviours could also be self-reported. *Mental model interviews* highly aim at attitudinal data. However, if interviews are supported by clickable prototypes, evaluators have the opportunity to ask participants how they would perform a certain task. The *question-suggestion protocol* can be used to analyse both attitudinal and behavioural data. *Cognitive walkthroughs* try to predict the behaviour of potential users. As evaluators try to empathise with the user's situation, also attitudes may be predicted.

All presented methods collect either qualitative or quantitative data with three exceptions (cf., Figure 3.14). $IsoMetrics^L$, which is the formative version of *IUI*, contains also free text fields for qualitative data collection. The *petri net based approach* quantifies the deviation of user's behaviour in fitness values. However, deviations can be further analysed as interactions and navigation path are modelled in petri nets. The last exception are *diaries* since they can include free text fields for qualitative data collection as well as quantitative elements, such as rating scales.

Additionally, Figure 3.14 shows whether the approaches are usually conducted in laboratory or field or if both conditions are possible. Specific to the *petri net based approach*,

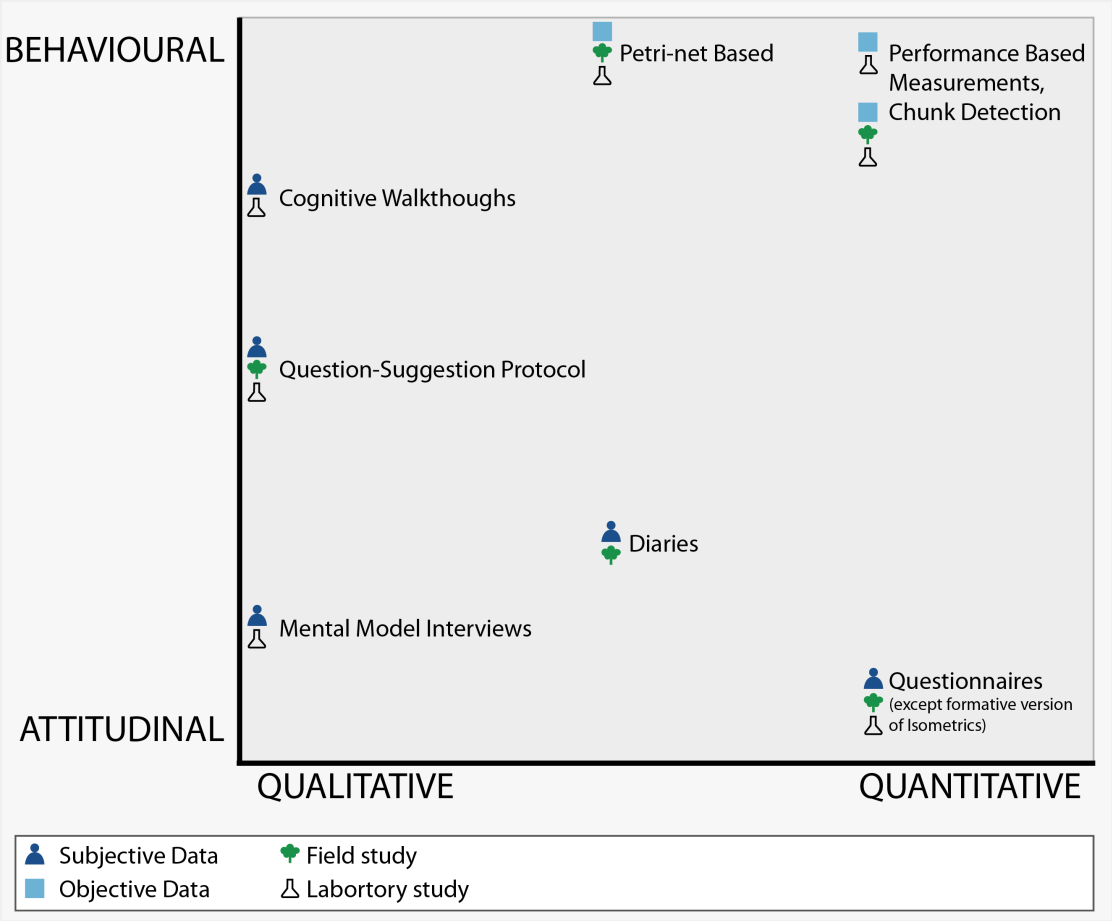


Figure 3.14: Classification of the presented methods to assess learnability with regard to common characteristics (based on [22, 57])

the *chunk detection*, *diaries* and *questionnaires* is that these methods can be carried out in the user's natural environment without having participants directly observed by evaluators. This reduces biased data caused by the effect that participants may behave differently when getting observed [74].

The *learnability attributes model* could not be categorized according to the behavioural-attitudinal dimension, since it is a formal method. However, it can be classified within the other categories: the predicted data is quantitative and objective, as it is based on measurable metrics, and is conducted in laboratory.

Looking more closely at the individual methods, it can be noticed that not only different goals (such as collecting subjective data) are persuaded, but also the way learnability is measured is quite different: For example, *performance based measurements* attempt to measure learnability based on the outcome of the learning process, while *chunk detection* tries to estimate learnability in a more direct way. This phenomena is, amongst other things, due to the manifold aspects of learnability definitions.

As discussed in Chapter 2.1, learnability is defined by aspects such as the increase in efficiency, error-freeness, satisfaction and the amount of required effort. It is conspicuous that the first three aspects are covered very well with presented methods, cognitive effort, however, is not considered in detail. There seems to be even disagreement as, for instance, according to *QUIM* (see Chapter 3.5.1), minimal action and minimal memory load are highly influencing learnability and should therefore considered when evaluation learnability [98]. In contrast, the *PUTQ* contains items that relate to these two aspects, but these form their own sub-scale and are not taken into account in the learnability score.

Generally, it was quite surprising that only few publications considered to observe the user's cognitive effort over time, although there are prominent and well established methods in HCI based on the cognitive load theory [114]. For example, to conduct the overall user-perceived workload the *NASA task load index* can be used [115]. For an objective estimation of the user's workload *secondary task techniques* are well established [116]. Measuring workload is widely applied in different fields of research, such as on educational systems (e.g., [117]) or autonomous driving (e.g., [22]).

Furthermore, as discussed in Chapter 2.2, learning and memory are closely interdependent. However, in the definition of usability and learnability by ISO 9241-110:2006 [10, 17] memorability is not mentioned. On the contrary, [8] defines memorability as a separate aspect of usability next to learnability and describes it with how easy users can re-establish proficiency after not using the system for a certain period. Likewise, it is striking that there are differences in the methods presented with regard to the inclusion of memorability. For instance, with regard to *performance based measurements*, [68] considers memorability in addition to learnability, whereas [69] explicit includes pauses

and distraction tasks to take memory losses into account when assessing learnability. Another example is the *IUI*, which in its learnability sub-scale contains an item that asks for the ease of re-learning a system after a long break [86]. The *PUTQ*, however, does not ask for memorability and memory losses at all [89].

No fundamental research on the relationship between usability, especially learnability, and memorability could be found. Hence, several questions remain unanswered, such as *How does memorability correlates with learnability and other aspects of usability, such as self-descriptiveness?*, *Is it worth the effort to include memory losses when evaluating learnability?* and *When conducting a study using, for example, a performance measurement to assess learnability, is there a significant difference in outcome between a 30-minute simulated distraction task to trigger memory losses (cf., [69]) and a pause of several days between each task execution?*.

Likewise, it was surprising that, although new approaches to measure learnability were developed over the last 40 years, only [9] conducted an extensive literature research on existing approaches. This publication was also the only one which compared at least two methods to find out if a certain method is particularly well-suited for assessing learnability compared to similar approaches (in this case, the *question-suggestion protocol* was compared with *thinking-aloud* [9]). Other publications used only, if any, other methods (usually *questionnaires*) to confirm the validity of their own approach (e.g., [99]).

However, considering advantages and disadvantages of the individual methods, no outstanding approach can be identified since all methods are diverse and have their own strength and weaknesses. To provide an overview, Table 3.4 summarises several strength and weaknesses for each approach. Additionally, the establishment of an approach is rated, which is a subjective assessment based on the extend to which a method has been recognized and how many examples of utilisation have been found.

Methods	Estab.	Strength	Weakness
Mental Model Interviews	+	• No functioning prototype required → applicable early in design process	• Deviations in the user's mental model do not have to lead to learnability issues

Methods	Estab.	Strength	Weakness
		• Involves users early in design process	
Question-Suggestion Protocol	+	• Enables measurement of extended learning [9] • Has the potential to eradicate causes of learnability issues • Specific questions, based on concrete situation, can be asked • No retrospective bias	• Not possible to observe how participants recover from errors and independently figure things out [9] • Possibility of biased data caused by leading participant through suggestions • Possibility of biased data caused by formulation of the questions
Performance Based Measurements	+++	• Quantify learnability • Simple in conduction	• No causes for learnability issues are analysed • Outcome may be hard to interpret without a reference measurement
Chunk Detection	++	• Quantify learnability • Allows further analysis of deviations: At which steps occur deviations? • Natural environment • Extended learning can be measured	• No causes for learnability issues are analysed • Detailed log-file needed • Outcome may be hard to interpret without a reference measurement
Petri Net Based	+	• Quantify learnability • Natural environment • Extended learning can be measured	• Appropriate log-file needed • Some "aspects of the interaction that can not be formalized through Petri nets" [78]

Methods	Estab.	Strength	Weakness
		• Although mainly quantitative: further data (such as were deviations from expected behaviour occur) can be analysed	• By now, the approach is only suitable for wizard-based and structured tasks [78] • Outcome may be hard to interpret without a reference measurement
Questionnaires	+++	• Low time expenditure for participants and evaluator • Relatively low cost [84] • Participants can be geographically dispersed [84] • High level of validity if survey is well-designed and correctly conducted [84] • Often, score interpretable without comparative value (e.g., [118])	• No possibility to ask following up questions [84] • More overview than detailed information [84] • Possibility of biased data (e.g., social desirability, questions related to mood) [84] • Creation and application of questionnaires may seem quite simple, but need to be well-designed to be generalizable [84]
Diaries	++	• Events can be recorded when they occur [96] → reduce retrospective bias [74] • Natural environment • Ideal for longitudinal studies [74] → extended learning can be observed • It can be investigate how participants freely explore the system [74]	• High effort for participants [74] • High effort for evaluator (free text fields have to be analysed) • Reduced compliance may occur [74] • Due to habituation, little changes in a daily questionnaire might be overseen [74] • Increasing chance of participants drop-outs [74]

Methods	Estab.	Strength	Weakness
Attributes Models	+	• Possible to predict how much effect an in increase in one metric has on overall learnability [106] • Helps to assess all aspects influencing learnability	• Many individual values have to be collected to be able to assess learnability as a whole • Mainly without involvement of users
Cognitive Walkthroughs	+++	• No functioning prototype required [62] → applicable early in design process • Helps designers to take the perspective of a user [62] • Can "help to define user's goals and assumptions" [62] • Detect relatively many severe problems [111] • Eradicate causes of learnability issues	• Quite lengthy [62, 113] • Depend on proper task selection: Only those issues are identified that potentially affect the course of the selected task [62, 111]. • Dependence on expertise of evaluator [111] • Without involvement of users [62]

Table 3.4: Establishment, strength and weaknesses of the presented approaches to measure respectively predict learnability

First, not only the specific strength and weaknesses of a method have to be taken into account, but also the advantages and disadvantages of different evaluation styles have to be considered. Field studies are great as users are in their natural environment with, for example, natural interruptions and ambient noise. However, these factors influence the study and internal validity may suffer. Unlike field studies, laboratories provide a controlled environment in a well-equipped room, but results are less generalizable [1, 22].

Also the choice of whether quantitative or qualitative data should be collected, depends on the goals and purpose of the evaluation. Shall it be formative or summative? Are details on how to further improve a system or facts needed, for example, to calculate a return on investment (ROI) to make restructuring of a system to my company's manage-

ment appealing? Qualitative methods are to be preferred if causes and possibilities for improvement are to be analysed. But, if someone just aims of getting a first impression of how users work with a system in order to estimate, whether efforts to improve learnability are necessary or to check whether predefined goals have been reached, quantitative data is preferable.

Now considering the special advantages and disadvantages of individual methods, these also depend heavily on the specific goals of the evaluation and the circumstances. For instance, the huge disadvantage of *chunk detection* and the *petri net based approach* is that an appropriate and very detailed interaction log is needed. If learnability of a complex system without an interaction log shall be evaluated, these methods would be practically unusable. However, if a system is to be evaluated, which anyway has a very detailed logging, the disadvantage is obsolete. Another example are *diaries*, which have many disadvantages. However, they are ideal if environmental influences occurring in longitudinal studies are to be taken into account.

All in all, no general recommendation for a specific method can be given. The choice of an ideal method depends heavily on the requirements and goal of the system being evaluated, its context of application and the objective of the evaluation.

However, one aspect is recommended in literature when evaluating usability in quantitative studies: consider performance as well as satisfaction since there is not always a perfect positive correlation [119, 120, 121]. However, no research could be found analysing the correlation between performance and satisfaction on learnability.

4

AHP

As discussed in the last chapter, there are a lot of different methods, all having their own advantages and disadvantages. Hence, no general advise which method should be used when measuring learnability can be given. Instead, the choice of method strongly depends on the requirements and goals of the system being evaluated, its context of use and the goals of the evaluation. In order to assist in the individual choice of the most appropriate method among the multitude of possibilities, this chapter presents a framework with which a sound decision can be made based on individual rankings of certain criteria. The proposed decision process is based on the *analytical hierarchy process* (AHP) [122].

4.1 AHP Method

> "The lack of a coherent procedure to make decisions is especially troublesome when our intuition alone cannot help us to determine which of several options is the most desirable, or the least objectionable, and neither logic nor intuition are of help" [122].

AHP is a multi-criteria decision-making process, that help to decide between several alternatives based on weighting of decision criteria. The goal of AHP is that decisions can be made in a more organized and rational way without needing much expertise [122].

AHP is based on the human capability to make informed judgements about slight problems. Therefore, using AHP, the problem is decomposed into smaller ones, resulting

in a hierarchical structure. With a pairwise comparisons within each level a decision for the overall problem can be made [122].

First of all, the decision problem needs to be defined. For instance, "to determine what kind of job would be best for him/her after getting his/her PhD" [123]. Afterwards, the problem is structured hierarchically. [123] recommends to define sub-goals of the overall goal and divide them into criteria that must be satisfied to reach the sub-goals. The criteria can be further decomposed [123]. Figure 4.1 shows the hierarchy of the problem to find the best fitting job. Additionally, the possible alternatives are presented (below in Figure 4.1): Job in a domestic or international company, in college or in state university.

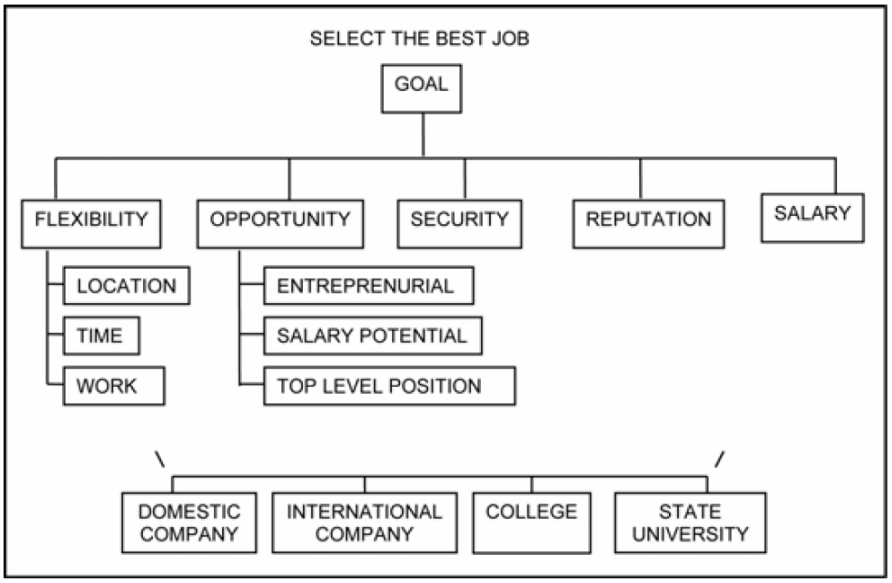


Figure 4.1: Decision hierarchy to find the best fitting job to a certain person [123]

Next, the criteria are compared in pairs, comparing all the direct child elements for each element at a higher level. In the presented example (cf., Figure 4.1), *flexibility*, *opportunity*, *security*, *reputation* and *salary* are compared with each other. Afterwards, the child elements of *flexibility* are compared with each other, then the comparison of the child elements of *opportunity* follows.

On a nine-step scale (1 = "equal important", 3 = "moderate importance", 5 = "strong importance", 7 = "very strong or demonstrated importance" and 9 = "extreme importance")

the decision maker's subjective importance of one criteria is determined in comparison. In each case, it is stated which importance each criterion has in comparison to another one [88, 123]. Figure 4.2 gives an example for a comparison matrix. For instance, *opportunities* is a little more than moderate important compared to *flexibility* regarding the parent element, in this case the overall goal. However, *opportunities* is less important than *security*.

	Flexibility	*Opportunities*	*Security*	*Reputation*	*Salary*	*Priorities*
Flexibility	1	1/4	1/6	1/4	1/8	0.036
Opportunities	4	1	1/3	3	1/7	0.122
Security	6	3	1	4	1/2	0.262
Reputation	4	1/3	1/4	1	1/7	0.075
Salary	8	7	2	7	1	0.506

Figure 4.2: Example for a pairwise comparison matrix [123]

Based on this matrix, the overall importance of each criteria regarding the parent (cf., Figure 4.2, Priorities) is conducted for each row by adding all ratios of the entry of that row divided by the sum of all entries of that column [122].

In addition, a consistency ratio (CR) of the importance judgement can be calculated, with CR = 0% for a perfectly consistent pairwise comparison matrix. Depending on the size of matrix, the CR should be maximal between 5% to 10% [122].

In the last step all alternatives are ranked. Likewise, this happens in pairwise comparison. For each element of the lowest level, all alternatives need to be compared regarding this element in a pairwise comparison matrix. In the example there are nine matrices: for "flexibility of location, time and work", entrepreneurial, salary potential, "top-level position, job security, reputation and salary" [123]. Figure 4.3 shows the matrix for the comparison of the alternatives regarding *salary*. Based on these priorities and on the ranking for each criterion respectively sub-criterion, the best alternative can be calculated [123].

	Domestic Co	*Int'l Co*	*College*	*State Univ.*	*Priorities*
Domestic company	1	4	3	6	0.555
Int'l company	1/4	1	3	5	0.258
College	1/3	1/3	1	2	0.124
State University	1/6	1/5	1/2	1	0.064

Figure 4.3: Example for a pairwise comparison matrix for the alternatives regarding *salary* [123]

4.2 Related Work

AHP is widely applied to a variety of decision-making problems. Also approaches using AHP in HCI can be found (e.g., [124]). One publication [125] discusses the usage of AHP for the choice of a usability evaluation method. However, the focus is on interactive adaptive systems. The proposed hierarchical structure is, therefore, not suitable for the choice of learnability measurement methods as it contains criteria such as *type of adaptation* or *adaptation layers*.

Another approach, described by [126], has a quite similar goal to this thesis: "to support the selection of the most appropriate methods depending on project and organizational constraints". Therefore, [126] developed a tool called *Usability Planner*. This tool aim to support in the choice of a method to evaluate usability over all project stages. Based on the individual selection of certain constraints, methods to evaluate usability are proposed. The tool is accessible under [127]. However, it is unclear how methods are proposed in the background. Obviously, the selection is not based on AHP. Therefore, it is not possible to express preferences such as one constraint is more important to oneself as another one. Furthermore, [126] only concentrated on project, user, task and product constraints. Constraints with respect to the evaluation goal were not considered.

4.3 AHP for Selecting Methods to Measure Learnability

In this thesis, support for finding the most appropriate method to measure learnability based on AHP is given. This has the advantage that the decision can be made in

an organized and rational way quite easily. Additionally, AHP allows a fine granular prioritization of criteria, which leads to a more proper decision based on individual requirements. Finally, several tools to support in the decision process based on AHP already exist, such as [128]. This chapter provides a hierarchical structure of the problem to find an evaluation method. Additionally, the previously presented methods are ranked with respect to this structure. Therefore, the idea is that a practitioner, searching for a method to evaluate learnability of an individual system within an individual project can transfer these rankings to an existing tool for AHP. Afterwards, the practitioner only needs to rank the importance of different criteria to him and gets the best method proposed for him.

4.3.1 Problem Hierarchy

The problem that needs to get solved, and, therefore, the goal that should be reached, is to find the most appropriate method for an individual project to evaluate learnability.

The aim is to provide universal criteria, so it can be widely applied. Unfortunately, no consistent advise could be found which criteria should be considered when selecting an evaluation method. Therefore, an own hierarchy was conducted in this thesis, that takes study conditions as well as study goals, participant requirements and effort for evaluators into account.

The hierarchical structure of the goal to find an appropriate method to evaluate learnability is presented in Figure 4.4.

The first criteria formalise whether there is a preference in the study conditions: How important is the involvement of participants? Are there any preferences for a certain type of study? How important is that the effort for participants (e.g., time expenditure) is minimal? Is it so important that an evaluation without participants would be acceptable even if user involvement would be favoured? Is it important how many participants are minimal required? The latter plays a role when representatives of users are hard to recruit, for example, if target users are persons with seldom diseases.

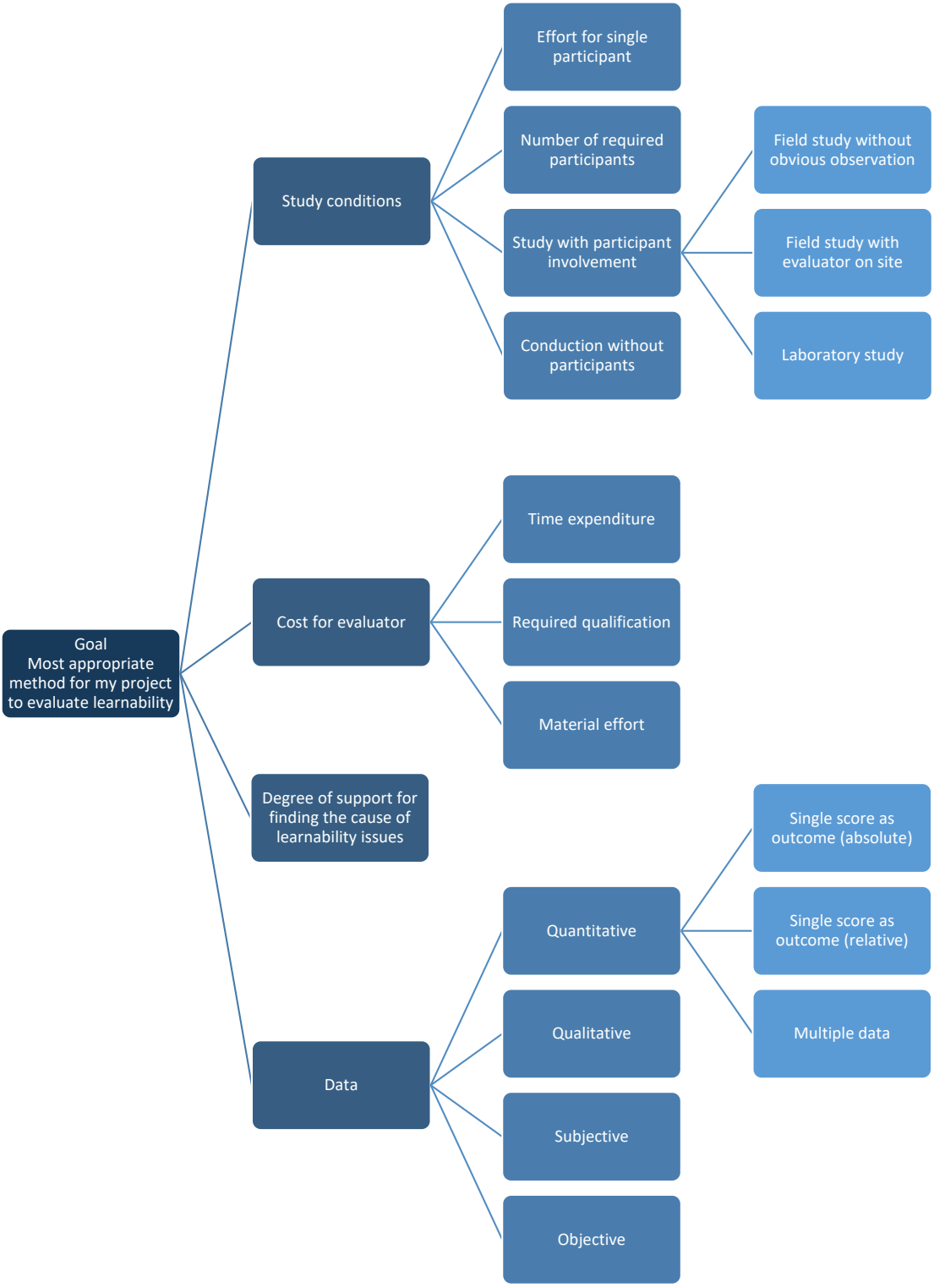


Figure 4.4: Decision hierarchy for finding the most appropriate method for my project to evaluate learnability

The second criterion takes the costs for the evaluator into account. It is further subdivided in the importance of time expenditure, which includes the required time for study preparation, conduction and evaluation, the required qualification of the evaluator and the material effort such as licence fees or equipment. Rating a high importance of one of these elements means that this element should be minimal.

The third criteria is the extend to which the method to measure learnability should assist in finding the cause of issues. A high rating expresses that the method should help as much as possible in finding causes.

Through last criteria, practitioners can express whether they have requirements on the resulting data. Is objective or subjective data desired? Is there any preference of quantitative or qualitative data? For quantitative data, is a single score favoured? For a single score, a distinction is made between those that can be interpreted by novice without having a reference system or an alternative design to compare the value with and those that can only be correctly interpreted by highly experienced evaluators or in comparison of two measurements (e.g., A/B testing). An example for such an 'absolute score' is the *SUS*. Because of the widespread appliance of the *SUS*, researchers can give advise on how to interpreted individual scores (e.g., [118]).

A fundamental precondition in choosing a method is the development stage and whether a detailed log-file exist. An *either-or decision* has to be made rather than weighting between several sub-criteria, such as early development stage versus end of development. Therefore, the development stage and the existence of a detailed log-file are not formalised as criteria in the decision hierarchy. Instead, these two factors have to be considered in the selection of the alternatives.

In Listing A.1 in the appendix, the hierarchical structure is provided in comma-separated values (CSV) in order to afford the opportunity to import the decision hierarchy into tools for AHP, such as [128].

4.3.2 Examples for Ratings of Criteria

To give an idea of how the criteria could be rated, some examples are given in the following.

Scenario 1

Starting point is a software manufacturer of a comprehensive, relatively complex expert system, which has been on the market for years and is regularly updated through releases. So far, system improvements have been based on general, unstructured customer feedback via a service hotline and the intuition of developers. However, the importance of targeted user involvement and evaluation of usability to improve product quality and attractiveness was recognized.

Since the system requires considerable familiarization period, which is partly accompanied by training classes, learnability has been identified as an important aspect. An evaluation of the system regarding learnability (and usability) has never been performed. For this reason, an initial overview of how good the learnability of the system actually is, to decide whether further efforts are necessary, is requested. Therefore, the effort for evaluators and participants should be low. This includes, for example, the avoidance of license fees. Perfect for the evaluators would be a score with which they can see at a glance how good the learnability is.

The ranking of the criteria that may arise under these circumstances is shown in Figure 4.5.

Scenario 2

Another scenario might appear in early design phase where only ideas, general workflows and possibly some mock-ups exist. A relatively quick and easy answer on how good the learnability of the system might be and what needs to be improved is required. Expensive user studies should be avoided at this stage. Usability experts are available and a comprehensive study with user representatives is planned later, when functional

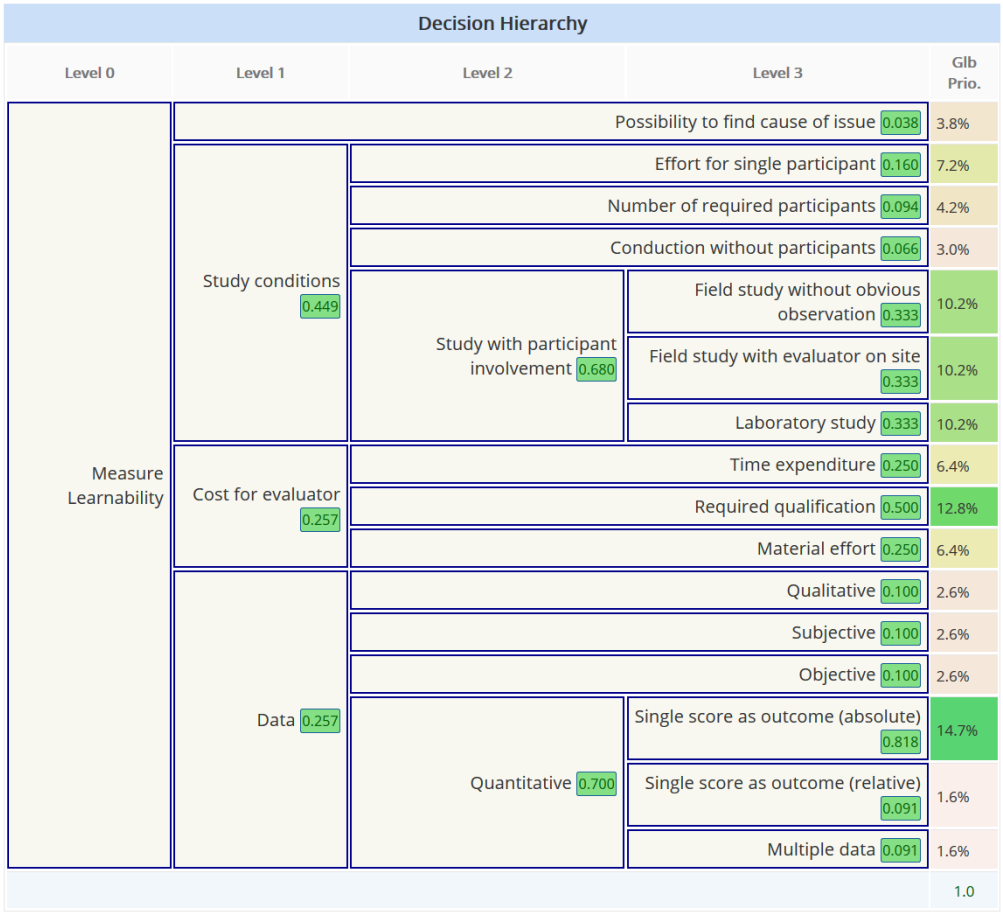


Figure 4.5: Example weighting of the criteria for scenario 1 (conducted with [128])

prototypes exist. The main purpose is to evaluate the design ideas quickly and get the possibility to repair fundamental learnability issues at early stages of development.

The ranking of the criteria that may arise under these circumstances is shown in Figure 4.6.

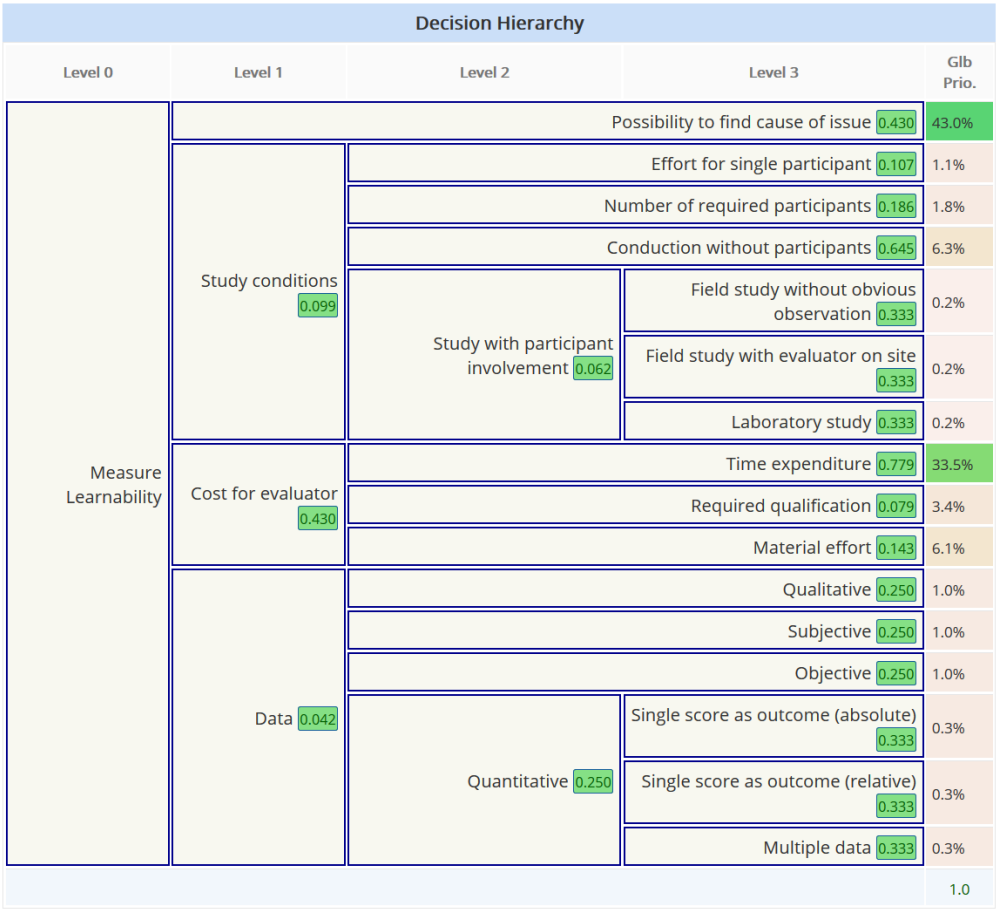


Figure 4.6: Example weighting of the criteria for scenario 2 (conducted with [128])

Scenario 3

The last scenario is relatively at the end of the development process. All important functionalities are implemented and a beta version can be released. The manufacturer has some exclusive beta test costumers. Therefore, the number of required participants

should be relatively low. Employees have expertise in evaluation. The system has a high demand on learnability as it has a long familiarization period. Additionally, not all system functions are in daily use. Therefore, a longitudinal study is desired. Furthermore, the context of use of the system is strongly influenced by the environment: there are often interruptions, for example, by incoming phone calls or situations in which more urgent tasks have to be done spontaneously. Until then, only laboratory studies have been conducted where such influences are difficult to reproduce. Therefore, the evaluators thought that a field study would be helpful.

The ranking of the criteria that may arise under these circumstances is shown in Figure 4.7.

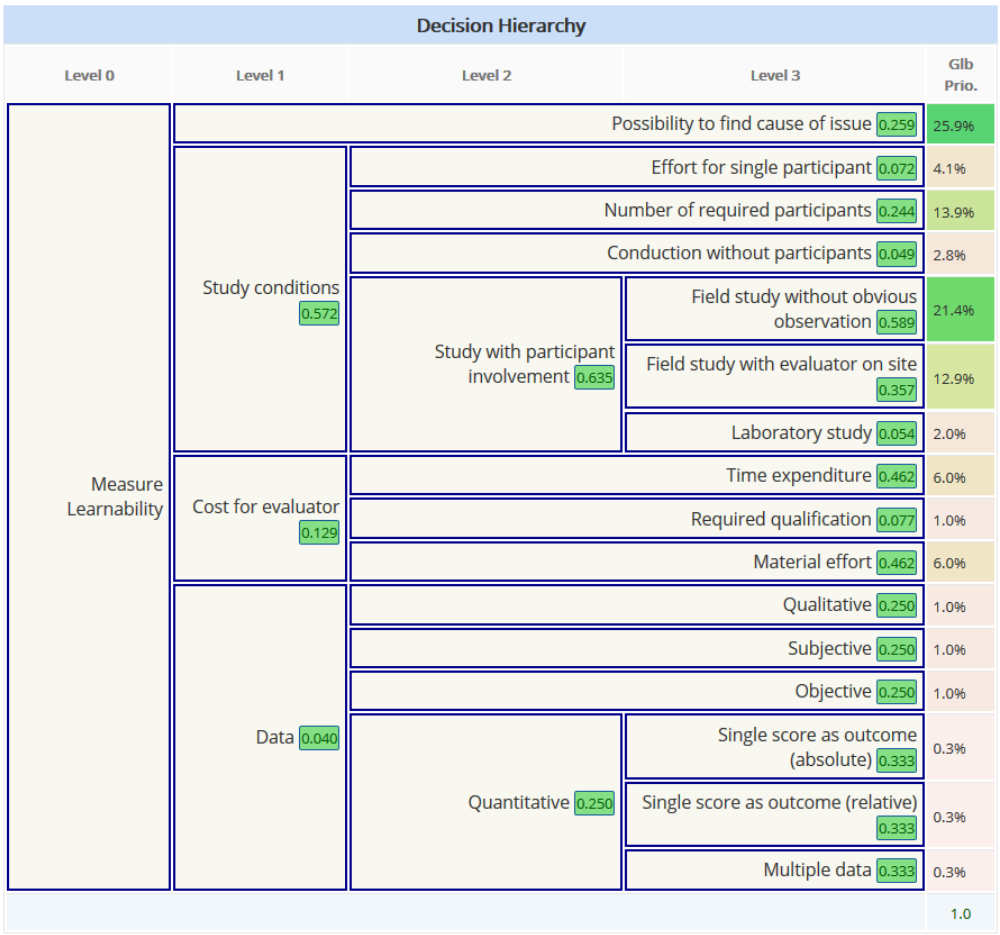


Figure 4.7: Example weighting of the criteria for scenario 3 (conducted with [128])

4.3.3 Ratings of Alternatives

Next, alternatives must be ranked with respect to each lowest level criteria of the decision hierarchy. The result is proposed in Table 4.1.

	MMI	QSP	PBM	CD	PNB	Ques	Diary	LAM	CWs
Possibility to find cause of issue	0.184	0.184	0.017	0.024	0.026	0.075	0.121	0.184	0.184
Effort for single participant	0.045	0.045	0.045	0.171	0.171	0.105	0.017	0.200	0.200
Number of required participants	0.141	0.141	0.032	0.032	0.032	0.032	0.141	0.226	0.226
Conduction without participants	0.040	0.040	0.040	0.040	0.040	0.040	0.040	0.360	0.360
Field study without obvious observation	0.027	0.027	0.027	0.321	0.321	0.111	0.111	0.027	0.027
Field study with evaluator on site	0.168	0.331	0.076	0.073	0.073	0.0731	0.073	0.065	0.065
Laboratory study	0.122	0.122	0.122	0.122	0.122	0.122	0.020	0.122	0.122
Time expenditure	0.075	0.075	0.075	0.288	0.075	0.217	0.012	0.041	0.135
Required qualification	0.068	0.042	0.190	0.111	0.040	0.305	0.098	0.130	0.018
Material effort	0.031	0.035	0.070	0.183	0.183	0.070	0.063	0.183	0.183
Qualitative	0.215	0.215	0.022	0.033	0.039	0.022	0.215	0.022	0.215
Subjective	0.184	0.184	0.020	0.020	0.020	0.184	0.184	0.020	0.184
Objective	0.024	0.024	0.220	0.220	0.220	0.024	0.024	0.220	0.024
Single score as outcome (absolute)	0.031	0.031	0.080	0.080	0.080	0.432	0.032	0.202	0.032
Single score as outcome (relative)	0.024	0.024	0.117	0.117	0.223	0.223	0.024	0.223	0.024
Multiple data	0.020	0.020	0.217	0.217	0.116	0.153	0.017	0.205	0.034

Table 4.1: Proposed preferences for methods to measure learnability with respect to each criterion (with MMI = *mental model interviews*, QSP = *question-suggestion protocol*, PBM = *performance based measurement*, CD = *chunk detection*, PNB = *petri net based approach*, Ques = *questionnaires*, LAM = *learnability attributes model* and CWs = cognitive walkthroughs)

Each row in Table 4.1 is the outcome of one pairwise decision matrix. The matrix of the first row is shown in Figure 4.8.

	Category	Priority	Rank
1	Mental Model Interviews	18.4%	1
2	Question-Suggestion Protocol	18.4%	1
3	Performance Based Measurements	1.7%	9
4	Chunk Detection	2.4%	8
5	Petri-Net Based	2.6%	7
6	Questionnaires	7.5%	6
7	Diaries	12.1%	5
8	Learnability Attributes Model	18.4%	1
9	Cognitive Walkthrough	18.4%	1

	1	2	3	4	5	6	7	8	9
1	1	1.00	9.00	7.00	7.00	3.00	2.00	1.00	1.00
2	1.00	1	9.00	7.00	7.00	3.00	2.00	1.00	1.00
3	0.11	0.11	1	0.50	0.50	0.20	0.17	0.11	0.11
4	0.14	0.14	2.00	1	0.50	0.25	0.20	0.14	0.14
5	0.14	0.14	2.00	2.00	1	0.17	0.14	0.14	0.14
6	0.33	0.33	5.00	4.00	6.00	1	0.33	0.33	0.33
7	0.50	0.50	6.00	5.00	7.00	3.00	1	0.50	0.50
8	1.00	1.00	9.00	7.00	7.00	3.00	2.00	1	1.00
9	1.00	1.00	9.00	7.00	7.00	3.00	2.00	1.00	1

Figure 4.8: Left, resulting priorities with respect to the *possibility to find the cause of learnability issues* are shown. Right, the individual judgements in a decision matrix are shown (conducted with [128])

The higher a value, the better the alternative is appropriate with respect to that criteria. For example, with respect to *possibility to find the cause of learnability issues mental model interviews*, *question-suggestion protocol*, *learnability attributes model* and *cognitive walkthroughs* are most appropriate. With respect to the *effort for a single participant learnability attributes model* and cognitive walkthroughs are most appropriate followed by *petri net based approach* and *chunk detection*. Participants have the highest effort at *mental model interviews* and the *question-suggestion protocol*.

Regarding scenario 1 (see Chapter 4.3.2), *questionnaires* are proposed to be the most appropriate method based on the suggested rating of the alternatives (cf., Table 4.1). The result is presented in Figure 4.9.

Note that the ranking of the alternatives (cf., Table 4.1) is a subjective weighting, which, furthermore, strongly depends on concrete circumstances such as the concrete utilisation of an evaluation method and the participant's system usage. For instance, the weighting of *diaries* with respect to quantitative data collection depends on whether the *diary* involve elements like rating scales. Another example is the rating of *chunk detection* with

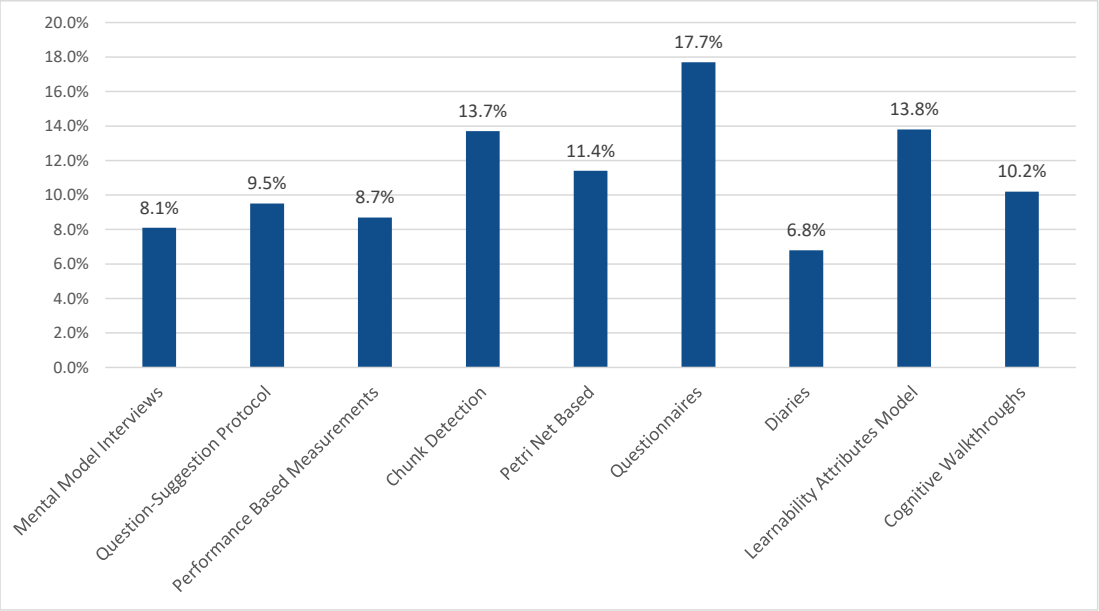


Figure 4.9: Weights of alternatives for scenario 1

respect to the required effort for a single participant. If the system to evaluate is already in-use by the participant or new users become participants, there is no extra effort for the participants as they are using the system anyway. But, if participants have to start using a system they would not have under other circumstances, effort would be ranked higher. A further example is the estimation of *performance based measurements* with respect to the effort for a single participant. The rating strongly depends on how much trials are planned.

For this ranking (cf., Table 4.1), it was assumed that *performance based measurements* are conducted over several trials, so the participants have a time expenditure of around 90 minutes. Furthermore, it was assumed that the *petri net based approach* and *chunk detection* are applied with regular users as participants. Moreover, regarding *questionnaires* this thesis only took the effort required to carry them out into account without a potentially required laboratory study. With respect to *diaries* a high compliance of the participant was assumed.

Note that the development phase and whether a detailed log-file exists are not formalised in the decision hierarchy. Therefore, practitioners have to eventually adapt the choice of alternatives. In early design phase, in order to evaluate ideas and non-functional mock-ups, only *mental model interviews* and *cognitive walkthroughs* are appropriate. Furthermore, if there is no possibility for a detailed log-file, *chunk detection* and the *petri net based approach* have to be excluded.

4.4 Discussion

In this thesis, an approach based on AHP to assist practitioners in selecting the most appropriate method to measure learnability was proposed. The proposed criteria of the decision hierarchy (cf., Figure 4.4) are kept very general. This allows to include diverse alternatives such as analytical next to empirical methods. Therefore, practitioners are invited to add further alternatives.

Additionally, the proposed weighting of the alternatives may need to be modified based on circumstances of the project and planned evaluation. However, the proposed weighting is intended on the one hand to provide a template for an individual ranking and on the other hand to provide a structured overview of the characteristics of existing alternatives.

In the future, a validation of the proposed hierarchy and the weightings of the alternatives towards their appropriateness in finding the most suitable method to measure learnability is necessary.

All in all, the approach is considered to be valuable in assisting to find the most appropriate method. However, no AHP tool free of charges could be found where existing weightings can be easily imported. [128] supports the specification of the decision hierarchy in CSV. Therefore, the proposed hierarchy can be easily transferred. But, despite the possibility to export data in CSV, no possibility could be found importing data, such as the weightings of alternatives.

Hence, for an improved assistance on finding possibilities to measure learnability, an easily accessible tool, comparable to the *Usability Planner* [127], but that is based on AHP and provides templates on alternatives, would be desirable.

5

Conclusions

This thesis gave a structured overview of the definitions and psychological background of learnability as well as of existing methods to measure learnability. Although, learnability has been of interest in HCI for the last 40 years, there is still no consensus on how to define and evaluate learnability.

Therefore, several different definitions exist, which describe learnability with diverse aspects, such as the increase in efficiency, satisfaction or the amount of required effort. Furthermore, there is discrepancy about whether the term *learnability* should be limited to initial learning.

Of course there are also different methods to measure learnability, since in general, diverse goals shall be attainable. Competing goals are, for example, finding learnability issues versus getting a single score or preferences of objective data versus subjective data. In addition, there are several methods in respond to available resources of evaluators, such as existing interaction logs, equipment or the evaluators' experience in usability evaluation. However, taking a detailed look to the methods, disunity can be observed. This includes, for example, the evaluation of cognitive effort or of memorability.

Furthermore, it was very surprising that although new approaches to measure learnability were developed over the last 40 years, only one publication could be found that conducted an extensive literature research on existing approaches. Likewise, this was the only publication that compared at least two similar methods for its value to measure learnability in a study. Several of the methods proposed in this thesis to measure learnability got only few attention.

Considering these aspects, it seems that there is a lack of fundamental research. Especially with respect to factors that should be considered or are not needed to be considered as well as the effectiveness of certain methods in measuring learnability and in uncovering learnability issues. Only for two aspects there is fundamental research in the area of learnability: mental models and chunking. Several approaches to evaluate usability are based on mental models. For learnability, *cognitive walkthroughs*, which are based on the theory of mental models, and *mental model interviews* were conducted. However, chunking mainly aim to design principles rather than on measurement techniques. Only one approach with respect to chunking was found and this have got only few attention. Apart from this publication, no research was found on whether chunking could be generally suitable for measuring learnability or not.

A practical problem, caused by the variety of methods in combination with the lack of juxtaposition, is the selection of the most appropriate method for one's own project. To assist practitioners in their choice, a framework based on AHP was conducted in this thesis. A hierarchy was proposed (cf., Figure 4.4) with general decision criteria. Additionally, the methods to measure learnability presented in this thesis were weighted with respect to that hierarchy. This hierarchy and the weighting of alternatives are intended to be used by practitioners as a template to find the most appropriate method for them. In addition, through the weighting of the alternatives with respect to the criteria, a structured overview of methods to asses learnability is provided. In summary, it can be said that the framework based on AHP is intended to propose the most appropriate alternative as well as give a structured overview of existing alternatives. For the future, however, a validation of the proposed hierarchy and the weightings of the alternatives in terms of their appropriateness in finding the most suitable method to measure learnability is required.

All in all, there are various possibilities to measure learnability. Nevertheless, because of several reasons, which were discussed in this thesis, practitioners are faced with the challenge of finding methods to measure learning as well as selecting a suitable one. This thesis is intended to provide an overview and assist in the choice of method. For the future, further research and an easily accessible tool providing informations on different methods and assisting in the choice of method based on AHP are desirable.

Bibliography

[1] Jacobsen, J., Lorena, M.: Praxisbuch Usability und UX. Rheinwerk Verlag (2017)

[2] Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., Diakopoulos, N.: Designing the user interface: strategies for effective human-computer interaction. Pearson (2018)

[3] Nielsen, J.: Usability 101: Introduction to usability (2003)

[4] Struckmeier, A.: Usability vs. User Experience – Hauptsache Spaß!? https://www.usabilityblog.de/usability-vs-user-experience-hauptsache-spas/ (2012) Accessed on 14 Aug 2018.

[5] Hern, A.: Hawaii missile false alarm due to badly designed user interface, reports say. https://www.theguardian.com/technology/2018/jan/15/hawaii-missile-false-alarm-design-user-interface (2018) Accessed on 14 Aug 2018.

[6] Flaherty, K.: What the Erroneous Hawaiian Missile Alert Can Teach Us About Error Prevention. https://www.nngroup.com/articles/error-prevention/ (2018) Accessed on 14 Aug 2018.

[7] Chistyakov, A., Soto-Sanfiel, M.T., Martí, E., Igarashi, T., Carrabina, J.: Objective Learnability Estimation of Software Systems. In: International Conference on Ubiquitous Computing and Ambient Intelligence, Springer (2016) 503–513

[8] Nielsen, J.: Usability engineering. Academic Press (1993)

[9] Grossman, T., Fitzmaurice, G., Attar, R.: A survey of software learnability: metrics, methodologies and guidelines. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM (2009) 649–658

[10] Sarodnick, F., Brau, H.: Methoden der Usability Evalution: Wissenschaftliche Grundlagen und praktische Anwendung. Hogrefe (2015)

[11] Michael, L.: Partial observability and learnability. Artificial Intelligence **174** (2010) 639–669

[12] Bau, D., Gray, J., Kelleher, C., Sheldon, J., Turbak, F.: Learnable programming: blocks and beyond. Communications of the ACM **60** (2017) 72–80

[13] Perlich, C., Merugu, S.: Multi-relational learning for genetic data: Issues and challenges. In: Fourth International Workshop on Multi-Relational Data Mining (MRDM-2005). (2005)

[14] Lohr, L.: Designing the instructional interface. Computers in Human Behavior **16** (2000) 161–182

[15] Brainerd, C.J., Reyna, V.F.: Can age× learnability interactions explain the development of forgetting? Developmental Psychology **26** (1990) 194

[16] Hornbæk, K.: Current practice in measuring usability: Challenges to usability studies and research. International journal of human-computer studies **64** (2006) 79–102

[17] Heimgärtner, R.: Human Factors of ISO 9241-110 in the Intercultural Context. Advances in Ergonomics In Design, Usability & Special Populations: Part **3** (2014) 18

[18] García-Mundo, L., Genero, M., Piattini, M.: Towards a construction and validation of a serious game product quality model. In: Games and Virtual Worlds for Serious Applications (VS-Games), 2015 7th International Conference on, IEEE (2015) 1–8

[19] Michelsen, C.D., Dominick, W.D., Urban, J.E.: A methodology for the objective evaluation of the user/system interfaces of the MADAM system using software engineering principles. In: Proceedings of the 18th annual Southeast regional conference, ACM (1980) 103–109

[20] Santos, P.J., Badre, A.: Discount learnability evaluation. Technical report, Georgia Institute of Technology (1995)

[21] Linja-aho, M.: Creating a framework for improving the learnability of a complex system. Human Technology: An Interdisciplinary Journal on Humans in ICT Environments (2006)

[22] Dix, A., Finlay, J.E., Abowd, G.D., Beale, R.: Human-Computer Interaction (3rd Edition). Prentice-Hall, Inc. (2003)

[23] Leung, R.A.: Improving the learnability of mobile devices for older adults. PhD thesis, University of British Columbia (2011)

[24] Gluck, M.A., Mercado, E., Cathrine, M.E.: Learning and Memory: From Brain to Behavior. Worth Publishers (2008)

[25] Lieberman, D.A.: Human Learning and Memory. Cambridge Univ. Press (2012)

[26] Seufert, T., Brünken, R., Leutner, D.: Psychologische Grundlagen des Lernens mit Neuen Medien. Univ., Zentrum für Qualitätssicherung in Studium und Weiterbildung (2003)

[27] Brown, P.C., Roediger, H.L., McDaniel, M.A.: Make it stick. Harvard University Press (2014)

[28] Eysenck, M.W.: Pyschology for AS Level. Taylor & Francis (2005)

[29] Atkinson, R.C., Shiffrin, R.M.: Human memory: A proposed system and its control processes1. In: Psychology of learning and motivation. Volume 2. Elsevier (1968) 89–195

[30] Anderson, J.R.: Cognitive psychology and its implications. Worth Publishers (2010)

[31] Miller, G.A.: The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological review **63** (1956) 81

[32] Craik, F.I., Lockhart, R.S.: Levels of processing: A framework for memory research. Journal of verbal learning and verbal behavior **11** (1972) 671–684

[33] Baddeley, A.D.: Working memory Oxford. England: Oxford Uni (1986)

[34] Anderson, J.R., Matessa, M., Lebiere, C.: ACT-R: A theory of higher level cognition and its relation to visual attention. Human-Computer Interaction **12** (1997) 439–462

[35] Tulving, E., et al.: Episodic and semantic memory. Organization of memory **1** (1972) 381–403

[36] Tulving, E.: Episodic memory: From mind to brain. Annual review of psychology **53** (2002) 1–25

[37] Vargha-Khadem, F., Gadian, D.G., Watkins, K.E., Connelly, A., Van Paesschen, W., Mishkin, M.: Differential effects of early hippocampal pathology on episodic and semantic memory. Science **277** (1997) 376–380

[38] Spiers, H.J., Maguire, E.A., Burgess, N.: Hippocampal Amnesia. Neurocase **7** (2001) 357–382

[39] Eysenck, M.W., Keane, M.T.: Cognitive psychology: A student's handbook. Taylor & Francis (2015)

[40] Gardiner, M.M., Christie, B.: Applying cognitive psychology to user-interface design. John Wiley & Sons, Inc. (1987)

[41] Newell, A., Rosenbloom, P.S.: Mechanisms of skill acquisition and the law of practice. Cognitive skills and their acquisition **1** (1981) 1–55

[42] Heathcote, A., Brown, S., Mewhort, D.: The power law repealed: The case for an exponential law of practice. Psychonomic bulletin & review **7** (2000) 185–207

[43] Leibowitz, N., Baum, B., Enden, G., Karniel, A.: The exponential learning equation as a function of successful trials results in sigmoid performance. Journal of Mathematical Psychology **54** (2010) 338–340

[44] Roessingh, J., Hilburn, B.: The Power Law of Practice in adaptive training applications. (2000)

[45] Rivera, D., Eng, P.: Efficiency Assessment of an e-Commerce Data Management Tool Using Learning Curves. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting. Volume 48., SAGE Publications Sage CA: Los Angeles, CA (2004) 902–906

[46] Johnson, E.J., Bellman, S., Lohse, G.L.: Cognitive lock-in and the power law of practice. Journal of Marketing **67** (2003) 62–75

[47] Chase, W.G., Simon, H.A.: Perception in chess. Cognitive psychology **4** (1973) 55–81

[48] Shneiderman, B.: Exploratory experiments in programmer behavior. International Journal of Computer & Information Sciences **5** (1976) 123–143

[49] Barfield, W.: Expert-novice differences for software: Implications for problem-solving and knowledge acquisition. Behaviour & Information Technology **5** (1986) 15–29

[50] Santos, P.J., Badre, A.N.: Automatic chunk detection in human-computer interaction. In: Proceedings of the workshop on Advanced visual interfaces, ACM (1994) 69–77

[51] Fein, R.M., Olson, G.M., Olson, J.S.: A mental model can help with learning to operate a complex device. In: INTERACT'93 and CHI'93 conference companion on Human factors in computing systems, ACM (1993) 157–158

[52] Norman, D.A.: Some observations on mental models. In: Mental models. Psychology Press (2014) 15–22

[53] Chandra, S., Blockley, D.I.: Cognitive and computer models of physical systems. International Journal of Human Computer Studies **43** (1995) 539–559

[54] Kellogg, W.A., Breen, T.J.: Evaluating User and System Models: Applying Scaling Techniques to Problems in Human-computer Interaction. In: Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface. CHI '87, ACM (1987) 303–308

[55] Gediga, G., Hamborg, K.C., Düntsch, I.: Evaluation of software systems. Encyclopedia of computer science and technology **45** (2002) 127–53

[56] Hanington, B., Martin, B.: Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions. Rockport Publishers (2012)

[57] Rohrer, C.: When to Use Which User-Experience Research Methods. https://www.nngroup.com/articles/which-ux-research-methods/ (2014) Accessed on 31 July 2018.

[58] Ivory, M.Y., Hearst, M.A.: The state of the art in automating usability evaluation of user interfaces. ACM Computing Surveys (CSUR) **33** (2001) 470–516

[59] Lewis, J.R.: Sample sizes for usability tests: mostly math, not magic. interactions **13** (2006) 29–33

[60] Stickel, C., Fink, J., Holzinger, A.: Enhancing universal access–EEG based learnability assessment. In: International conference on universal access in human-computer interaction, Springer (2007) 813–822

[61] Rieman, J.: The diary study: a workplace-oriented research tool to guide laboratory efforts. In: Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems, ACM (1993) 321–326

[62] Holzinger, A.: Usability engineering methods for software developers. Communications of the ACM **48** (2005) 71–74

[63] Linja-aho, M.: Evaluating and Improving the Learnability of a Building Modeling System. Master's thesis, Helsinki University of Technology (2005)

[64] Bravo-Lillo, C., Cranor, L.F., Downs, J., Komanduri, S.: Bridging the gap in computer security warnings: A mental model approach. IEEE Security & Privacy **9** (2011) 18–26

[65] Young, I.: Mental models: aligning design strategy with human behavior. Rosenfeld Media (2008)

[66] Kato, T.: What "question-asking protocols" can say about the user interface. International Journal of Man-Machine Studies **25** (1986) 659–673

[67] Isaksen, H., Iversen, M., Kaasboll, J., Kanjo, C.: Design of tooltips for health data. In: IST-Africa Week Conference (IST-Africa), 2017, IEEE (2017) 1–8

[68] Fares, E., Cheung, V., Girouard, A.: Effects of Bend Gesture Training on Learnability and Memorability in a Mobile Game. In: Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces, ACM (2017) 240–245

[69] Coyle, C.L., Peterson, M.: Learnability Testing of a Complex Software Application. In: International Conference of Design, User Experience, and Usability, Springer (2016) 560–568

[70] Albert, W., Tullis, T.: Measuring the user experience: collecting, analyzing, and presenting usability metrics. Newnes (2013)

[71] Hoffman, R.R., Marx, M., Amin, R., McDermott, P.L.: Measurement for evaluating the learnability and resilience of methods of cognitive work. Theoretical Issues in Ergonomics Science **11** (2010) 561–575

[72] Hofman, R.: Range statistics and the exact modeling of discrete non-gaussian distributions on learnability data. In: International Conference of Design, User Experience, and Usability, Springer (2011) 421–430

[73] Billman, D., Archdeacon, J., Deshmukh, R., Feary, M., Holbrook, J., Stewart, M.: Alignment of Technology to Work: Design & Evaluation Representation. In: INTERACT 2015 Adjunct Proceedings: 15th IFIP TC. 13 International Conference on Human-Computer Interaction 14-18 September 2015, Bamberg, Germany. Volume 22., University of Bamberg Press (2015) 81

[74] Gerken, J.: Longitudinal Research in Human-Computer Interaction. PhD thesis, University Konstanz (2011)

[75] Card, S.K., Moran, T.P., Newell, A.: The keystroke-level model for user performance time with interactive systems. Communications of the ACM **23** (1980) 396–410

[76] Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement. Journal of experimental psychology **47** (1954) 381

[77] Hanteer, O., Marrella, A., Mecella, M., Catarci, T.: A petri-net based approach to measure the learnability of interactive systems. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, ACM (2016) 312–313

[78] Marrella, A., Catarci, T.: Measuring the Learnability of Interactive Systems Using a Petri Net Based Approach. In: Proceedings of the 2018 on Designing Interactive Systems Conference 2018, ACM (2018) 1309–1319

[79] Murata, T.: Petri nets: Properties, analysis and applications. Proceedings of the IEEE **77** (1989) 541–580

[80] Pizziol, S., Tessier, C., Dehais, F.: Petri net-based modelling of human–automation conflicts in aviation. Ergonomics **57** (2014) 319–331

[81] Bernonville, S., Kolski, C., Leroy, N., Beuscart-Zéphir, M.C.: Integrating the SE and HCI models in the human factors engineering cycle for re-engineering Computerized Physician Order Entry systems for medications: Basic principles illustrated by a case study. International journal of medical informatics **79** (2010) e35–e42

[82] Riahi, I., Moussa, F.: A formal approach for modeling context-aware human–computer system. Computers & Electrical Engineering **44** (2015) 241–261

[83] Rauterberg, M., Aeppli, R.: Learning in man-machine systems: the measurement of behavioural and cognitive complexity. In: Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on. Volume 5., IEEE (1995) 4685–4690

[84] Lazar, J., Feng, J.H., Hochheiser, H.: Research methods in human-computer interaction. Morgan Kaufmann (2017)

[85] Gediga, G., Hamborg, K.C., Düntsch, I.: The IsoMetrics usability inventory: an operationalization of ISO 9241-10 supporting summative and formative evaluation of software systems. Behaviour & Information Technology **18** (1999) 151–164

[86] Hamborg, K.C.: IsoMetrics Questionnaires. `http://www.isometrics.uni-osnabrueck.de/qn.htm` (n.d.) Accessed on 08 June 2018.

[87] Prümper, J.: Software-evaluation based upon ISO 9241 part 10. In: Human Computer Interaction. Springer (1993) 255–265

[88] Pataki, K., Prümper, J., Thüring, M.: Die Gewichtung von Usability-Aspekten anhand der Analytic Hierarchy Process-Methode von Saaty. Tagungsband UP07 (2007)

[89] Lin, H.X., Choong, Y.Y., Salvendy, G.: A proposed index of usability: a method for comparing the relative usability of different software systems. Behaviour & information technology **16** (1997) 267—277

[90] Chin, J.P., Diehl, V.A., Norman, K.L.: Development of an instrument measuring user satisfaction of the human-computer interface. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM (1988) 213—218

[91] Sauro, J., Lewis, J.R.: Quantifying the user experience: Practical statistics for user research. Morgan Kaufmann (2016)

[92] Kirakowski, J., Corbett, M.: SUMI: The software usability measurement inventory. British journal of educational technology **24** (1993) 210—212

[93] Brooke, J.: SUS: a 'quick and dirty' usability scale. In Jordan, P.W., Thomas, B., McClelland, I.L., Weerdmeester, B., eds.: Usability Evaluation in Industry. Taylor & Francis (1996) 189–194

[94] Lewis, J.R., Sauro, J.: The factor structure of the system usability scale. In: International conference on human centered design, Springer (2009) 94–103

[95] Lewis, J.R.: Psychometric evaluation of the PSSUQ using data from five years of usability studies. International Journal of Human-Computer Interaction **14** (2002) 463—488

[96] Tomitsch, M., Singh, N., Javadian, G.: Using diaries for evaluating interactive products: the relevance of form and context. In: Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction, ACM (2010) 204–207

[97] MacDonald, C.M., Atwood, M.E.: Changing perspectives on evaluation in HCI: past, present, and future. In: CHI'13 extended abstracts on human factors in computing systems, ACM (2013) 1969–1978

[98] Seffah, A., Donyaee, M., Kline, R.B., Padda, H.K.: Usability measurement and metrics: A consolidated model. Software Quality Journal **14** (2006) 159–178

[99] Rafique, I., Weng, J., Wang, Y., Abbasi, M.Q., Lew, P., Wang, X.: Evaluating software learnability: A learnability attributes model. In: Systems and Informatics (ICSAI), 2012 International Conference on, IEEE (2012) 2443–2447

[100] Coyle, C.L., Peterson, M.: Learnability Testing of a Complex Software Application. In: International Conference of Design, User Experience, and Usability, Springer (2016) 560–568

[101] Dubey, S.K., Rana, A., Sharma, A.: Usability evaluation of object oriented software system using fuzzy logic approach. Int. J. Comput. Appl **43** (2012) 35–41

[102] Shackel, B.: Usability-context, framework, definition, design and evaluation. Human factors for informatics usability (1991) 21–37

[103] Abran, A., Khelifi, A., Suryn, W., Seffah, A.: Usability Meanings and Interpretations in ISO Standards. Software Quality Journal **11** (2003) 325–338

[104] Hasan, L.A., Al-Sarayreh, K.T.: An integrated measurement model for evaluating usability attributes. In: Proceedings of the International Conference on Intelligent Information Processing, Security and Advanced Communication, ACM (2015) 94

[105] Donyaee, M.K.: Towards an integrated model for specifying and measuring quality in use. Master's thesis, Concordia University (2001)

[106] Padda, H.K.: QUIM map: a repository for usability/quality in use measurement. Master's thesis, Concordia University (2003)

[107] Wardhana, S., Sabariah, M.K., Effendy, V., Kusumo, D.S.: User interface design model for parental control application on mobile smartphone using user centered design method. In: Information and Communication Technology (ICoIC7), 2017 5th International Conference on, IEEE (2017) 1–6

[108] Wharton, C., Bradford, J., Jeffries, R., Franzke, M.: Applying cognitive walkthroughs to more complex user interfaces: experiences, issues, and recommendations. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM (1992) 381–388

[109] Polson, P.G., Lewis, C., Rieman, J., Wharton, C.: Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. International Journal of man-machine studies **36** (1992) 741–773

[110] Khajouei, R., Hasman, A., Jaspers, M.W.: Determination of the effectiveness of two methods for usability evaluation using a CPOE medication ordering system. international journal of medical informatics **80** (2011) 341–350

[111] Khajouei, R., Zahiri Esfahani, M., Jahani, Y.: Comparison of heuristic and cognitive walkthrough usability evaluation methods for evaluating health information systems. Journal of the American Medical Informatics Association **24** (2017) e55–e60

[112] Mahatody, T., Sagar, M., Kolski, C.: State of the art on the cognitive walkthrough method, its variants and evolutions. Intl. Journal of Human–Computer Interaction **26** (2010) 741–785

[113] Spencer, R.: The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems, ACM (2000) 353–359

[114] Chandler, P., Sweller, J.: The split-attention effect as a factor in the design of instruction. British Journal of Educational Psychology **62** (1992) 233–246

[115] Hart, S.G., Staveland, L.E.: Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In: Advances in psychology. Volume 52. Elsevier (1988) 139–183

[116] Baber, C., Jenkins, D.P., Walker, G.H., Rafferty, L.A., Salmon, P.M., Stanton, N.A.: Human Factors Methods: A Practical Guide for Engineering and Design. Ashgate Publishing, Ltd. (2013)

[117] Hollender, N., Hofmann, C., Deneke, M., Schmitz, B.: Integrating cognitive load theory and concepts of human–computer interaction. Computers in Human Behavior **26** (2010) 1278–1288

[118] Bangor, A., Kortum, P., Miller, J.: Determining what individual SUS scores mean: Adding an adjective rating scale. Journal of usability studies **4** (2009) 114–123

[119] Nielsen, J., Levy, J.: Measuring usability: preference vs. performance. Communications of the ACM **37** (1994) 66–75

[120] Frøkjær, E., Hertzum, M., Hornbæk, K.: Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems, ACM (2000) 345–352

[121] Nielsen, J.: User Satisfaction vs. Performance Metrics. https://www.nngroup.com/articles/satisfaction-vs-performance-metrics/ (2012) Accessed on 04 Aug 2018.

[122] Saaty, T.L.: How to make a decision: the analytic hierarchy process. Interfaces **24** (1994) 19–43

[123] Saaty, T.L.: Decision making with the analytic hierarchy process. International journal of services sciences **1** (2008) 83–98

[124] Delice, E.K., Güngör, Z.: The usability analysis with heuristic evaluation and analytic hierarchy process. International Journal of Industrial Ergonomics **39** (2009) 934–939

[125] Dhouib, A., Trabelsi, A., Kolski, C., Neji, M.: An approach for the selection of evaluation methods for interactive adaptive systems using analytic hierarchy process. In: Research Challenges in Information Science (RCIS), 2016 IEEE Tenth International Conference on, IEEE (2016) 1–10

[126] Ferre, X., Bevan, N.: Usability planner: a tool to support the process of selecting usability methods. In: IFIP Conference on Human-Computer Interaction, Springer (2011) 652–655

[127] Bevan, N., Ferre, X., Antón Escobar, T.: Usability planner: Plan which methods to use to support User Centred Design. Optionally prioritize the project stages where usability will provide most benefit. http://www.usabilityplanner.org (n.d.) Accessed on 31 Aug 2018.

[128] Goepel, K.: AHP Online System - BPMSG. https://bpmsg.com/academic/ahp.php (2017) Accessed on 16 July 2018.

A

Appendix

```
Measure Learnability: Study conditions, Cost for evaluator, Possibility to
    find cause of issue, Data;

Study conditions: Effort for single participant, Number of required
    participants, Study with participant involvement, Conduction without
    participants;

Study with participant involvement: Field study without obvious observation,
    Field study with evaluator on site, Laboratory study;

Cost for evaluator: Time expenditure, Required qualification, Material
    effort;

Data: Quantitative, Qualitative, Subjective, Objective;

Quantitative: Single score as outcome (absolute), Single score as outcome
    (relative), Multiple data;
```

Listing A.1: Decision hierarchy in CSV for finding the most appropriate method for my project to evaluate learnability

List of Figures

1.1 Aspects of user experience (adapted from [1, 4]) 2

2.1 Dialogue principles of ISO 9241-110:2006 (own representation, based on [17]) 6
2.2 Usability attributes by [8] (own representation) 8
2.3 Multi-store model (adapted from [28, 30]) 13
2.4 Hypothesized structure of long-term memory (adapted from [28]) 16
2.5 Visualisation of measured data. In the left plot the typical curve of the power be seen. In the right plot the data is presented in log-log coordinates [44] 20
2.6 Learning curves by Nielsen [8] 22

3.1 Overview of methods to assess learnability (general classification inspired by [1, 10, 58]) 27
3.2 Comparison of the number of learnability issues averaged over all tasks identified by the *question-suggestion* and the *thinking-aloud protocol*. Results are grouped by the level of experience of the participants [9] . . . 33
3.3 Categories of observed learnability issues [9] 34
3.4 Completion time in seconds for different training conditions [68] 36
3.5 Percentage improvement in completion time for each task [69] 37
3.6 Visit duration observed over number of visits of various travel websites [46] 37
3.7 Classification of user actions to chunks [20] 43
3.8 Petri net used to represent an interaction model [77] 46
3.9 Third item of the learnability sub-scale of $IsoMetrics^L$ [86] 50
3.10 Reports participants are supposed to fill out whenever they make progress or fail [61] 51
3.11 Top three levels of the *learnability attributes model* [99] 54
3.12 Results of the evaluation of *Interface Understandability* and *Task Match* [99] 55
3.13 Proposed form to record results of the preparation phase of a *CW* [109] . 60

3.14 Classification of the presented methods to assess learnability with regard to common characteristics (based on [22, 57]) 63

4.1 Decision hierarchy to find the best fitting job to a certain person [123] . . . 72
4.2 Example for a pairwise comparison matrix [123] 73
4.3 Example for a pairwise comparison matrix for the alternatives regarding *salary* [123] 74
4.4 Decision hierarchy for finding the most appropriate method for my project to evaluate learnability 76
4.5 Example weighting of the criteria for scenario 1 (conducted with [128]) . . 79
4.6 Example weighting of the criteria for scenario 2 (conducted with [128]) . . 80
4.7 Example weighting of the criteria for scenario 3 (conducted with [128]) . . 81
4.8 Left, resulting priorities with respect to the *possibility to find the cause of learnability issues* are shown. Right, the individual judgements in a decision matrix are shown (conducted with [128]) 83
4.9 Weights of alternatives for scenario 1 84

List of Tables

3.1 Illustrative interpretation of the $\bar{i}$ scale by [71, 72] 40
3.2 Popular usability questionnaires with a sub-scale for learnability 49
3.4 Establishment, strength and weaknesses of the presented approaches to measure respectively predict learnability 68

4.1 Proposed preferences for methods to measure learnability with respect to each criterion (with MMI = *mental model interviews*, QSP = *question-suggestion protocol*, PBM = *performance based measurement*, CD = *chunk detection*, PNB = *petri net based approach*, Ques = *questionnaires*, LAM = *learnability attributes model* and CWs = cognitive walkthroughs) . 82

Name: Manuela Unsöld Matriculation number: 936735

Honesty disclaimer

I hereby affirm that I wrote this thesis independently and that I did not use any other sources or tools than the ones specified.

Ulm, ...

. . . . . .

Manuela Unsöld