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

Verbesserung der Meetinganalyse in Entwicklerteams unter Verwendung von praktischen Tutorials

Improving meeting analysis in development teams using tutorials

Masterarbeit

im Studiengang Informatik

von

Ann-Christin Haak

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

Hannover, 09.06.2020

Zusammenfassung

Die Kommunikation in Softwareprojekt-Teams ist ein wichtiger Aspekt für eine gute Zusammenarbeit und für die erfolgreiche Entwicklung der Software und somit schlussendlich für den Projekterfolg. Ein wichtiges Medium zur Kommunikation und zur Informationsweitergabe sind Meetings, da in relativ kurzer Zeit viele Personen informiert werden können. Die Interaktionen der Softwareentwickler während der Meetings können sich dabei in positiver sowie in negativer Weise auf den weiteren Verlauf des Projektes auswirken. Daher kann eine Interaktionsanalyse in regulären Team-Meetings eingesetzt werden, um frühzeitig potenzielle Probleme aufzudecken.
Ein bereits in der Psychologie etabliertes videobasiertes Verfahren zur Analyse des Kommunikationsverhaltens in Meetings ist *act4teams*, welches wissenschaftlich fundiert Kompetenzen von Teams misst. Da dieses Verfahren nur wenig geeignet für die Anwendung in Softwareprojekt-Meetings ist, wurde *act4teams-SHORT* eingeführt, welches es Softwareentwicklern ermöglicht, eine vereinfachte Analyse während des Meeting durchzuführen. Jedoch birgt die anfängliche Nutzung von *act4teams-SHORT* Schwierigkeiten für unerfahrene Meetingbeobachter.

Um die anfänglichen Schwierigkeiten zu adressieren, wurden Personen befragt, die *act4teams-SHORT* bereits nutzten. Daraus wurden Maßnahmen abgeleitet, die in zwei Tutorialkonzepte integriert wurden. Beide Konzepte verfolgen verschiedene Vorgehensweisen und präsentieren die Lerninhalte auf unterschiedliche Weisen. Im Anschluss wurden beide Tutorials in das bereits bestehende Tool integriert. Dabei galt es herauszufinden, welches Tutorial effektiver ist und somit die Softwareentwickler auf die Interaktionsanalyse mit *act4teams-SHORT* besser vorbereitet.
Im Rahmen eines Experimentes mit 30 Probanden wurde getestet, inwieweit die Tutorials einen messbaren Lerneffekt hervorrufen. Zudem wurde überprüft, welches der beiden Tutorials effektiver und einfacher anzuwenden ist sowie welches den Nutzern besser gefällt.
Die Ergebnisse zeigen, dass beide Tutorials einen positiven Einfluss auf die Interaktionsanalysen von unerfahrenen Meetingbeobachtern haben. Die subjektiven Angaben der Probanden zeigen, dass beide Tutorials den Teilnehmern gefallen und einfach anzuwenden sind. Eine Präferenz für ein Tutorial kann nicht ausgesprochen werden, da beide sehr ähnlich gute objektive sowie subjektive Ergebnisse erzielten.

Abstract

Communication in software development teams is an important aspect for a good collaboration and a successful development of software and consequently for the project's success. A relevant medium for communication and information sharing are meetings, since many people can be informed in a relatively short time. The interactions of software engineers during meetings can affect the further process of a project in a positive as well as in a negative manner. Therefore, an interaction analysis can be used in regular team meetings in order to detect potential problems at an early stage.
A video-based observation method for analyzing communication behavior in meetings already established in psychology is *act4teams*, which scientifically assesses the competences of teams. As this method is not very suitable for use in software development meetings, *act4teams-SHORT* was introduced, which enables software developers to conduct a simplified analysis during the meeting. However, the initial use of *act4teams-SHORT* holds difficulties for inexperienced meeting observers.

People who had already used *act4teams-SHORT* were interviewed in order to address the initial difficulties. Based on this, measures were derived, which were integrated into two tutorial concepts. Both concepts follow different approaches and present the learning content in different ways. Subsequently, both tutorials were integrated into the already existing tool. The goal was to find out which tutorial is more effective and thus prepares the software developers better for the interaction analysis with *act4teams-SHORT*.
To what extent the tutorials cause measurable learning effects was evaluated in an experiment with 30 subjects. Furthermore, it was verified which tutorial is more effective and easier to use, and also which tutorial users like better. The results show that both tutorials have a positive impact on the interaction analysis by inexperienced meeting observers. The subjective statements of the study participants show that both tutorials are easy to be applied and that both are liked. A preference for a tutorial cannot be pronounced, because both achieved similarly good objective and subjective results.

Inhaltsverzeichnis

1 Einleitung **1**
- 1.1 Motivation . . . 1
- 1.2 Zielsetzung . . . 2
- 1.3 Struktur der Arbeit . . . 3

2 Grundlagen **5**
- 2.1 Act4teams . . . 5
  - 2.1.1 Codierschema . . . 5
  - 2.1.2 Interpretation einer Meetinganalyse . . . 7
  - 2.1.3 Anwendbarkeit auf Softwaremeetings . . . 8
- 2.2 Act4teams-SHORT . . . 9
  - 2.2.1 Entwicklung . . . 9
  - 2.2.2 Codierschema . . . 10
  - 2.2.3 Das Tool und dessen Anwendung . . . 11
  - 2.2.4 Herausforderung der Analyse . . . 14
- 2.3 Lernstrategien . . . 14
- 2.4 Usability . . . 15
  - 2.4.1 Personas . . . 18
  - 2.4.2 Papierprototypen . . . 19

3 Verwandte Arbeiten **21**
- 3.1 Interaktionsanalysen von Meetings aus verschiedenen Bereichen . . . 21
- 3.2 Interaktionsanalysen von Softwareprojekt-Meetings . . . 22

4 Konzept und Implementierung der Lernverfahren **25**
- 4.1 Ideenfindung . . . 26
  - 4.1.1 Zukünftige Nutzer . . . 26
  - 4.1.2 Schwierigkeiten . . . 30
  - 4.1.3 Maßnahmen . . . 31
  - 4.1.4 Konzepte . . . 33
  - 4.1.5 Papierprototypen . . . 34
- 4.2 Meetingaufnahme . . . 39
- 4.3 Implementierung der Lernverfahren . . . 40
  - 4.3.1 Erklärungen zu den Kategorien . . . 41
  - 4.3.2 Tutorials . . . 41

4.3.3 Probeanalyse . . . 44
4.3.4 Zeitstrahlgegenüberstellung . . . 45

5 Evaluation **47**
5.1 Zielsetzung und Hypothesen . . . 47
5.1.1 Zielsetzung . . . 47
5.1.2 Hypothesen . . . 48
5.2 Planung des Experimentes . . . 50
5.2.1 Meetingausschnitte . . . 50
5.2.2 Experimentsubjekte . . . 52
5.2.3 Design des Experimentes . . . 53
5.3 Durchführung des Experimentes . . . 54
5.3.1 Experimentumgebung . . . 54
5.3.2 Ablauf des Experimentes . . . 55
5.3.3 Zeitraum und Dauer . . . 58
5.4 Analyse und Interpretation . . . 59
5.4.1 Deskriptive Statistiken der Probanden . . . 59
5.4.2 Datenaufbereitung . . . 61
5.4.3 Datenanalyse . . . 62
5.4.4 Ergebnisse . . . 64
5.4.5 Verbesserte Meetinganalysen . . . 64
5.4.6 Vergleich der Tutorials . . . 67
5.4.7 Zusammenfassung der Ergebnisse der Hypothesenprüfung . . . 71
5.5 Einschränkungen der Validität . . . 71
5.5.1 Conclusion validity . . . 71
5.5.2 Internal validity . . . 72
5.5.3 Construct validity . . . 73
5.5.4 External validity . . . 74
5.6 Zusammenfassung . . . 74
5.7 Implikationen für die Anwendung der Tutorials . . . 75

6 Zusammenfassung und Ausblick **77**
6.1 Zusammenfassung . . . 77
6.2 Ausblick . . . 78

Literaturverzeichnis **83**

Abbildungsverzeichnis **85**

Tabellenverzeichnis **87**

Anhang **89**

1 Einleitung

Der kontinuierliche Informationsaustausch innerhalb von Entwicklerteams ist unerlässlich für einen positiven Entwicklungsprozess und somit einen erfolgreichen Projektabschluss [24]. Werden wichtige Informationen unvollständig oder gar nicht weitergegeben, kann dies zu Problemen mit potenziell hohem Ausmaß führen [15].
Eine effektive Möglichkeit wichtige Informationen in kurzer Zeit an viele Personen weiterzutragen, sind Meetings [15]. In Meetings wird die Expertise aller Mitarbeiter eines Teams gebündelt und somit ergeben sich wichtige Anstöße weiterer Arbeitsschritte [12]. Diese Vorteile ließen Meetings im Entwicklungsprozess immer mehr an Relevanz zunehmen und fordern daher mittlerweile viel Zeit der Software Engineers ein [15], [25]. Verbringen die Teammitglieder jedoch zu viel Zeit in Meetings, wirkt sich dieses eher negativ auf die wahrgenommene Produktivität aus [18]. Aber nicht nur die verbrachte Zeit, sondern auch die Qualität der Meetings hat eine Auswirkung auf das subjektive Produktivitätsempfinden [11]. So kann eine schlechte Qualität eines Meetings Frustration und Unzufriedenheit bei den Teilnehmern auslösen [11].
Ein Beispiel ist die prozedurale Kommunikation. Diese ist eine verbale Verhaltensweise, die Gruppendiskussionen strukturiert [11]. In Studien konnte gezeigt werden, dass negative prozedurale Kommunikation sich negativ auf die Meetingzufriedenheit, die wahrgenommene Produktivität und den organisatorischen Erfolg auswirkt [11].
Um funktionale und dysfunktionale Interaktionsprozesse in einem Meeting aufzudecken, kann eine Interaktionsanalyse von Meetings angewendet werden [11].

1.1 Motivation

Ein bereits in der Psychologie etabliertes Verfahren zur Interaktionsanalyse ist *act4teams*. Dabei wird ein Teammeeting auf Video aufgezeichnet und im Anschluss von einem Experten analysiert [11]. Anhand der Analyseergebnisse können dann Handlungsempfehlungen für künftige Meetings abgeleitet werden [11].
Die Verwendung des *act4teams*-Verfahrens in der Softwareentwicklung birgt jedoch einige Probleme [13], [25]. Beispiele sind (datenschutz-)rechtliche Belange beim Aufzeichnen von Videos oder das notwendige Hintergrundwissen des Analysten für eine

korrekte Codierung [13].
Ein weiteres Problem ist die Zeit, die eine *act4teams*-Analyse beansprucht [24]. Des Weiteren kann die zeitaufwändige Analyse aufgrund des komplexen Codierschemas nur von einem Experten durchgeführt werden [24].
Diese Gründe veranlassten eine Ableitung des komplexen Codieschemas von *act4teams* zu dem leichtgewichtigeren Konzept *act4teams-SHORT* [13]. Mit dem Verfahren ist es möglich, dass auch Softwareentwickler eine erleichterte Meetinganalyse durchführen können [13]. Auf eine Videoaufnahme wird verzichtet, indem die Interaktionsanalyse in Echtzeit während des Meetings stattfindet [13].
Klünder [13] untersuchte das *act4teams-SHORT*-Konzept mit unerfahrenen Meetinganalysten als Probanden in einem Experiment. Diese codierten ein Meeting anhand elf Kategorien des Codierschemas. Die Untersuchung ergab unter anderem, dass die subjektive Schwierigkeit der Differenzierung zwischen den Kategorien des Codierschemas über die Zeit etwas abnahm, jedoch die Unterscheidung den Probanden auch gegen Ende nicht leicht fiel.

1.2 Zielsetzung

Um die Schwierigkeiten bei der anfänglichen Nutzung der *act4teams-SHORT*-Meetinganalyse zu reduzieren, soll vor der ersten Meetinganalyse ein Tutorial zum Einsatz kommen. Dieses soll helfen Begrifflichkeiten zu verstehen und eine Entscheidungsbasis aufzubauen. Dafür werden im Rahmen dieser Arbeit zwei Tutorialkonzepte herausgearbeitet. Diese bestehen aus unterschiedlichen Vorgehensweisen und verfolgen verschiedene Lernstrategien. Eines der Tutorials beinhaltet einen auditiven Meetingausschnitt mit dargelegter Referenzcodierung, welche vom Nutzer nachvollzogen werden kann. Das andere Tutorial beinhaltet ein Quiz, in dem Beispielaussagen zu jeweils einer Kategorie einsortiert werden sollen. Beide werden in die bereits bestehende *act4teams-SHORT*-Anwendung integriert und sollen auf ihre Effektivität untersucht werden.

Aus dieser Motivation beschäftigt sich die Arbeit mit der Beantwortung folgender Forschungsfragen:

Forschungsfrage 1

Welche Art von Tutorials eignen sich, um Schwierigkeiten bei der anfänglichen Nutzung der *act4teams-SHORT*-Analyse zu reduzieren?

Diese Frage untersucht, welche Vorgehensweise zur Präsentation der Lerninhalte sich für die Integration in ein Tutorial eignet, damit dadurch ein potenzieller Lerneffekt auftritt und somit die anfänglichen Schwierigkeiten bei der Nutzung der *act4teams-SHORT*-Analyse reduziert werden.

Forschungsfrage 2

Wie kann ein Tutorial vor der ersten Interaktionsanalyse die resultierende Codierung eines unerfahrenen Beobachters beeinflussen?

Diese Frage untersucht, ob eines der Tutorials einen Einfluss auf die Übereinstimmung zwischen unerfahrenen Meetinganalysten und der Referenzcodierung hat. Dabei sollen Nutzer, die zuvor ein Tutorial durchgeführt haben, idealerweise bessere Meetinganalysen erzeugen, als Nutzer die kein Tutorial durchführten.

Forschungsfrage 3

Welches der beiden Tutorials ist besser in Bezug auf den Lerneffekt und das subjektive Empfinden des Nutzers?

Nicht nur die Frage, ob ein Tutorial einen messbaren Einfluss auf eine Meetinganalyse hat, sondern wenn ja, welches der Tutorials höhere Effekte aufweist, ist ein zu untersuchender Aspekt dieser Arbeit. Bei dem Vergleich beider Tutorials spielt auch das subjektive Empfinden der Nutzer eine Rolle.
Um diese Fragen beantworten zu können, wurde im Rahmen dieser Arbeit ein Experiment durchgeführt.

1.3 Struktur der Arbeit

Zu Beginn werden in Kapitel 2 einige Grundlagen, die für das Verständnis dieser Arbeit benötigt werden, erklärt. Darunter fällt das Verfahren *act4teams* und das daraus abgeleitete Verfahren *act4teams-SHORT*. Des Weiteren werden Lernstrategien erklärt und das Thema Usability behandelt. Kapitel 3 gibt Aufschluss über die verwandten Arbeiten. Kapitel 4 beschäftigt sich mit dem Konzept und der Implementierung der Lernverfahren. Dafür wird zuerst darauf eingegangen, woher die Ideen für die Konzeptualisierung stammen (Kapitel 4.1). Danach wird die zu integrierende Meetingaufnahme beschrieben und im Anschluss auf die Implementierung eingegangen (Kapitel 4.2 und Kapitel 4.3).

Kapitel 5 beschreibt die Evaluation. Dafür werden zu Beginn die Ziele und Hypothesen aufgezeigt (Kapitel 5.1). Danach wird auf die Planung des Experimentes eingegangen, indem Meetingausschnitte, Experimentsubjekte und das Design des Experimentes vorgestellt werden (Kapitel 5.2). Im Anschluss folgt die Durchführung des Experimentes (Kapitel 5.3). Es wird die Experimentumgebung und der Ablauf des Experimentes beschrieben und der Zeitraum und die Dauer aufgeführt. Als nächstes schließen die Analyse und Interpretation an (Kapitel 5.4). Dazu werden zunächst die Probanden anhand von deskriptiven Statistiken beschrieben. Danach folgt die Datenaufbereitung und -analyse. Folgend werden die Ergebnisse der formal geprüften Hypothesen vorgestellt. Als nächstes wird die Validität diskutiert (Kapitel 5.5). Das Kapitel schließt mit einer Zusammenfassung und Implikationen für die Anwendung der Tutorials (Kapitel 5.6 und Kapitel 5.7). Das Kapitel 6 beinhaltet eine Zusammenfassung der Arbeit und präsentiert einen Ausblick auf Möglichkeiten für anknüpfende Studien.

2 Grundlagen

In diesem Kapitel werden Grundlagen eingeführt, die für das Verständnis dieser Arbeit benötigt werden. Dazu wird zuerst auf das Verfahren *act4teams* und das daraus abgeleitete Verfahren *act4teams-SHORT* eingegangen. Danach werden Lernstrategien aufgezeigt und Usability-Aspekte und Designrichtlinien erläutert. Zuletzt werden noch die Instrumente *Personas* und *Papierprototypen* beschrieben.

2.1 Act4teams

Act4teams ist ein prozessanalytisches Instrument, welches wissenschaftlich fundiert Kompetenzen von Führungskräften und Teams misst [12]. Dabei steht die Bezeichnung *act4teams* für 'advanced interaction analysis for teams' [12].
Bei *act4teams* handelt es sich um eine strukturierte Methode zur Interaktionsanalyse von Meetings, die am Lehrstuhl für Arbeits-, Organisations- und Sozialpsycholgie an der Technischen Universität Braunschweig entwickelt wurde [13].
Die Vorgehensweise beinhaltet das Aufzeichnen einer regulären Teambesprechung auf Video. Dieses wird im Anschluss zuerst transkribiert und im nächsten Schritt von einem Psychologen analysiert [11], [12]. Dazu werden die verbalen Äußerungen der einzelnen Teammitglieder in Sinneinheiten unterteilt [11]. Eine Sinneinheit kann dabei ein einzelnes Wort, ein Teilsatz oder auch ein vollständiger Satz sein [13]. Darauf folgend wird jede Sinneinheit mit einem Code versehen, dazu stehen dem Analysten eine Auswahl von 44 Codes zur Verfügung [11].

2.1.1 Codierschema

Die 44 Kategorien des Codierschemas resultieren aus zwölf Kompetenzaspekten. Diese wiederum lassen sich zu vier Kompetenzbereichen vereinen. Eine Übersicht über die Struktur kann Tabelle 2.1 entnommen werden. Die vier Kompetenzbereiche werden im Folgenden vorgestellt [11]:

Kompetenzbereiche	*Kompetenzaspekte*	*Kategorien*
Problem-fokussierte Aussagen	Problemdifferenzierung	Problem
		Problemerläuterung
	Problemvernetzung	Verknüpfung bei der Problemanalyse
	Lösungsdifferenzierung	Sollentwurf
		Lösungsvorschlag
		Lösungserläuterung
	Lösungsvernetzung	Problem zur Lösung
		Verknüpfung mit Lösung
	Wissen zur Organisation	Organisationales Wissen
	Wissensmanagement	Wissen wer
		Frage
Prozedurale Aussagen	Positive prozedurale Aussagen (Strukturierung)	Zielorientierung
		Klärung/Konkretisierung
		Verfahrensvorschlag
		Verfahrensfrage
		Priorisieren
		Zeitmanagement
		Aufgabenverteilung
		Visualisierung
		Kosten-Nutzen-Abwägung
		Zusammenfassung
	Negative prozedurale Aussagen	Verlieren in Details und Beispielen
Sozio-emotionale Aussagen	Positive sozio-emotionale Aussagen (Kollegiale Interaktion)	Ermunternde Ansprache
		Unterstützung
		Aktives Zuhören
		Ablehnung
		Rückmeldung
		Atmosphärische Auflockerung
		Ich-Botschaft: Trennung von Meinung und Tatsache
		Gefühle
		Lob
	Negative sozio-emotionale Aussagen (Unkollegiale Interaktion)	Tadel/Abwertung
		Unterbrechung
		Seitengespräch
		Reputation
Aktionsorientierte Aussagen	Proaktives Verhalten	Interesse an Veränderungen
		Eigenverantwortung
		Maßnahmenplanung
	Destruktives Verhalten	Kein Interesse an Veränderungen
		Jammern
		Abbruch
		Schuldigensuche
		Betonung autoritärer Elemente
		Phrase

Tabelle 2.1: Zusammenhang zwischen den Kompetenzbereichen und -aspekten und Kategorien von *act4teams* [4], nach [11]

Problem-fokussierte Aussagen. Dazu zählen Nennung und Beschreibung von Problemen und Lösungen sowie das Ableiten von Ursache und Wirkung von Problemen, genauso wie das Verknüpfen von Problemen mit Lösungen. Des Weiteren sind auch Wissen zur Organisation und das Wissensmanagement inbegriffen.

Prozedurale Aussagen. Zum einen gibt es die positiven prozeduralen Aussagen, die unter anderem eine Diskussion strukturieren und organisieren. Zum anderen gibt es die negativen prozeduralen Aussagen, die sich durch lange Monologe einzelner Teammitglieder oder in nichtzielführenden Diskussionen äußern.

Sozio-emotionale Aussagen. Diese sind ebenfalls in positiv und negativ aufgeteilt. Positiv sozio-emotionale Aussagen sind Aussagen, die Kollegen einbeziehen oder unterstützen, sowie aktives Zuhören und das Aussprechen von Lob. Negative sozio-emotionale Aussagen beinhalten Tadel und Abwertung von Kollegen sowie das Führen von Seitengesprächen.

Aktionsorientierte Aussagen. Dabei wird zwischen proaktiven und destruktiven Verhalten unterschieden. Proaktives Verhalten zeigt sich in der Bereitschaft des Teams, Maßnahmen zu ergreifen, um ihre Arbeit zu verbessern. Des Weiteren zeigt es sich durch das Zeigen von Interesse an Veränderungen und von Eigenverantwortung. Destruktives Verhalten wird durch Jammern, Schuldigensuche und Desinteresse an Veränderungen geäußert.

2.1.2 Interpretation einer Meetinganalyse

Die *act4teams*-Analyse gibt Aufschluss über drei wichtige Aspekte für den Projekterfolg. Die Aspekte lauten Meetingzufriedenheit, Teamproduktiviät und organisatorischer Erfolg [11]. Im Weiteren werden diese als Meetingresultate bezeichnet.
Dabei beeinflusst jede codierte Sinneinheit der Kategorie entsprechend mindestens eines dieser drei Meetingresultate [11]. Signifikante Auswirkungen der Kategorien auf die Meetingresultate werden anhand der übergeordneten Kompetenzaspekte in Tabelle 2.2 dargestellt.

Festzustellen ist, dass sich negative Aussagen negativ auf die Ergebnisse auswirken. Die problem-fokussierten Aussagen sowie die positiven Aussagen, ausgenommen der sozio-emotionalen Aussagen, wirken positiv auf die Meetingergebnisse.

Anhand dieser nachgewiesenen Zusammenhänge zwischen den Kategorien und dem Meetingerfolg werden mit dem Unternehmen gemeinsam Maßnahmen zur Teamentwicklung abgeleitet [11]. Handlungsfelder werden aufgezeigt, die in Relation mit der Produktivität und dem Erfolg eines Unternehmens stehen [11]. Derart können wirkungs-

Kompetenzaspekte von *act4teams*	Meeting-zufriedenheit	Teamproduktivität	Organisatorischer Erfolg
Problemdifferenzierung		+	
Problemvernetzung	+		+
Lösungsdifferenzierung	+	+	+
Lösungsvernetzung	+	+	+
Wissen zur Organisation		+	
Wissensmanagement		+	
Positive prozedurale Aussagen	+		+
Negative prozedurale Aussagen	-	-	-
Positive sozio-emotionale Aussagen	-		
Negative sozio-emotionale Aussagen	-		-
Proaktives Verhalten	+	+	+
Destruktives Verhalten	-		-

Tabelle 2.2: Einflüsse der *act4teams* Kategorien auf die Meetingergebnisse [4], nach [11]

volle Entwicklungsmaßnahmen abgeleitet werden [11].

2.1.3 Anwendbarkeit auf Softwaremeetings

Act4teams ist ein in der Psychologie etabliertes Verfahren zur Interaktionsanalyse. Jedoch kann dieses Verfahren im Bereich der Softwareentwicklung nicht uneingeschränkt eingesetzt werden [13], [25]. Folgende Probleme wurden dabei herausgearbeitet [13], [25]:

Das Codierschema. *Act4teams* ist darauf ausgelegt Meetings zu analysieren, die zur Verbesserung von Produktionsprozessen dienen. Softwareprojekt-Meetings hingegen dienen überwiegen der Planung und des Informationsaustausches. Somit ist das bestehende Codierschema nicht optimal, um detaillierte Aussagen über die Güte von Softwareprojekt-Meetings treffen zu können.

Die (datenschutz-)rechtliche Grundlage. Aufgrund von Datenschutzbestimmungen von Unternehmen ist die Videoaufnahme meist nicht möglich. Die Aufnahme ist für die Interaktionsanalyse jedoch unerlässlich.

Das Hintergrundwissen. Eine Schulung im Umfang von 200 Zeitstunden ist erforderlich, damit vom Beobachter unabhängige Codierungen erzeugt werden.

Der Zeitaufwand. Pro Stunde Videomaterial dauert die Codierung meist mehr als acht Stunden. Eine schlechte Qualität einer Aufnahme erhöht den Zeitaufwand zusätzlich.

Unter den genannten Punkten ist folglich eine Interaktionsanalyse eines Softwareprojekt-Meetings nicht realisierbar [13]. Um den Einsatz dennoch zu ermöglichen, bestand die Notwendigkeit, das Konzept unter Berücksichtigung der aufgeführten Schwierigkeiten anzupassen [13]. Im Folgenden wird das für die Softwareentwicklung optimierte Verfahren *act4teams-SHORT* vorgestellt [13].

2.2 Act4teams-SHORT

Act4teams-SHORT ist ein von *act4teams* abgeleitetes Verfahren, welches durch eine erhebliche **Komplexitätsminderung des Codierschemas** eine vereinfachte Interaktionsanalyse von Softwareprojekt-Meetings ermöglicht [13].
Die Komplexitätsreduzierung ermöglicht die Beseitigung der beschriebenen Schwierigkeiten der Anwendung bei Softwareprojekt-Meetings [13]:

Die (datenschutz-)rechtliche Grundlage. Die vereinfachte Analyse beseitigt die Notwendigkeit einer Videoaufnahme des Meetings, da die Interaktionsanalyse in Echtzeit während des Meetings vollzogen werden kann. Datenschutzbestimmungen der Unternehmen sind daher für die Durchführung einer Interaktionsanalyse nicht mehr von Interesse.

Das Hintergrundwissen. Auch die für *act4teams* erforderliche Schulung im Umfang von 200 Stunden muss aufgrund der Komplexitätsminderung des Codierschemas nicht mehr vollzogen werden. Das Hintergrundwissen für das *act4teams-SHORT*-Verfahren soll dem unerfahrenen Analysten stattdessen anhand eines interaktiven Tutorials beigebracht werden. Dadurch kann auch ein Software Engineer ohne aufwändige Einarbeitungszeit die Meetinganalyse durchführen.

Der Zeitaufwand. Durch die Analyse während des Meetings in Echtzeit entfällt die aufwändige Codierung des auf Video aufgenommen Meetings im Nachhinein.

2.2.1 Entwicklung

Mehrere Iterationen durchlief *act4teams-SHORT* bis zu dessen aktueller Version [13]. Der Inhalt der Iterationen ist dargestellt in Tabelle 2.3.

Iteration	Inhalt
Iteration 1	Definition des Kategorieschemas
Iteration 2	Implementierung des Tools
Iteration 3	Anpassung des Kategorieschemas
Iteration 4	Anpassung des Kategorieschemas

Tabelle 2.3: Inhalt der Iterationen (in Anlehnung an [13])

In der ersten Iteration erfolgte die Ableitung des *act4teams*-Schemas zum *act4teams-SHORT*-Schema unter Berücksichtigung der am häufigsten auftretenden Kategorien in Meetings sowie des zeitlichen Anteils der Kategorien in Meetings und der Relevanz der Kategorien nach Einschätzung von *act4teams*-Experten [13]. Weitere Anpassungen am Kategorienschema sowie die Implementierung in eine Anwendung und die Weiterentwicklung dieser folgten in den zwei darauffolgenden Iterationen [4], [13], [23]. Aktuell befindet sich *act4teams-SHORT* in der vierten Iteration und umfasst neun Kategorien.

2.2.2 Codierschema

Das *act4teams-SHORT*-Codierschema umfasst die folgenden neun Kategorien:

- Probleme: Die Kategorie umfasst unter anderem das Benennen und Erläutern von Problemen [4], [13].
- Lösungen: Darunter fällt die Nennung und Ausarbeitung sowie das Sammeln von Lösungsvorschlägen [4], [13].
- Verknüpfungen und Vernetzungen: Diese Kategorie enthält das Darstellen von Zusammenhängen zwischen Problemen und Lösungen/Vorschlägen [4], [13].
- Destruktives Verhalten: Dieses umfasst unter anderem das Tadeln oder Lästern einer Person, Schuldigensuche, Jammern und das Zeigen von keinem Interesse an Veränderungen [4], [13].
- Proaktives Verhalten: Dieses macht sich bemerkbar zum Beispiel in dem Zeigen von Engagement für Veränderungen oder im Planen von konkreten Maßnahmen [4], [13].
- Kollegiales Verhalten: Dieses äußert sich beispielsweise in ermunternden Ansprachen, aktivem Zuhören oder in unterstützenden Aussagen [13].
- Methodisch-strukturierendes Verhalten: Diese Kategorie umfasst unter anderem das Verweisen auf Ziele, Setzen von Prioritäten und das Zusammenfassen von

Aufgaben [4], [13].

- Informationsweitergabe: Darunter fällt das Informieren anderer Meetingteilnehmer über Fakten und Ereignisse [4], [13].
- Sonstiges: Sonstiges kann gewählt werden, wenn keine andere Kategorie zutrifft.

2.2.3 Das Tool und dessen Anwendung

Im Folgenden werden die Oberfläche, die Interaktionsmöglichkeiten und der Analyseablauf des *act4teams-SHORT*-Tools beschrieben.
Das *act4teams-SHORT*-Tool wurde im Rahmen einer Abschlussarbeit entwickelt und darauf folgend iterativ erweitert und angepasst (vgl. Kapitel 2.2.1) [4], [13], [23]. Die Anwendung wurde im Speziellen für ein Microsoft Surface Pro 4 erstellt, da dieses durch eine gute Handhabung und eine Touchbedienung die Analyse während eines Meetings einfach und ohne Geräuscherzeugung ermöglicht. Der Analyst des Meetings ist selbst kein Teil der Besprechung, sondern ausschließlich mit der Interaktionsanalyse beschäftigt. Diese erfordet von dem Meetingbeobachter die ungeteilte Aufmerksamkeit, damit eine korrekte Codierung erfolgen kann.

Abbildung 2.1: Menü

Abbildung 2.1 zeigt das Menü. Vor der Meetinganalyse wird hier eine Meetingart ausgewählt. Falls kein passender Vorschlag dabei ist, kann der User selbst eine Meetingart erstellen. Desweiteren besteht die Möglichkeit den Speicherort für die csv-Datei, in der die im Meeting erfassten Tags gespeichert werden, zu ändern.

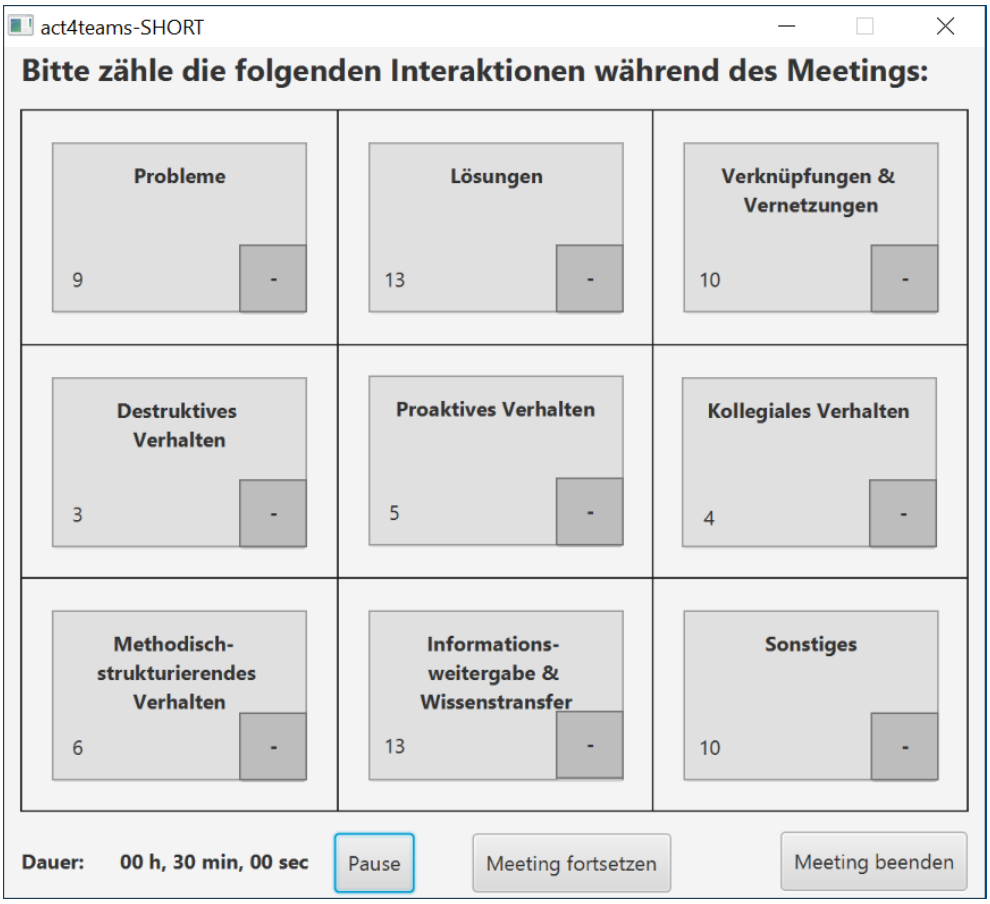

Abbildung 2.2: Oberfläche der Datenerhebung

Abbidung 2.2 zeigt die Oberfläche, mit der der Analyst während des Meetings die Tags erfasst. Alle Kategorien des *act4teams-SHORT*-Schemas sind enthalten. Hat der Analyst eine Aussage vernommen und einer der Kategorien zugeordnet, klickt er den mit der Kategoriebezeichnung versehenen hellgrauen Button an. Daraufhin wird der entsprechende Zähler um eins inkrementiert und die Kategorie plus Zeitstempel werden programmintern abgespeichert. Ein falsch gesetzter Tag kann mithilfe des kleinen dunkelgrauen Buttons, auf dem ein Minus zu erkennen ist, zurückgenommen werden. Daraufhin wird der Zähler um eins dekrementiert und das Vorkommen der Kategorie intern gelöscht, sodass dieses in der resultierenden csv-Datei nicht aufgelistet wird.
Ist das Meeting beendet und die Analyse abgeschlossen, klickt der Nutzer auf *Meeting beenden*. Daraufhin wird die csv-Datei erstellt, die alle Vorkommen erfasster Kategorien mit Zeitstempel und eine Übersicht über die Gesamtanzahl einzelner Kategorien beinhaltet.

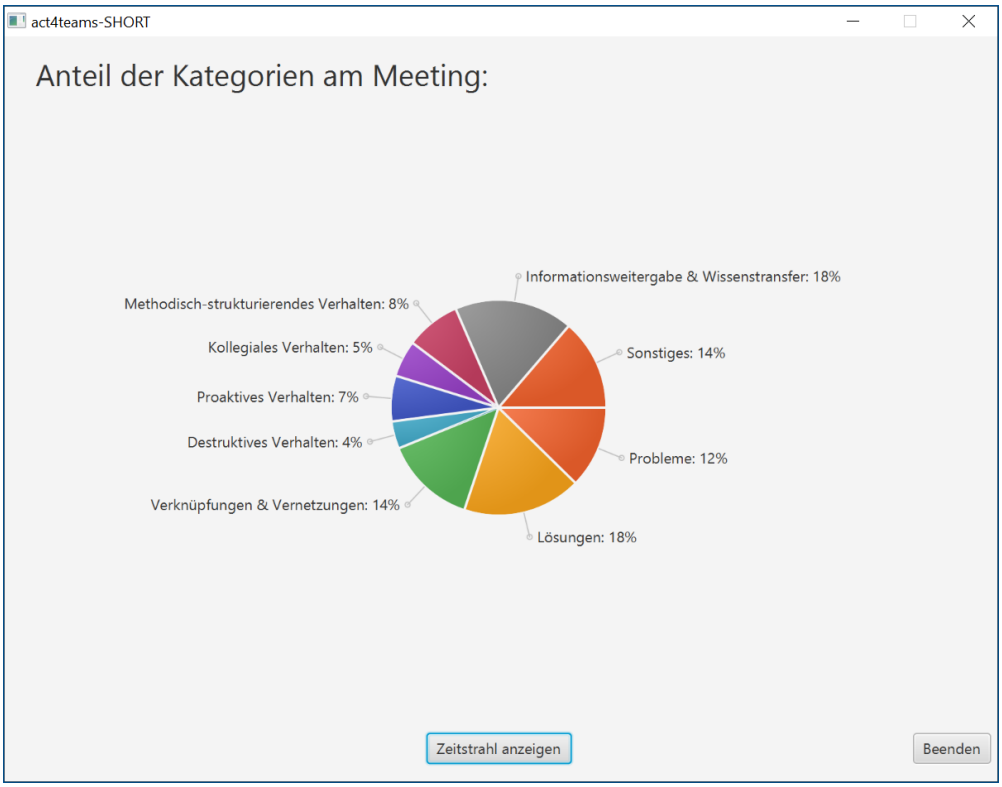

Abbildung 2.3: Kreisdiagramm einer Beispielanalyse

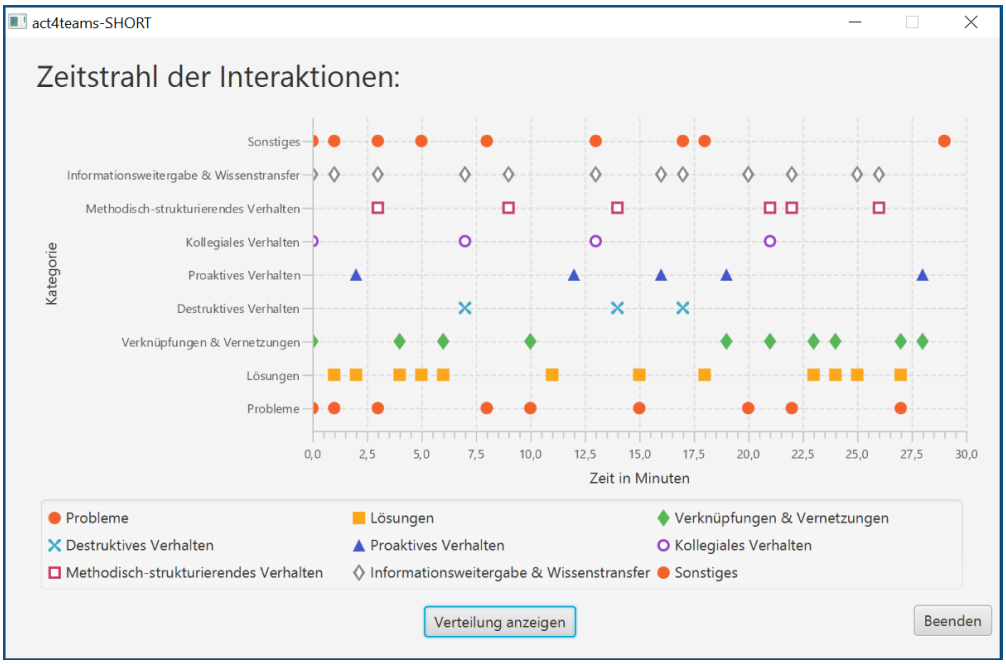

Abbildung 2.4: Zeitstrahl einer Beispielanalyse

Als nächstes stehen dem Meetingbeobachter zwei Diagrammarten als Interpretationshilfe zur Verfügung. Dabei handelt es sich zum einen um ein Kreisdiagramm, beispielhaft dargestellt in Abbildung 2.3. Dieses zeigt den prozentualen Anteil der einzelnen Kategorien im Verhältnis zur Gesamtanzahl aller Kategorien an. Zum anderen kann der Nutzer den Zeitstrahl, beispielhaft dargestellt in Abbildung 2.4, begutachten. Dieser zeigt jeden Tag über die Zeit an.

2.2.4 Herausforderung der Analyse

Die Analyse fordert vom Beobachter, die Bedeutungen der Kategorien und insbesondere die exakte Differenzierung zwischen diesen zu kennen. Bei unerfahrenen Beobachtern kann dies große Schwierigkeiten hervorrufen. Klünder [13] hat im Rahmen ihrer Dissertation ein Experiment durchgeführt, in dem zwölf unerfahrene Meetingbeobachter mit dem *act4teams-SHORT*-Tool in der dritten Iteration ein Meeting codierten. Diese bekamen im Vorfeld eine kurze Einführung in die zu unterscheidenden Kategorien. Ziel war es unter anderem herauszufinden, ob es unerfahrenen Meetingbeobachtern nach deren subjektiver Wahrnehmung im Laufe eines Meetings einfacher fällt, zwischen den Kategorien zu unterscheiden. Daher wurde den Probanden nach 10 und nach 20 Minuten der Analyse ein Fragebogen ausgehändigt. Auf diesem sollten die Probanden ihr subjektives Verständnis in Bezug zu den Kategorien angeben. Dazu gaben die Probanden unter anderem an, ob ihnen die Unterscheidung zwischen den Kategorien leicht fiel. Dabei stellte sich heraus, dass den Probanden die Unterscheidung zwischen den Kategorien im zweiten Abschnitt etwas leichter fiel als im ersten Abschnitt. Dennoch gaben die Probanden auch nach 20-minütiger Analyse an, dass die Unsicherheit in Bezug auf die Wahl einer Kategorie immer noch groß sei.
Um diese Schwierigkeiten zu reduzieren, soll eine weiterführende Einweisung in Form zweier interaktiver Tutorials in die Anwendung *act4teams-SHORT* integriert werden.

2.3 Lernstrategien

Das Hauptziel eines Tutorials ist es, Wissen über einen speziellen Sachverhalt zu vermitteln. Ein Tutorial kann mehrere Lernstrategien erfassen, die dann auf die Nutzer angewendet werden. Lernstrategien sind zielgerichtete kognitive Operationen, dienen also dem kognitiven Einprägen und sind potenziell bewusste und kontrollierbare Aktivitäten, die den Lernenden dabei helfen sollen, eine Lernanforderung zu bewältigen [6].

Die kognitiven Lernstrategien lassen sich in Bezug auf ihre Funktion folgendermaßen unterscheiden [3], [6], [8]:

Wiederholungsstrategie (Mnemotechniken). Bei dieser Strategie ist das Hauptziel das Einprägen von isolierten Fakten. Dabei soll idealerweise ein Transfer der Informationen ins Langzeitgedächtnis stattfinden.
Beispiel: einfaches Wiederholen

Organisationsstrategie (strukturierende Strategien). Diese Strategie dient dazu, Informationen auf das Wesentliche zu reduzieren. Die Informationen werden also mit dem Ziel organisiert, sie in verständlicher Weise aufeinander in Bezug zu setzen.
Beispiel: Zusammenfassung

Elaborationsstrategien (generative Strategien). Diese Strategien zielen darauf, ein tieferes Verständnis des Lernstoffs zu erreichen, indem neue Informationen mit dem Vorwissen verknüpft werden.
Beispiel: praktische Beispiele

2.4 Usability

Damit der Nutzer uneingeschränkt und in optimaler Weise den Inhalt der Tutorials aufnehmen kann, ist Usability ein wichtiger Faktor, der beim Designprozess berücksichtigt werden muss. Der Begriff Usability steht dafür, wie gut Nutzer die Funktionalität eines Systems anwenden können [21]. Dabei ist Usability keine einzelne Eigenschaft einer Benutzeroberfläche, sondern besteht aus mehreren Komponenten. Klassischerweise wird Usability mit den folgenden fünf Aspekten assoziiert [21]:
Learnability. *Das System sollte leicht zu erlernen sein, damit der Benutzer schnell mit der Interaktion beginnen kann.*
Efficiency. *Das System sollte effizient zu bedienen sein, damit eine hohe Produktivität möglich ist, sobald der Nutzer das System verstanden hat.*
Memorability. *Das System sollte einfach zu merken sein, damit ein Nutzer, der eine längere Zeit dieses nicht verwendet hat, das System verwenden kann, ohne es wieder komplett neu lernen zu müssen.*
Errors. *Das System sollte eine geringe Fehlerquote haben, sodass User nur wenig Fehler bei der Nutzung machen. Wenn ein Fehler hervorgerufen wurde, soll dieser einfach rückgängig zu machen sein. Des Weiteren dürfen Fehler mit weitreichenden Folgen nicht auftreten.*
Satisfaction. *Das System sollte angenehm zu nutzen sein, damit User bei der Verwendung subjektiv zufrieden sind. Nutzern sollten die Anwendung gefallen.*

Bei der Anwendung von Tutorials spielen besonders die Aspekte *Learnability*, *Efficiency* und *Satisfaction* eine übergeordnete Rolle. *Learnability* ist ein wichtiger Aspekt der Tutorials, da der Fokus des Lernens auf dem Inhalt liegen soll und nicht auf der Handhabung der Tutorials. Des Weiteren ist eine effiziente Oberfläche sehr wichtig, damit die Nutzer präzise und schnell mit den Tutorials interagieren können und nicht durch unnötig lange Interaktionsabfolgen abgelenkt oder aufgehalten werden. Zudem soll die Nutzung der Tutorials den Usern gefallen, damit diese von den Anwendern akzeptiert werden. Die Fehlerquote ist nicht unbedingt ein Aspekt, der bei einem Tutorial gering gehalten werden muss. Denn es ist möglich, dass es Teil einer Lernsoftware ist, Fehler zu machen, um aus ihnen im Anschluss etwas lernen zu können.

Es gibt einige Designrichtlinien, die sich nachweislich positiv oder auch negativ auf die Benutzbarkeit einer Anwendung auswirken [21]. Im Folgenden werden einige relevante beschrieben [21]:
Bei der Konzeption eines Interfaces gilt es, dieses so **einfach** wie möglich zu gestalten. Jedes zusätzliche Element und jede zusätzliche Information, die gelernt und richtig verstanden werden muss, birgt die potenzielle Gefahr falsch interpretiert zu werden. Ein Interface gilt als benutzerfreundlicher, wenn die Informationen, die für das Erledigen der Aufgabe notwendig sind, ausschließlich auf das Wesentliche reduziert werden. Dies ist vor allem wichtig, da jede auf dem Screen enthaltene Information vom User, während er sich durch das Interface navigiert, betrachtet wird. Das hat zur Folge, dass bei steigenden Elementen die Performanz sinkt. Eine Möglichkeit, die Performanz zu steigern, ist es, Abfolgen von mehreren Eingaben, die von Usern häufig getätigt werden, in **Shortcuts** zusammenzufassen. Dem Nutzer bleibt somit die Durchführung immer gleicher Abfolgesequenzen erspart und der Zeitaufwand wird reduziert. Weiterhin ist es vorteilhaft, wenn die Aufgabe, die das Interface bereithält, dem Nutzer so **natürlich** wie möglich präsentiert wird. Dies sollte auf eine Weise geschehen, in der die Computerelemente die natürlichen Konzepte des Users widerspiegeln und diese somit intuitiv verständlich sind. Des Weiteren gilt es auch, eine natürliche Sprache zu verwenden und bestenfalls auf jegliche Codes zu verzichten. Noch dazu wäre es von Vorteil, wenn das Interface die Muttersprache des jeweiligen Anwenders verwendet.
Wenn es um das visuelle Design eines Interfaces geht, gibt es ebenfalls einige Richtlinien zu beachten. Die **Zusammengehörigkeit** von Elementen kann visualisiert werden, indem diese dicht beieinander liegen, sich synchron bewegen oder ändern, durch Linien oder eine Box eingeschlossen sind oder sich durch ihre Form, Farbe und Größe ähneln. Eine **Priorisierung** von Elementen kann durch deren Größe, eine auffällige Erscheinung oder anhand der natürlichen Leserichtung erfolgen. Zur Priorisierung

sparsam einzusetzen ist beispielsweise die Darstellung von Wörtern oder Sätzen in Großbuchstaben, da diese zehn Prozent langsamer gelesen werden als normaler Text. **Farben** können auch genutzt werden, um Elemente zu kategorisieren, differenzieren und hervorzuheben. Sie sollten jedoch nicht eingesetzt werden, um Informationen zu vermitteln, da es einige Menschen gibt, die unter einer Form von Farbenfehlsichtigkeit leiden. Daher muss sichergestellt werden, dass das Interface auch ohne Farben nutzbar ist.
Des Weiteren ist ein sehr wichtiges Usability-Prinzip die **Konsistenz**. Gleiche Aktionen oder gleiche Kommandos sollen stets den gleichen Effekt hervorrufen. Hinzu kommt, dass Informationen, die auf mehreren Screens vorhanden sind, jeweils an derselben Stelle in derselben Darstellungsform vorhanden sein sollten. Das erleichtert die Wiedererkennung des Nutzers. Konsistenz sollte nicht nur das Design beinhalten, sondern auch die Aufgabe und die Struktur der Funktionalität, um die Aufgabe für den Nutzer verständlicher zu machen. Wichtig ist es auch, darauf zu achten, dass die Oberflächenelemente die User Task Expectation erfüllen. Das bedeutet, dass ein Element sich nach einer Useraktion so verhält, wie der User es von anderen Programmen bereits kennt.
Ein weiteres Usability-Prinzip von Wichtigkeit ist das **Feedback**. Der User sollte stets den Überblick behalten, an welcher Stelle des Programms er sich befindet. Noch dazu sollte er nach einer Interaktion direktes Feedback vom System erhalten, wie sich die jeweilige Eingabe auf das Programm ausgewirkt hat. Es ist zudem wichtig, dass die Implementation der Anwendung so effizient wie möglich gestaltet wird, um Feedback- und Antwortzeiten des Programms so kurz wie möglich zuhalten. Dabei gilt, wenn das Feedback in weniger als 0,1 Sekunden auf dem Screen zu sehen ist, dass der Nutzer das Gefühl hat, sofortiges Feedback erhalten zu haben. Bei einer Antwortzeit zwischen 0,1 und einer Sekunde spürt der User eine kurze Verzögerung, fühlt sich aber durch diese nicht gestört. Das Limit des Nutzers, seine Aufmerksamkeit dem Warten auf Feedback zu überlassen, ist auf zehn Sekunden begrenzt. Längere Wartezeiten führen dazu, dass der User sich anderen Aufgaben am Rechner widmet. Das System sollte somit grundsätzlich so schnell wie möglich auf die Eingaben des Nutzers reagieren. Es gibt aber dennoch Ausnahmen, in denen die Anwendung möglicherweise zu schnell reagiert, sodass der User Schwierigkeiten hat, dem Feedback zu folgen. Ein Beispiel ist das Scrollen. Wenn dieses zu schnell fortschreitet, wird es dem Nutzer schwer möglich gemacht, an gewünschter Stelle anzuhalten, und der Stopp tritt erst an späterer Position ein. Bei solchem Verhalten soll an Stelle der Ausführungsgeschwindigkeit des jeweiligen Computers eine Real-Time-Uhr als Referenz gelten.
Ein letzter wichtiger Punkt ist auch die **Help-and-Documentation**-Aufbereitung. Gerade wenn es sich um neue Anwender eines Tools handelt, ist ein Help-System an der einen oder anderen Stelle angebracht. Der Versuch, im Vornherein mit längeren

Help-Seiten den Umgang mit dem System zu erklären, scheitert häufig, da User diese nicht bewusst und aufmerksam lesen oder im schlimmsten Fall überspringen. Realisiert durch Tooltips werden an direkter Stelle Hinweise zur Interaktion gegeben. Diese haben den Vorteil kontextsensitiv zu sein, indem sie passend zur Situation die richtige Hilfestellung bereitstellen. Dabei sollte darauf geachtet werden, dem User die Reihenfolge der Kommandoausführungen in treffender Weise zu vermittelt.

2.4.1 Personas

Zwei allgemeine Prinzipien, die den meisten Software Guidelines zugrunde liegen, sind *know the system* und *know the user* [1]. Das System *act4teams-SHORT* wurde zuvor bereits erläutert (vgl. Kapitel 2.2). *Know the user* bedeutet, den intendierten Nutzer zu bestimmen und zu analysieren [21]. Dies ist sehr wichtig, da individuelle Benutzereigenschaften und variable Tätigkeitsfelder die beiden Faktoren sind, die den größten Einfluss auf die Usability haben [21]. Durch die Kenntnis der Berufserfahrung, des Bildungsniveaus, des Alters und der bisherigen Erfahrung im Umgang mit Computern der Nutzer ist es möglich, in einem gewissen Umfang die Lernschwierigkeiten zu antizipieren und geeignete Grenzen für die Komplexität der Benutzeroberfläche festzulegen [21].

Eine Vorgehensweise, die die Beteiligung zukünftiger Nutzer am Designprozess gewährleistet, auch wenn diese, beispielsweise durch ihre schiere Anzahl, nicht alle direkt an der Erstellung teilhaben können, ist die Erstellung von Personas. Personas sind prototypische Nutzerprofile, die stellvertretend für eine Anwendergruppe stehen [10]. Eine Persona vereinigt die typischen Eigenschaften von Nutzern einer Gruppe und deren Verhaltensweisen und Motive in Bezug auf die entsprechende Anwendung in kurzer, prägnanter Weise [10]. Sie dienen dazu, die Anforderungen und Bedürfnisse der Nutzer herauszufinden und diese während der Entwicklung der jeweiligen Software zu berücksichtigen [10]. Eine Persona sollte folgende Informationen mit sich bringen [10]:
soziologische Angaben, persönliche Eigenschaften und Merkmale des Nutzers, die Zielsetzung/Intention und Aufgaben bei der Nutzung der Anwendung, die Vorerfahrung und Kenntnisse hinsichtlich der betreffenden Thematik.
Dabei gilt, je detaillierter die Beschreibung der Persona ist, desto einheitlicher ist die Vorstellung der Persona bei allen Beteiligten [10]. Jedoch sollte darauf geachtet werden, dass die Personas im Großen und Ganzen übersichtlich bleiben und lange Texte vermieden werden [10].

2.4.2 Papierprototypen

Wurden die Benutzeranforderungen schließlich herausgearbeitet und Ideen für ein Konzept entwickelt, empfiehlt es sich, dieses zuerst anhand von Prototypen zu visualisieren. Diese sind hilfreich, um Ideen anderen gegenüber zu präsentieren, die grobe Funktionsfähigkeit darzustellen oder den Entwurf von Nutzern testen zu lassen [10]. Dafür eignen sich besonders gut Papierprototypen, da sich diese mittels Bleistift und Papier einfach, schnell und ohne Einsatz von elektronischen Tools erstellen lassen. Die Beschreibung, Anwendung und Vorteile von Papierprototyp werden folgend aufgeführt [10]:
Ein Papierprototyp bezeichnet ein Abbild des Screens einer Anwendung auf Papier. Papierprototypen werden verwendet um die Abfolge, die Zusammenhänge, die grundsätzliche Struktur von Screens sowie grobe Navigations- und Interaktionsideen abzubilden. Die Prototypen werden einfach gestaltet, daher ist dies ein schneller und kostengünstiger Weg, um Ansätze zur Nutzeroberfläche darzustellen. Papierprototyping kann iterativ solange angewendet werden, bis die Idee ausgereift und ausdetailliert ist. Ausgehend von dem Zweck der Erstellung werden nur die zentralen Screens, Funktionalitäten und Interaktionsmöglichkeiten als Papiermodell nachgebaut. Bei der Entwicklung des Prototyps sollte möglichst auf hochwertiges Papier und Zeichenmaterial verzichtet werden, um unpassende Ansätze leichter und schneller wieder verwerfen zu können. Diese Einfachheit der Papierprototypen hat den Vorteil, dass Testpersonen leichter kritisieren als bei einem detailliert ausgearbeiteten Design. Aufgrund dessen wird in der Regel ein sehr offenes und damit auch wertvolles Feedback hervorgebracht.

3 Verwandte Arbeiten

Der Einsatz einer Interaktionsanalyse in Meetings ist keine kürzliche Erscheinung. Weit verbreitet sind diese vor allem in der Psychologie, wo diese oft zur Verbesserung von Produktionsprozessen eingesetzt werden. Aber auch in der Softwareentwicklung wird mittlerweile vermehrt von ihnen Gebrauch gemacht, da Meetings im Softwareentwicklungsprozess immer mehr an Relevanz zugenommen haben.

3.1 Interaktionsanalysen von Meetings aus verschiedenen Bereichen

In dieser Arbeit wird sich mit der Kommunikationsanalyse in Softwareprojekt-Meetings beschäftigt. Jedoch werden im Folgenden zuerst verwandte Arbeiten vorgestellt, die nicht auf den Bereich der Softwareentwicklung eingegrenzt sind. Auch die Erkenntnisse aus anderen Wissenschaften in Bezug auf Interaktionsanalysen sind für Softwareprojekt-Meetings von Bedeutung, da nicht ausschließlich die Meetings als solche wichtig sind, sondern vor allem die Interaktionen während dieser.

Tobia und Becker [28] beschäftigten sich bereits 1990 mit der Relevanz von effektiven Meetings, da diese teilweise sehr viele Stunden einer Arbeitswoche beanspruchen. Sie berichteten, dass drei Fähigkeiten von Meetingteilnehmern besonders wichtig sind für den Meetingerfolg. Die erste Fähigkeit ist der Umgang mit störendem Verhalten. Eine Konfrontation mit einer störenden Person könnte einen offenen Konflikt auslösen und sich somit auf die Effektivität des Meetings auswirken. Die beiden anderen Fähigkeiten, die die Meetingeffektivität positiv beinflussen können, sind zum einen aktives Zuhören und zum anderen Kommunikationsfähigkeiten.

Loung und Rogelberg [18] untersuchten die Arbeitswoche von 37 Probanden, um den Einfluss der Häufigkeiten von Meetings zu analysieren. Dabei pflegten die Probanden eine Woche täglich ihre Meetings in ein Arbeitstagebuch ein und schrieben Berichte über ihr Wohlergehen. Diese Daten wurden anhand von hierarchischer linearer Modellierungsanalyse untersucht. Es lies sich ein positiver Zusammenhang zwischen der Anzahl von Meetings und der Arbeitnehmermüdigkeit sowie subjektiver

Arbeitsbelastung feststellen.

Kauffeld und Lehmann-Willenbrock [11] zeigten den Einfluss des Kommunikationsverhaltens in Meetings auf den Team- und organisatorischen Erfolg. Der organisatorische Erfolg wurde dabei beispielsweise als Umsatz des Unternehmens oder der Anteil neuer Produkte vom gesamten Umsatz beschrieben. Dafür wurden 92 Teammeetings mit dem Codierschema *act4teams* analysiert und ausgehend davon Zusammenhänge zur Meetingzufriedenheit, Teamproduktivität und zum organisatorischen Erfolg erzeugt (vgl. Kapitel 2.1.2). Beispielsweise konnte eine höhere Meetingzufriedenheit bei Teams, die mehr funktionale Interaktion zeigten, festgestellt werden. Im Gegensatz dazu konnte festgestellt werden, dass sich dysfunktionale Kommunikation, wie zum Bespiel das Verlieren in Details und Beispielen, negativ auf die Meetingzufriedenheit, die Teamproduktivität und den organisatorschen Erfolg auswirkt.

Lehmann-Willenbrock et al. [17] analysierten den Zusammenhang zwischen verbalen Interaktionssequenzen und der Gruppenstimmung in Teammeetings. Dafür wurden 52 Arbeitsgruppengespräche aus zwei Industrieunternehmen mit *act4teams* codiert. Dabei wurden *Beschwerde*-Zyklen und *Interesse-an-Veränderungen*-Zyklen aufgedeckt. Es wurde festgestellt, dass sich *Beschwerde*-Zyklen negativ auf die Gruppenstimmung auswirken und *Interesse-an-Veränderungen*-Zyklen mit einer aktiven Gruppenstimmung verknüpft sind.

Lehmann-Willenbrock et al. [16] analysierten den potenziellen Vorteil von prozeduraler Kommunikation in Team-Meetings. Die prozedurale Kommunikation ist definiert als verbale Verhaltensweise, die eine Gruppendiskussion strukturiert und somit das Erreichen von Zielen fördert. Dafür wurden 59 Team-Meetings aus 19 Organisationen analysiert. Dabei stellte sich unter anderem heraus, dass je gleichmäßiger prozedurale Kommunikation unter den Meetingteilnehmern verteilt ist, desto zufriedener sind die Teilnehmer mit dem Diskussionsverauf und dessen Ergebnissen.

3.2 Interaktionsanalysen von Softwareprojekt-Meetings

Mittlerweile werden Interaktionsanalysen auch in der Softwareentwicklung zunehmend eingesetzt.

Dabei wurden bereits 1992 von Olsen et al. [22] Softwaredesign-Meetings analysiert, indem diese auf Video aufgenommen und transkribiert wurden. Insgesamt zehn Design-Meetings aus vier Projekten und zwei Organisationen wurden aufgenommen. Das verwendete Codierschema untersuchte vor allem Aktivitäten zur Problemlösung, Ko-

ordination und Verwaltung der Teilnehmer. Dabei zeigte sich, dass insgesamt nur 40 Prozent der gesamten Zeit für direkte Diskussionen über das Design aufgewendet wurde. 30 Prozent der Meetingszeit verbrachten die Teilnehmer mit Zusammenfassungen und Walkthroughs von Fortschritten und 20 Prozent mit reinen Koordinationsaktivitäten.

Klünder et al. [15] untersuchten das Meeting- und Kommunikationsverhalten in acht Softwareprojekt-Entwicklerteams in einer Zeit von 15 Wochen, um herauszufinden, wie sich das Kommunikationsverhalten über die Projektdauer ändert. Dabei stellte sich heraus, dass die Häufigkeiten von Meetings über die Zeit weniger und die Kommunikation zwischen den Teilnehmern dezentraler wurden.

Schneider et al. [25] analysierten das Kommunikationsverhalten von 32 initialen studentischen Softwareprojekt-Meetings anhand des *act4teams*-Codierschemas. Aus den Analyseergebnissen konnte entnommen werden, dass proaktive Aussagen einen merklichen positiven Einfluss auf den Gruppenaffekt haben. Dieser Effekt trat jedoch nur auf, wenn diese von unterstützenden Aussagen begleitet wurden.

Im Rahmen ihrer Bachelorarbeit [4] implementierte Haak die erste Version der von *act4teams* abgeleiteten, vereinfachten Analyse *act4teams-SHORT* in eine Desktop Anwendung, welche im Anschluss im Rahmen anderer Projekte weiterentwickelt wurde [13], [23], [24]. Acht Probanden codierten insgesamt je eine Hälfte eines Meetings anhand der *act4teams-SHORT*-Analyse der ersten Iteration. Im Anschluss wurde anhand einer deskriptiven Analyse die Codierungen der *act4teams-SHORT*-Analysen mit der *act4teams*-Referenzcodierung desselben Meetings verglichen.

Prenner [23] und Prenner et. al [24] erweiterten die von Haak [4] implementierte erste Version des *act4teams-SHORT*-Tools um eine Feedbackmethode. Diese misst den Erfolg eines Meetings anhand des Feedbacks der Meetingteilnehmer. Prenner [23] und Prenner et al. [24] verglichen die Zufriedenheit der Entwickler nach einem Meeting, erhoben mit der Feedbackmethode des *act4teams-SHORT*-Tools, mit der objektiven Interaktionsanalyse des gleichen Meetings. Daraus folgte, dass anhand der Feedbackmethode ebenfalls Rückschlüsse darüber gezogen werden können, ob ein Meeting eher gut oder schlecht verlief. Zudem lassen sich Aussagen über die Zufriedenheit der Teilnehmer mit dem Meeting ableiten, welches ein wichtiger Faktor für den Projekterfolg ist. Des Weiteren fiel Prenner et al. [24] auf, dass das Codierschema von *act4teams-SHORT* zu grob-granular gefasst ist, um angemessene Vergleiche ziehen zu können.

Klünder [13] führte im Rahmen ihrer Dissertation unter anderem ein Experiment mit zwölf unerfahrenen Meetingbeobachtern durch. Die Probanden codierten alle ein Meeting mit der *act4teams-SHORT*-Analyse. Die erhobenen Meetingprofile wurden im

Anschluss mit zweier von erfahrenen Analysten erhobenen verglichen und es stellte sich heraus, dass diese zu einem annehmbaren Grad übereinstimmten. Des Weiteren sollten die Teilnehmer im Anschluss Handlungsempfehlungen aussprechen. Dabei zeigte sich, dass auch unerfahrene Beobachter die Ergebnisse ihrer Datenerhebungen interpretieren und Empfehlungen ableiten können. Allerdings hängen diese sehr stark vom Beobachter und dessen Erfahrungen ab. Hinzu wurden die Teilnehmer gefragt, ob ihnen die Differenzierung zwischen den Kategorien während der Analyse schwer fiel. Dabei stellte sich heraus, dass die Unterscheidung der Kategorien den unerfahrenen Meetingbeobachtern nicht leicht fiel. Um diesen entgegenzuwirken, wurden im Rahmen dieser Arbeit Tutorials erstellt, die die unerfahrenen Meetingbeobachter auf die erste *act4teams-SHORT*-Analyse vorbereiten sollen.

Klünder et al. [14] untersuchten mit einer verfeinerten Version des Codierschemas der *act4teams-SHORT*-Analyse im Rahmen einer industriellen Fallstudie bei Volkswagen Commercial Tires zehn teaminterne Meetings in frühen Projektphasen. Dabei wurde zum einen untersucht, wie Teammitglieder eines Softwareprojektes in frühen Projektphasen in Meetings interagieren. Zum anderen wurde untersucht, ob die Teilnehmer ausschließlich über Probleme sprechen oder diese auch versuchen zu lösen. Dabei stellte sich heraus, dass in frühen Projektphasen die meiste Zeit Informationen ausgetauscht wurden, gefolgt von der Identifikation von Problemen, Diskussion über Lösungen und dem Lösen von Problemen. Des Weiteren stellte sich heraus, dass das Team direkt von Beginn mit dem Lösen von Problemen begann. Je mehr Probleme benannt wurden während des Meetings, desto mehr sprach das Team über Lösungen.

4 Konzept und Implementierung der Lernverfahren

Der Fokus von Nutzern von edukativer Software soll auf dem Lernen liegen, nicht auf der Bedienung des Computers [1]. Um dies zu erreichen, muss die Software von klaren edukativen Prinzipien und Überlegungen designt werden [1]. Einige wichtige, grundsätzliche Schritte zum Design von edukativer Software sind folgende [1]:

- Definition des edukativen Ziels
- Identifikation der Lernbedürfnisse
- Bestimmung der Lernmethode
- Design und Implementierung der Software
- Evaluation der Software

In Bezug auf die definierten edukativen Ziele werden in diesem Kapitel unter anderem die Schritte *Identifikation der Lernbedürfnisse*, *Bestimmung der Lernmethode* und *Design und Implementierung der Software* beschrieben. Zuerst wird im Abschnitt *Ideenfindung* darauf eingegangen, woher die grundlegenden Ideen zur Konzeptualisierung stammen (Kapitel 4.1). Dazu werden die zukünftigen Nutzergruppen anhand von Personas dargestellt und beschrieben. Danach werden Schwierigkeiten aufgeführt, welche Nutzer bei der anfänglichen Analyse mit dem *act4teams-SHORT*-Tool hatten. Folgend werden aus diesen Schwierigkeiten Maßnahmen und Lernmethoden abgeleitet, die die Grundlage für die beiden Tutorialkonzepte legen. Die Konzepte werden daraufhin aufgeführt. Im Anschluss werden anhand von *Papierprototypen* die Konzepte prototypisch dargestellt. Im Abschnitt *Meetingaufnahme* wird das in den Tutorials integrierte Aufnahmematerial beschrieben (Kapitel 4.2). Abschließend werden beide Lernverfahren vorgestellt und deren Implementierung aufgezeigt (Kapitel 4.3).

4.1 Ideenfindung

Um auf die Nutzer zugeschnittene Tutorials zu konzeptualisieren, wurden Erfahrungen von *act4teams-SHORT*-Nutzern gesammelt und diese in den Entwicklungsprozess mit einbezogen.
Dafür wurde ein Brainstorm-Meeting organisiert, an dem ausschließlich Personen teilnahmen, die bereits Erfahrungen mit der *act4teams-SHORT*-Analyse gemacht hatten. Der erste Beitrag des Meetings war es, die zukünftigen Nutzergruppen zu adressieren. Dabei wurden Eigenschaften, Einschränkungen und Fähigkeiten dieser herausgearbeitet und im Nachhinein mithilfe einer Persona als Repräsentant einer Nutzergruppe dargestellt. Danach wurden Erfahrungen, Schwierigkeiten und Probleme der Meetingteilnehmer mit der anfänglichen *act4teams-SHORT*-Analyse ausgetauscht. Anhand dessen und mit Blick auf die zukünftigen Nutzer wurden Maßnahmen abgeleitet, die diese anfänglichen Schwierigkeiten mit der *act4teams-SHORT*-Analyse reduzieren sollen.

4.1.1 Zukünftige Nutzer

Folgende potenzielle Usergruppen wurden während des Brainstorm Meetings adressiert: Projektmanager, Teamleiter und Softwareentwickler.

Der Projektmanager ist verantwortlich für den Projekterfolg. Daher ist es für ihn von großem Interesse, dass die einzelnen Teams gut zusammenarbeiten können und produktive Meetings gestalten. Um dies zu untersuchen, setzt er sich stichprobenartig in verschiedene Meetings einzelner Teams und analysiert deren Interaktionen. Auch der Teamleiter bemüht sich um produktive Meetings seines Teams. Denn auch er möchte zum Projekterfolg beitragen und ist daher ein zukünftiger Nutzer des Tools. Bei Abwesenheit des Teamleiters kann an dessen Stelle auch der Softwareentwickler die Analyse des Meetings durchführen. Auch dieser ist interessiert an einer guten Zusammenarbeit im Team und ist motiviert eine Meetinganalyse zu erheben.

Im Folgenden werden für jede der drei Nutzergruppen eine Persona vorgestellt und daraus Schwierigkeiten und Fähigkeiten beim Umgang mit dem *act4teams-SHORT*-Tool abgeleitet.

Persona 1:

Thomas Müller, Projektmanager

- Alter: 56
- Familienstand: verheiratet, 1 Kind
- Personalität: extrovertiert, analytisch, erfolgsorientiert
- Wohnhaft: Hannover
- Ausbildung: Master BWL
- Verantwortlichkeiten: Projekterfolg
- Fähigkeiten: Organisation, Führung, gute Menschenkenntnisse
- Einschränkungen: keinen Bezug zur Softwareentwicklung, keine Programmiererfahrung, keinen technischen Hintergrund

Thomas Müller begann seine Karriere vor 35 Jahren bei einem großen hannoverschen Unternehmen. Angefangen als strategischer Einkäufer ging er über zum Vertriebsleiter. Die vielen Jahre der Personalverantwortung schulten seine Führungsqualitäten und er bewies sich bei Organisationsaufgaben. Durch den täglichen Umgang mit seinen Mitarbeitern eignete er sich sehr gute Menschenkenntnisse an. Dies qualifizierte ihn zum Aufstieg in das Projektmanagement, wo er letztendlich vor zehn Jahren die Position des Projektmanagers einnahm.
Durch unzählige Meetings, die er während seiner beruflichen Laufbahn führte, hat er ein gutes Gespür dafür entwickelt, wie Aussagen und Interaktionen der Kollegen einzuordnen sind. Er versteht eine Aussage zu bewerten und daher ist auf psychologischer Basis eine gute Intuition zu erwarten, diese zu einer der *act4teams-SHORT*-Kategorien zuzuordnen. Schwierigkeiten könnten ihm dabei jedoch die informationstechnologischen Begriffe in Softwareprojekt-Meetings bereiten, da er keinen Bezug zur Softwareentwicklung oder zum Programmieren hat. Somit können inhaltliche Verständnisprobleme auftreten, die ihm das Einordnen von Aussagen in eine der Kategorien erschweren.

Persona 2:

Susanne Weigelt, Teamleiterin

- Alter: 41
- Familienstand: verheiratet, 2 Kinder
- Personalität: extrovertiert, analytisch, kreativ
- Wohnhaft: München
- Ausbildung: Bachelor Wirtschaftsinformatik, Master Informatik
- Verantwortlichkeiten: Teamerfolg, Teilprojekterfolg
- Fähigkeiten: Organisation, Durchsetzungsvermögen, IT-Kenntnisse
- Einschränkungen: keine praktische Programmieranwendung mehr

Susanne Weigelt bemerkte schon im Laufe ihres Grundstudiums ihr Interesse an IT-Security, sodass sie von Wirtschaftsinformatik im Bachelor zu Informatik im Master wechselte. Nach ihrem Abschluss ergatterte sie sich ihren Wunschjob als Softwareentwicklerin für Sicherheitssysteme. Ihr Durchsetzungsvermögen und ihre Kommunikationsfähigkeit verhalfen ihr zu der Position der Teamleiterin, die sie nun seit sieben Jahren ausübt. Seitdem hat sich einiges verändert bezüglich ihrer täglichen Aufgabenbereiche, denn die praktische Programmieranwendung gehört nicht mehr dazu. Stattdessen ist sie verantwortlich für die Koordination von Aufgaben und Mitarbeitern und hat die Funktion als Ansprechpartner für Projektfortschritte und für Mitarbeiterbelange. Bei der Analyse mithilfe des *act4teams-SHORT*-Tools hat sie im Vergleich zum Projektmanager ein grundlegendes Verständis über die Themen in den Meetings und auch fachspezifische Ausdrücke versteht sie. Schwierigkeiten könnte die Interpretation der Kategorien bereiten, da mit der bisherigen Berufserfahrung die psychologischen Begrifflichkeiten womöglich noch unerklärt sind. Auch die Interpretation der Aussagen und die Zuordnung dieser zu einer der Kategorien könnten Schwierigkeiten bereiten.

Persona 3:

Simon Kimski, Softwareentwickler

- Alter: 26
- Familienstand: ledig
- Personalität: kreativ, analytisch
- Wohnhaft: Hameln
- Ausbildung: Master Informatik
- Verantwortlichkeiten: persönlicher Arbeitsauftrag
- Fähigkeiten: IT-Praxis, Programmierkenntnisse, Wissen von aktuellen IT-Themen
- Einschränkungen: wenig Berufserfahrung, wenig Erfahrung mit Teamarbeit und wenig Meetingerfahrung

Simon Kimski ist seit seinem Abschluss in einem internationalen Unternehmen in einem großen Projekt als Junior-Systemanalytiker tätig. Das Unternehmen setzt auf agile Entwicklungsmethoden und orientiert sich am *SCRUM*-Prozess. Da dieser Entwicklungprozess regelmäßige Meetings wie Sprint Plannings, Reviews und Retrospektiven, sowie Daily Scrums beinhaltet, konnte er in den mittlerweile drei Jahren im Berufsleben erste Meetingerfahrung sammeln. Bei der Verwendung des *act4teams-SHORT*-Tools ist aufgrund der wenigen Berufserfahrung zu erwarten, dass ihm die Kategoriedefintionen zum größten Teil unklar sind. Auch die Interpretation der Aussagen mittels der Kategorien wird demzufolge eine Herausforderung darstellen. Einen Vorteil gegenüber dem Teamleiter und dem Projektmanager hat er, da er Teil der Softwareentwicklung ist und die Themen im Meeting ihm dadurch bestens bekannt sind. Generell ist sein Wissen an aktuellen IT-Themen sehr hoch und die Meetinginhalte sind ihm vertraut.

Bei der Betrachtung und Gegenüberstellung der Personas stellt sich heraus, dass besonders zwei Fähigkeiten für die Benutzung des Analysetools von Vorteil sind. Zum einen sind es die technischen Skills, die den IT-Bezug, Programmiererfahrung und Wissen von aktuellen IT-Themen beinhalten. Diese sind wichtig, um dem Softwareprojekt-Meeting inhaltlich folgen zu können und die Zusammenhänge in den Aussagen zu verstehen. Zum anderen sind es Soft Skills, insbesondere die sozialen und methodischen Kom-

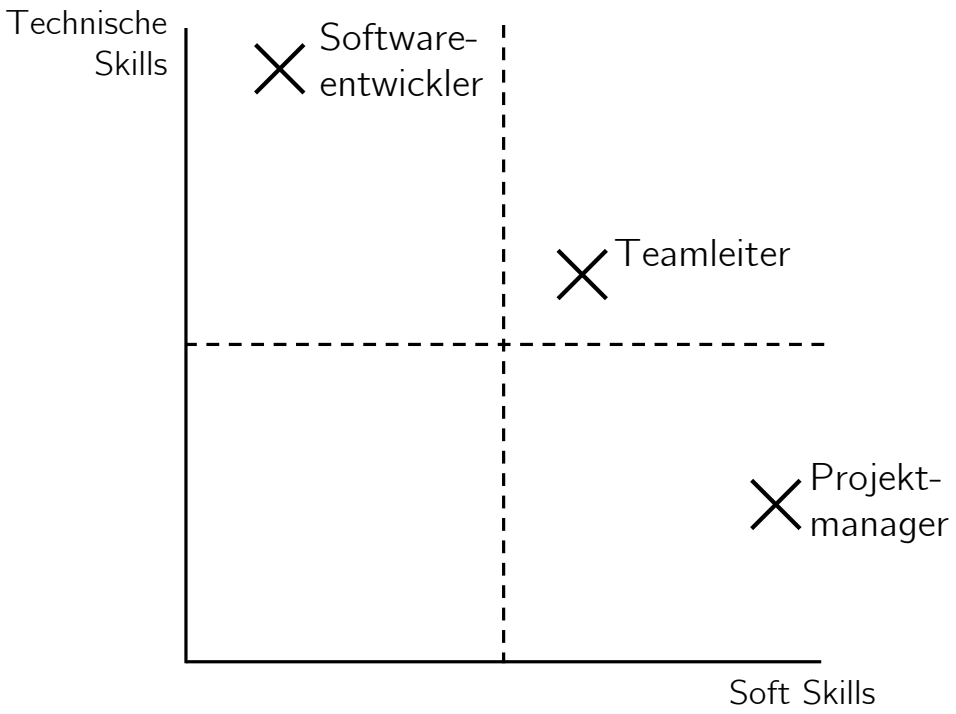

Abbildung 4.1: Technische vs. Soft Skills

petenzen. Diese sind wichtig, um auf der Beziehungsebene eine Aussage deuten zu können. Eine graphische Übersicht über die Einordnung der jeweiligen Nutzergruppe zu den technischen und Soft Skills zeigt die Abbildung 4.1. Der Softwareentwickler hat die höchste technische Kompetenz, da tägliche IT Aufgaben seinen Alltag bestimmen. Im Gegensatz dazu hat er aber die geringsten Soft Skills, da Personalverantwortung bisher nicht in seinem Aufgabenbereich lag. Der Projektmanager hat wenig technische Skills, da er zuvor nicht in der Softwareentwicklung tätig war. Jedoch hat er dafür, durch die langjährige Personalführung, sehr hohe Soft Skills erreicht. Der Teamleiter hat leicht überdurchschnittliche technische und Soft Skills, da dieser sowohl einen IT-Bezug als auch bereits Erfahrung in der Personalführung gesammelt hat.

4.1.2 Schwierigkeiten

Im Folgenden werden die Probleme und Schwierigkeiten aufgeführt, die im Brainstorm-Meeting in Bezug auf die anfängliche Nutzung der *act4teams-SHORT*-Analyse gesammelt wurden.

Problem 1 - **Codierung.** Es wurde die Schwierigkeit aufgezeigt, den Anfang und das

Ende eines Tags zu bestimmen. Dies ist darauf zurückzuführen, dass zu codierende Sinneinheiten teilweise mehrfach in einer Aussage auftreten können. Dieses Problem wurde anhand der Kategorie *Informationsweitergabe* erläutert. Die Nutzer konnten schwer identifizieren, ob es sich um eine einzelne oder mehrere Informationsweitergaben handelte und wie oft sie dabei hätten einen Tag setzen müssen. Ein weiteres Beispiel wurde anhand der Kategorie *Probleme* aufgezeigt. Dabei wussten die Nutzer nicht, wie oft sie taggen sollen, wenn länger über dasselbige Problem gesprochen wurde, dieses aber unterschiedlich aufgeführt wurde.

Problem 2 - **Deutung.** Probleme kamen auch bei der Deutung von Aussagen auf. Oftmals empfanden die Nutzer es schwierig, Aussagen zu interpretieren und diese der richtigen Kategorie zuzuordnen. Die inhaltlichen Aussagen in Kombination mit dem Tonfall der sprechenden Person führte bei den Nutzern zu verschiedenen Interpretationsmöglichkeiten. Dabei wurde die Situation beschrieben, in der eine Information in einem angreifenden Tonfall weitergeben wurde. Dies führte zu der Mehrdeutigkeit in der Interpretation, ob diese Aussage zur Kategorie *Informationsweitergabe* oder *Destruktives Verhalten* eingeordnet werden soll.

Problem 3 - **Abgrenzung.** Hinzu gibt es Kategorien, bei denen die Abgrenzung untereinander teilweise schwer fiel. Dies wurde in Bezug auf die Kategorien *Probleme*, *Lösungen* und *Vernetzungen und Verknüpfungen* aufgezeigt.

Problem 4 - **Definitionen.** Die Nutzer wurden in schriftlicher Form anhand von längeren Definitionspassagen über die Kategorien und deren Bedeutungen aufgeklärt. Dabei fiel es den Nutzern schwer, sich diese zu merken und sie griffen deswegen während der Analyse teilweise auf selbst gedachte Definitionen zurück.

Problem 5 - **Schneller Aussagenwechsel.** Es wurde als schwierig empfunden, eine Diskussion zu codieren. Der schnelle Wechsel zwischen Personen und Aussagen stellte die Nutzer vor eine große Herausforderung, da kaum Zeit zum Nachdenken und zum Einsortieren der Tags zu den Kategorien blieb.

4.1.3 Maßnahmen

Anhand dieser Schwierigkeiten wurden im Nachhinein sowie bereits teilweise während des Brainstorm-Meetings Maßnahmen zur Reduzierung dieser entworfen. Eine Übersicht, welcher Lösungsvorschlag welches Problem beheben soll, bietet Tabelle 4.1. Folgend werden die Lösungsvorschläge vorgestellt:

Schwierigkeiten	*Lösungsvorschläge*
Codierung Deutung	Vorcodierte Meetinganalyse Quiz Probeanalyse
Abgrenzung	Zusammenhängende Beispiele
Definitionen	Buzzword-Definitionen
Schneller Aussagenwechsel	Kombination aus allen Lösungsvorschlägen

Tabelle 4.1: Schwierigkeiten bei der anfänglichen *act4teams-SHORT*-Nutzung mit entsprechenden Lösungsvorschlägen

Lösungsvorschläge für die Probleme 1 und 2 - Codierung und Deutung:
Vorschlag 1 - **Vorcodierte Meetinganalyse**: Zur Reduzierung dieser Schwierigkeiten wird ein Meetingausschnitt abgespielt, in dem die tatsächliche Codierung visuell eingearbeitet ist. So bekommt der Nutzer einen Überblick, wann und wie oft Tags zu jeweiligen Aussagen gesetzt werden.
Vorschlag 2 - **Quiz**: Der Nutzer führt ein Quiz durch, anhand dessen er beispielhafte Aussagen aus einem Softwareprojekt-Meeting kategorisiert. Danach bekommt der User die Lösungen dazu und er kann nachprüfen, in welchen Fällen er richtig lag oder nicht. Anhand dieser Beispiele bekommt der Nutzer ein Gefühl von dem Umfang einer zu kategorisierenden Aussage.
Vorschlag 3 - **Probeanalyse**: Der Nutzer führt eine Probeanalyse durch. Dabei codiert der Nutzer einen Meetingausschnitt und im Anschluss kann er seine eigene Codierung mit der Referenzcodierung vergleichen. Die Gegenüberstellung kann mit einem Zeistrahl visualisiert werden. Der Nutzer bekommt so einen Überblick, an welchen Stellen er etwas anderes hätte codieren müssen und bekommt somit noch einige echte Beispiele von zugeordneten Aussagen zu einer Kategorie aufgezeigt.

Lösungsvorschlag für das Problem 3 - Abgrenzung.
Vorschlag - **Zusammenhängende Beispiele**: Kategorien, die untereinander schwer abzugrenzen sind, können anhand von zusammenhängenden Beispielen erklärt werden. Umgesetzt werden kann dieses anhand von kleinen Dialogen, die die jeweiligen Kategorien beinhalten. Zum Beispiel die Kombination aus *Problemen*, *Lösungen* und *Verknüpfungen und Vernetzungen*.

Lösungsvorschlag für das Problem 4 - Definitionen.
Vorschlag - **Buzzword-Definitionen**: Die Verwendung von kurzen, prägnanten Buzzword-Definitionen zur Erklärung der Kategorien, anstelle von langen Definitionspassagen. Dies soll dazu führen, dass sich die Nutzer die Definitionen besser merken können und

somit auf keine Eigendefinitionen zurückgegeriffen werden muss.

Lösungsvorschlag für das Problem 5 - Schneller Aussagenwechsel.
Vorschlag - **Tutorial**: Die Schwierigkeit einen schnellen Aussagenwechsel zu codieren, soll mit der Durchführung eines Tutorials reduziert werden. Anhand des Tutorials soll eine Entscheidungsbasis aufgebaut werden, die während der Analyse schneller abgerufen werden kann. So soll idealerweise nicht so viel Zeit zum Nachdenken gebraucht werden.

4.1.4 Konzepte

Die Lösungsvorschläge **Buzzword-Definitionen** und **Zusammenhängende Beispiele** eignen sich gut, um den Nutzern eine kurze Einführung in die Kategorien zu geben, bevor das eigentliche Tutorial durchgeführt wird. Die beiden Vorschläge beinhalten zum einen die Lernstrategie *Organisationsstrategie*, da die Informationen so organisiert werden, dass sie in verständlicher Weise aufeinander Bezug nehmen und diese zudem auf das Wesentliche reduziert sind. Zum anderen zeigt sich eine *Wiederholungsstrategie*, da der Nutzer sich durch wiederholtes Durchlesen die Kategorien einprägen soll. Die Lösungsvorschläge **Vorcodierte Meetinganalyse** und **Quiz** eignen sich bestens für jeweils eins der Tutorialkonzepte, da es sich dabei um *Elaborationsstrategien* handeln. Diese sollen ein tieferes Verständnis des Lernstoffs bewirken, indem neue Informationen mit dem Vorwissen verknüpft werden. Dabei verfolgen beide Vorschläge unterschiedliche Vorgehensweisen bei der Aufbereitung der Lerninhalte. Die Vorgehensweise des Vorschlags **Vorcodierte Meetinganalyse** präsentiert die Lerninhalte auf bimodale Weise, da die Codierung zum einen visuell und der Meetingausschnitt zum anderen auditiv verfolgt werden kann. Es ist bewiesen, dass bimodal aufbereitete Informationen besser im Gedächtnis bleiben, als ausschließlich visuelle oder auditive [20]. Die Vorgehensweise eines **Quiz** unterscheidet sich grundlegend von der der **Vorcodierten Meetinganalyse**. Statt sich mit einem bereits codierten Meeting zu beschäftigen, sollen Nutzer Aussagen selbst kategorisieren. Der Nutzer wird somit zum aktiven Überlegen und eigenständigen Nachdenken gebracht, was durch Studien belegt zu einem Lerneffekt führt [26]. Der Lösungsvorschlag **Probeanalyse** eignet sich eher weniger für ein Tutorialkonzept, da abgesehen von einer Kategoriebeschreibung keine weitere Vorbereitung stattfindet, bevor die Analyse durchgeführt werden soll. Jedoch kann die Probeanalyse eingesetzt werden, um das Gelernte aus den Tutorials unter Beweis zu stellen. Anhand dessen kann abgeschätzt werden, wie effektiv die beiden Konzepte sind. Abschließend kann der Nutzer sich Feedback anhand der Gegenüberstellung seiner Codierung und der Referenzcodierung einholen. Daraus kann abgelesen

werden, zu welchen Zeitpunkten der Nutzer Tags gesetzt hat und wann welche in der Referenzcodierung auftauchen.

Die zuvor aufgeführten Maßnahmen werden in folgender Reihenfolge zu den zwei Konzepten zusammengestellt:

1. Kombination aus **Buzzword Definitionen** und **zusammenhängender Beispiele** zur Übersicht und Beschreibung der Kategorien.
2a. Tutorial: **Vorcodierte Meetinganalyse** oder
2b. Tutorial: **Quiz**
zum Aufbau einer Entscheidungsbasis und zum Generieren eines tieferen Verständnisses.
3. **Probeanalyse**, um das aus dem Tutorial Gelernte unter Beweis zu stellen. Feedback im Anschluss anhand einer Gegenüberstellung der Codierung des Nutzers und der Referenzcodierung.

4.1.5 Papierprototypen

Die **Probeanalyse**, das Tutorial **Vorcodierte Meetinganalyse** und das **Quiz**-Tutorial wurden mit Bedacht auf die Designrichtlinien zur Usability in Papierprototypen umgesetzt. Die **Buzzword-Definitionen** und die **zusammenhängenden Beispiele** wurden nicht in einen Papierprototyp umgewandelt, weil die Beschreibung derer für eine Vorstellung ausreicht.
Bei der Umsetzung in die Papierprototypen fiel auf, dass für die Integration eines Meetingausschnitts eine Audiodatei besser geeignet ist als eine Videoaufnahme. Eine Videoaufnahme würde einen Großteil des Screens einnehmen, sodass zu wenig Platz für die anderen Elemente übrig bliebe. Eine genaue Ausführung, was zu der Wahl einer Audiodatei geführt hat, folgt im nächsten Abschnitt *Meetingausschnitte*.
Die Abbildungen 4.2 und 4.3 zeigen die Papierprototypen zu dem Entwurf der Oberfläche der **Probeanalyse**. Dabei ist in Abbildung 4.2 das Interface abgebildet, mit welchem die Daten der Probeanalyse erhoben werden soll. Das Interface soll so ähnlich wie möglich dem des *echten* Analyse-Interfaces sein, um eine Konsistenz zu schaffen und dem Nutzer so auf die Analyse bestmöglich vorzubereiten, indem dieser Aufbau und Funktionen bereits kennenlernt (vgl. Abbildung 2.2: Oberfläche der Datenerhebung). Der Probeanalysen-Screen wurde um einen Slider oberhalb der Codierungsfläche ergänzt. Dieser gibt dem Nutzer fortlaufend Feedback, wie weit die Meetinganalyse bereits fortgeschritten ist. Im Gegensatz zu der richtigen Meetinganalyse-Oberfläche wird die Dauer nicht unterhalb der Codierungsfläche angezeigt, sondern oberhalb am Ende des Sliders. Dies ist eine Anordnung, die häufig so verwendet wird und deshalb

dem Nutzer bekannt vorkommt.

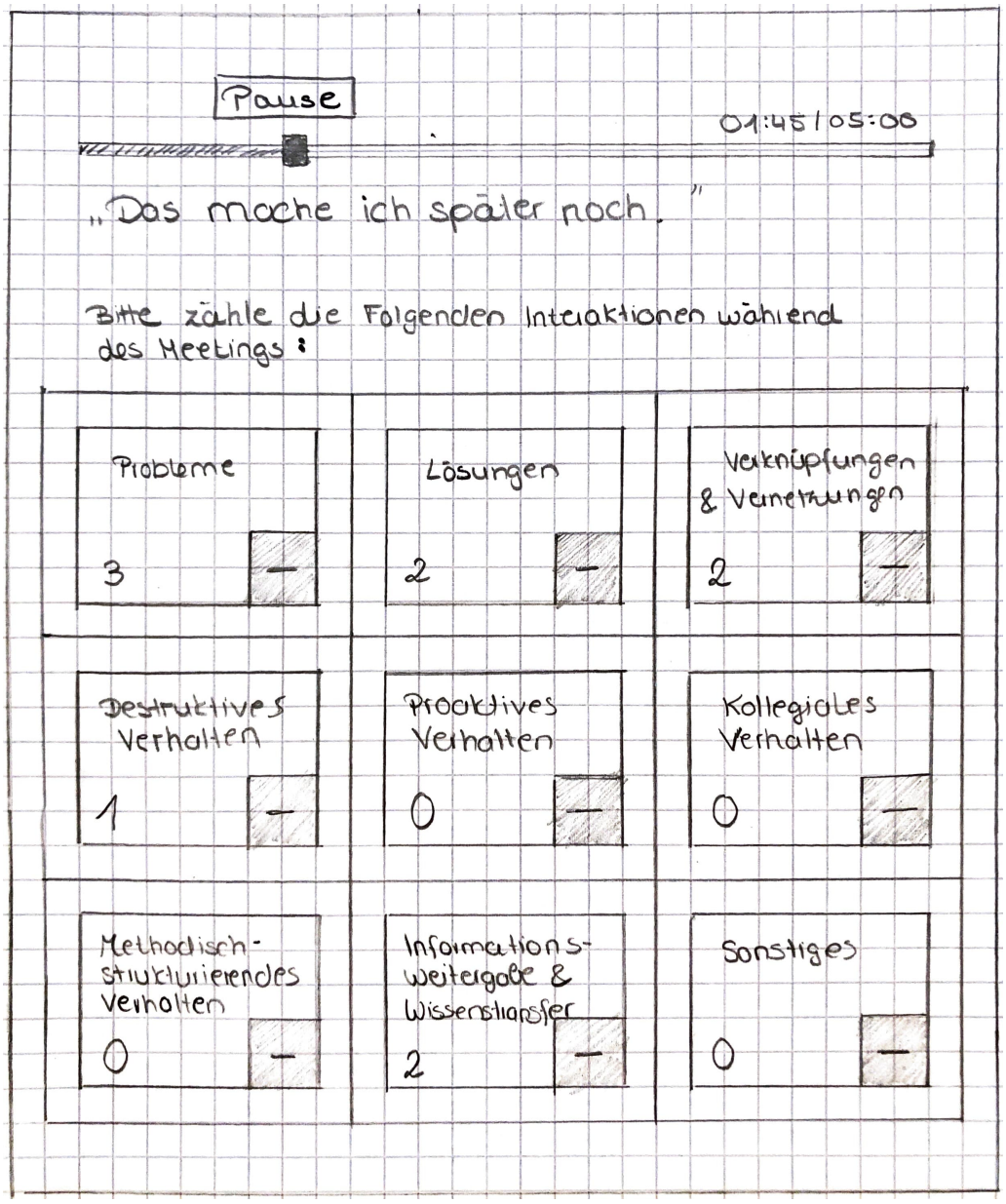

Abbildung 4.2: Papierprototyps des Probeanalyse-Interfaces

Zudem macht es den Zusammenhang zwischen Slider und Dauer deutlich. Auf dem Papierprototypen ist der aktuell gesprochene Text des Meetingausschnitts angezeigt, damit der Nutzer diesen nebenbei mitlesen kann. Bei der Implementierung wurde dieser jedoch herausgenommen, damit der Nutzer sich vollständig auf die Codierung und die Codierschaltfäche fokussieren kann und dabei nicht abgelenkt wird. Denn es gilt bei dem Design eines Interfaces, sich ausschließlich auf das Wesentliche zu beschränken.

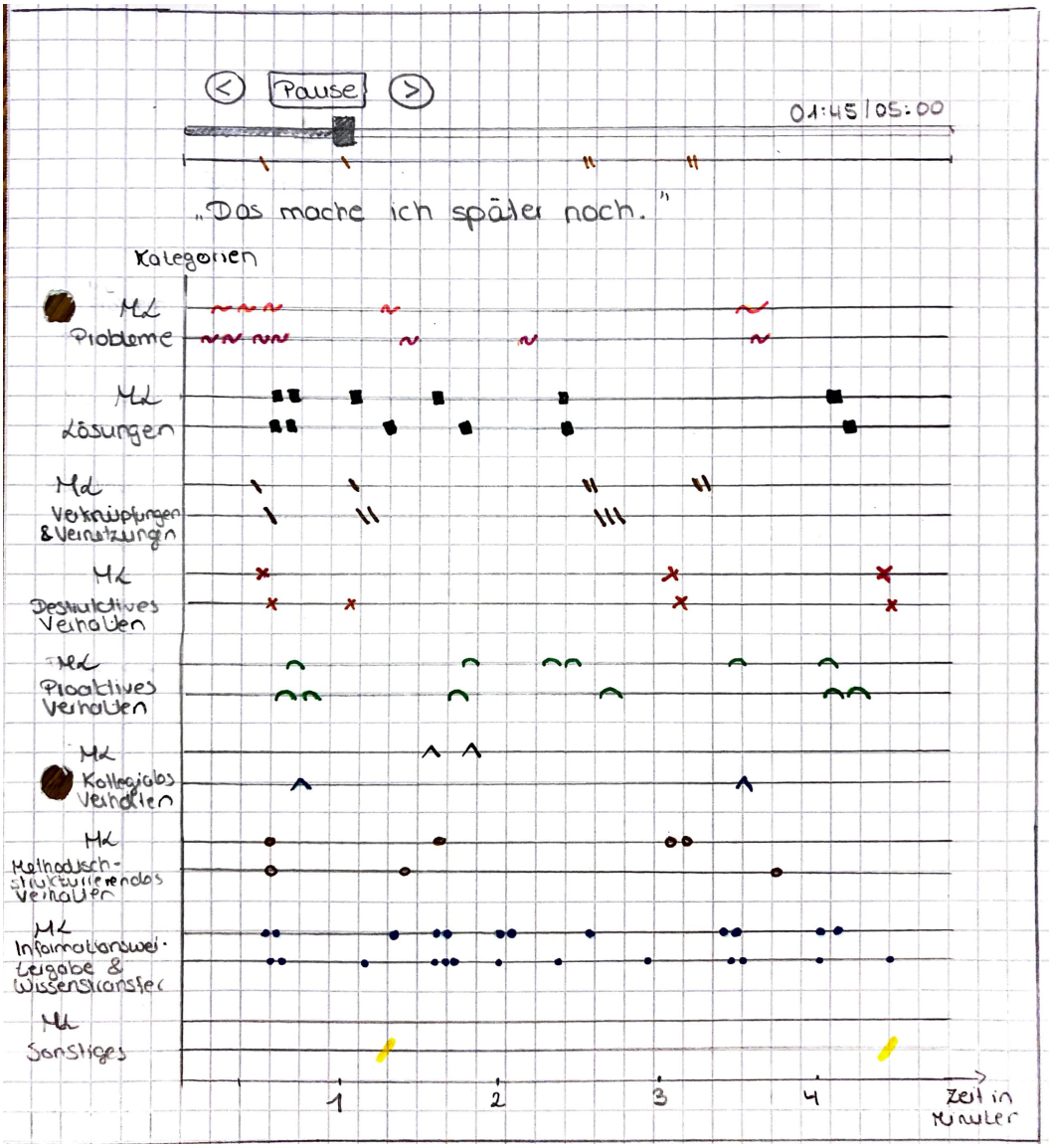

Abbildung 4.3: Papierprototyps des Vergleichs zwischen der Codierung des Nutzers und der Referenzcodierung anhand eines Zeitstrahls

Abbildung 4.3 zeigt den Papierprototypen des Interfaces, welches nach der Probeanalyse erscheint. Darauf sind alle vom Nutzer gesetzten Tags und die der Referenzcodierung des Meetingausschnitts der Probeanalyse über die Zeit abgebildet. Zum Beispiel sind auf der zweiten Linie von oben die vom Nutzer gesetzten Tags zur Kategorie *Probleme* abgebildet. Auf der Linie darüber sind die Tags der Kategorie *Probleme* der Musterlösung gezeigt. Diese Anordnung wurde so gewählt, um Zusammengehörigkeit zu demonstrieren. Entsprechend wurde so zu allen Kategorien verfahren. Die Zeitstrahldar-

stellung wurde gewählt, da nach einer *realen* Meetinganalyse eine ähnliche Darstellung generiert wird (vgl. Abbildung 2.4: Zeitstrahl einer Beispielanalyse). Oberhalb des Zeitstrahls ist wieder ein Slider mit der gleichen Audiodatei von der Probeanalyse. Auch die Angabe der Dauer befindet sich an der gleichen Position. Diese Anordnung wurde so gewählt, um den Konsistenzaspekt zu berücksichtigen. Der Slider hat das gleiche Längenverhältnis wie die Zeitachse, damit die Tags auf dem Zeitstrahl der gleichen Position wie auf dem Slider entsprechen.

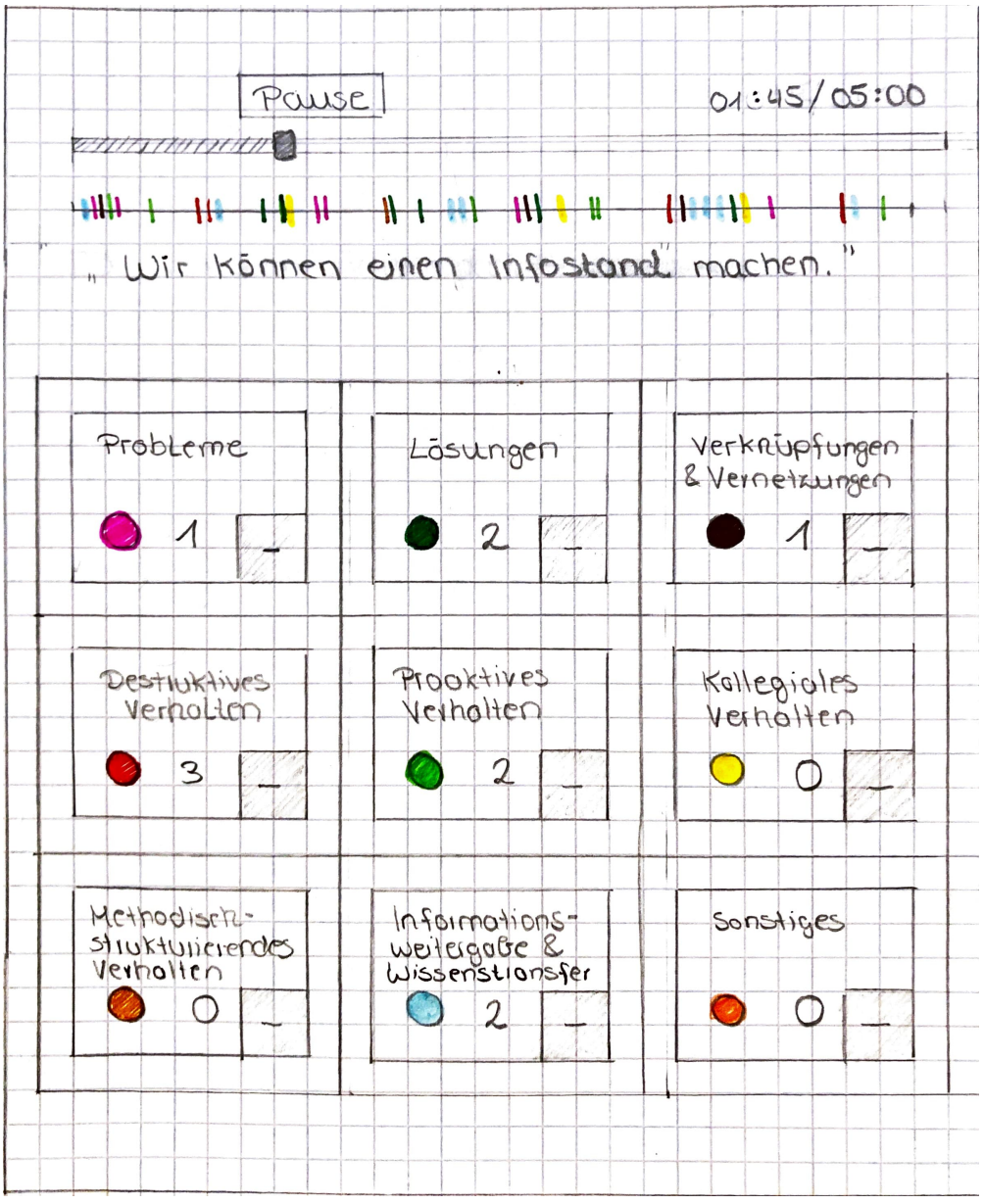

Abbildung 4.4: Papierprototyp eines Beispielmeetingausschnitts mit Referenzcodierung

Abbildung 4.4 zeigt den Entwurf zum Oberflächendesign der **Vorcodierten Meetinganalyse** anhand eines Papierprototypen. Dabei wurde wieder die gleiche Kategorieanordnung gewählt wie schon bei der Probeanalyse-Oberfläche und der für die *reale* Analyse. Oberhalb des Interfaces ist der Slider, der den auditiven Meetingausschnitt beinhaltet. Auch dieser findet befindet sich in der gleichen Position wie in den zuvor beschriebenen Oberflächen. Direkt darunter findet sich ein Zeitstrahl, auf dem die Referenzcodierung abgebildet ist. Dabei zeigt die Position des Tags auf dem Zeitstrahl, an welcher Stelle in der Audiodatei dieser gesetzt wurde. Damit der Nutzer die Codierung nicht einfach auswendig lernt und während der Probeanalyse angibt, wird hierbei ein anderer auditiver Meetinausschnitt verwendet. Die farblichen Punkte auf den Kategoriebuttons sind dafür da, die jeweilige Kategorie zu selektieren oder zu deselektieren. Eine Deselektion führt dazu, dass alle Kategorievorkommen vom Zeitstrahl verschwinden. Dies soll es den Nutzern ermöglichen, den Fokus jeweils nur auf bestimmte Kategorien zu richten. Eine entsprechende Selektion fügt die Vorkommen der entsprechenden Kategorie wieder dem Zeitstrahl hinzu. Die Tags auf dem Slider sind in der Farbe des Punktes der jeweiligen Kategorie eingefärbt, um die Zusammengehörigkeit darzustellen. Die Anordnung der Punkte auf die Kategoriebuttons soll die Relation der beiden Elemente symbolisieren. Während der Meetingausschnitt abgespielt wird, soll die Referenzcodierung der Codierungsschaltfläche entnommen werden können. Dies soll realisiert werden, indem immer, wenn ein Tag in der Referenzcodierung vorkommt, der entsprechende Kategoriebutton im selben Moment aufleuchtet.
Abbildung 4.5 zeigt den Entwurf der Oberfläche des **Quiz**-Tutorials als Papierprototyp dargestellt. Bei diesem Interface wurde wie auch schon in den anderen Oberflächen zuvor die gleiche Anordnung der Kategorienschaltfläche gewählt. Über den Kategorien steht jeweils die aktuelle Aussage, die vom Nutzer kategorisiert werden soll. Entscheidet der Nutzer sich für eine Antwort, klickt er diese an und bekommt im Anschluss direktes Feedback anhand der Lösung aufgezeigt. Danach kann er zur nächsten Aussage weitergehen. So wird verfahren, bis alle Aussagen von ihm kategorisiert wurden. Rechts oberhalb der abgebildeten Aussage ist eine Anzeige, die dem Nutzer permanentes Feedback über das Verhältnis der kategorisierten Aussagen zu der Gesamtanzahl der zu kategorisierenden Aussagen gibt. So weiß der Nutzer jederzeit an welcher Stelle er sich im Tutorial befindet.

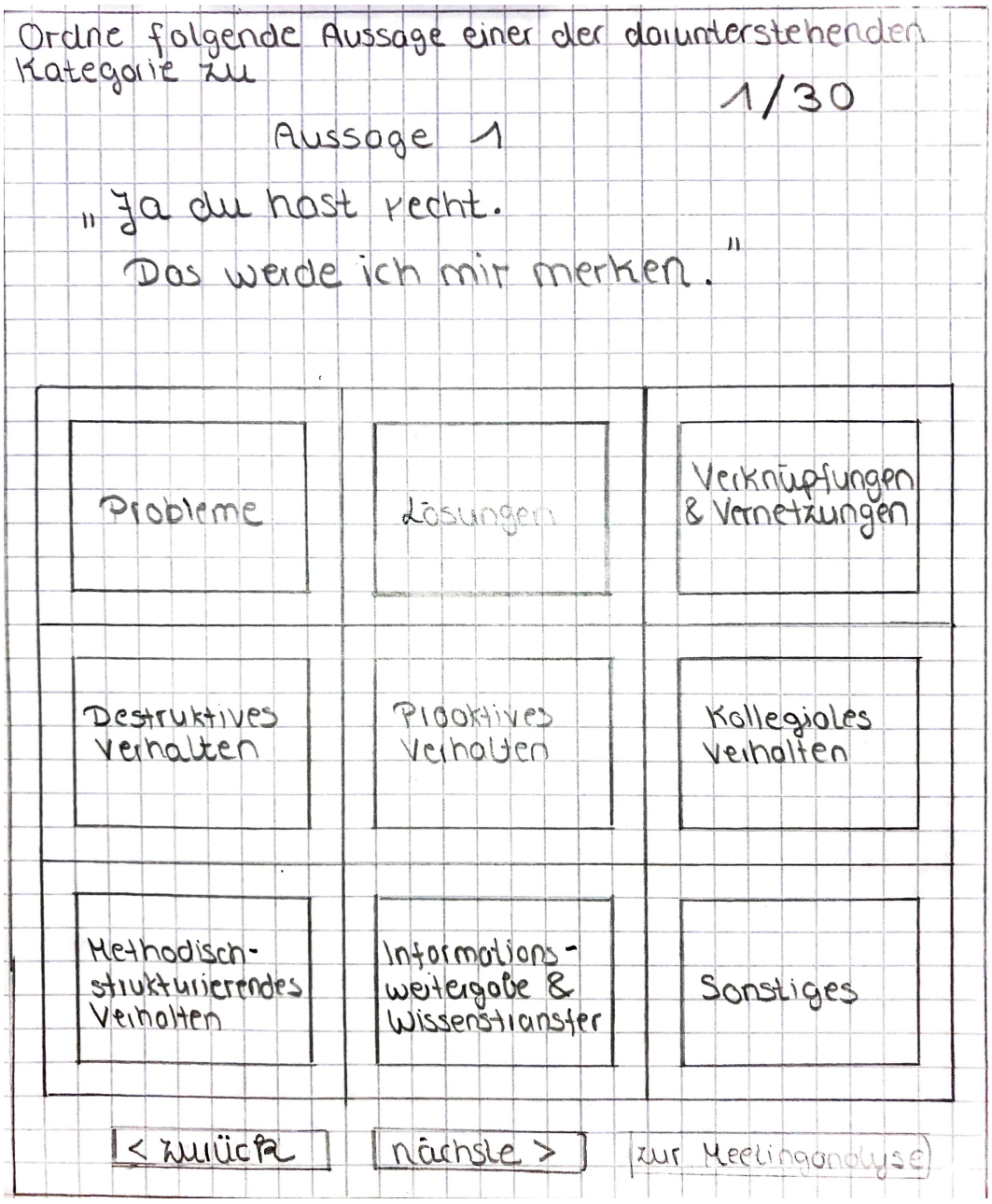

Abbildung 4.5: Papierprototyps Quiz

4.2 Meetingaufnahme

Für die Integration der Meetingausschnitte stand ein im Rahmen der Lehrveranstaltung *Softwareprojekt* auf Video aufgenommenes Softwareprojekt-Meeting des Bachelorstudiengangs Informatik der Leibniz Universität Hannover zur Verfügung. Es wurde sich

jedoch gegen eine Integration des Videomaterials und für eine Integration einer Audiospur entschieden. Gründe dafür waren zum einen die Notwendigkeit, die Teilnehmer in dem Video aus Datenschutzbestimmungen unkenntich machen zu müssen. Dadurch hat ein Video keinerlei Vorteile gegenüber der einfachen Audiospur mehr, da die Mimik zur Interpretation der Aussagen für den Meetingbeobachter nicht mehr sichtbar wäre. Noch hinzu kann aufgrund der Unkenntlichkeit die Zuordnung von Gesagtem zu den entsprechenden Personen nicht eindeutig nachvollzogen werden, wodurch ein weiterer Vorteil eines Videos abhanden käme. Zum anderen würde eine Videodarstellung auch zu viel Platz auf dem Screen einnehmen, sodass andere Elemente, wie die Kategorieübersicht nicht mehr in der Darstellung des Interfaces zur Datenerhebung der echten Meetinganalyse entsprechen würde.
Bei der Untersuchung der Audiospur des Videos des Softwareprojekt-Meetings nach zwei passenden Ausschnitten zur Integration in die Lernsoftwares kamen weitere Probleme auf. Ein großes Problem stellte dabei die nicht ausreichende Tonqualität dar. Abhängig von der Distanz zwischen Meetingteilnehmer und Aufnahmemedium können einige Teilnehmer nur sehr erschwert verstanden werden. Noch dazu führt die Datei ein grundlegendes Rauschen mit sich, welches manche Aussagen völlig unkenntlich macht. Des Weiteren überlagern sich die Teilnehmer beim Sprechen, was ebenfalls dazu führt, dass einige Aussagen untergehen und keine klare Differenzierung mehr möglich ist.
Aus diesen Gründen wurden Passagen aus dem Meeting transkribiert und darauf folgend simuliert eingesprochen. Dies hat zudem den Vorteil, dass insbesondere bei dem Lernvideo darauf geachtet werden konnte, dass alle Kategorien in der Codierung enthalten sind und der Nutzer somit zu jeder Kategorie Beispiele vorgestellt bekommt, die er zum Lernen dieser nutzen kann.

4.3 Implementierung der Lernverfahren

Die Lernverfahren stellen sich wie folgt zusammen:

1. Erklärungen zu den Kategorien anhand der Definitionen und eines Beispiels
2. Tutorial (Vorcodierte Meetinganalyse oder Quiz)
3. Probeanalyse
4. Zeitstrahlgegenüberstellung der Nutzercodierung un der Referenzcodierung der Probeanalyse

4.3.1 Erklärungen zu den Kategorien

Beide Lernverfahren beginnen mit einer Erklärung zu den Kategorien. Abbildung 4.6 zeigt die implementierte Übersicht. Jeweils am Rand stehen die Kategorien, gefolgt von prägnanten, kurzen Beschreibungen zu diesen. Jede Sprechblase in der Mitte enthält eine Beispielaussage zu der verbundenen Kategorie. Diese wurden in einen Kontext gesetzt, um schwer voneinander abzugrenzende Kategorien in einem zusammenhängenden Beispiel zu erklären.

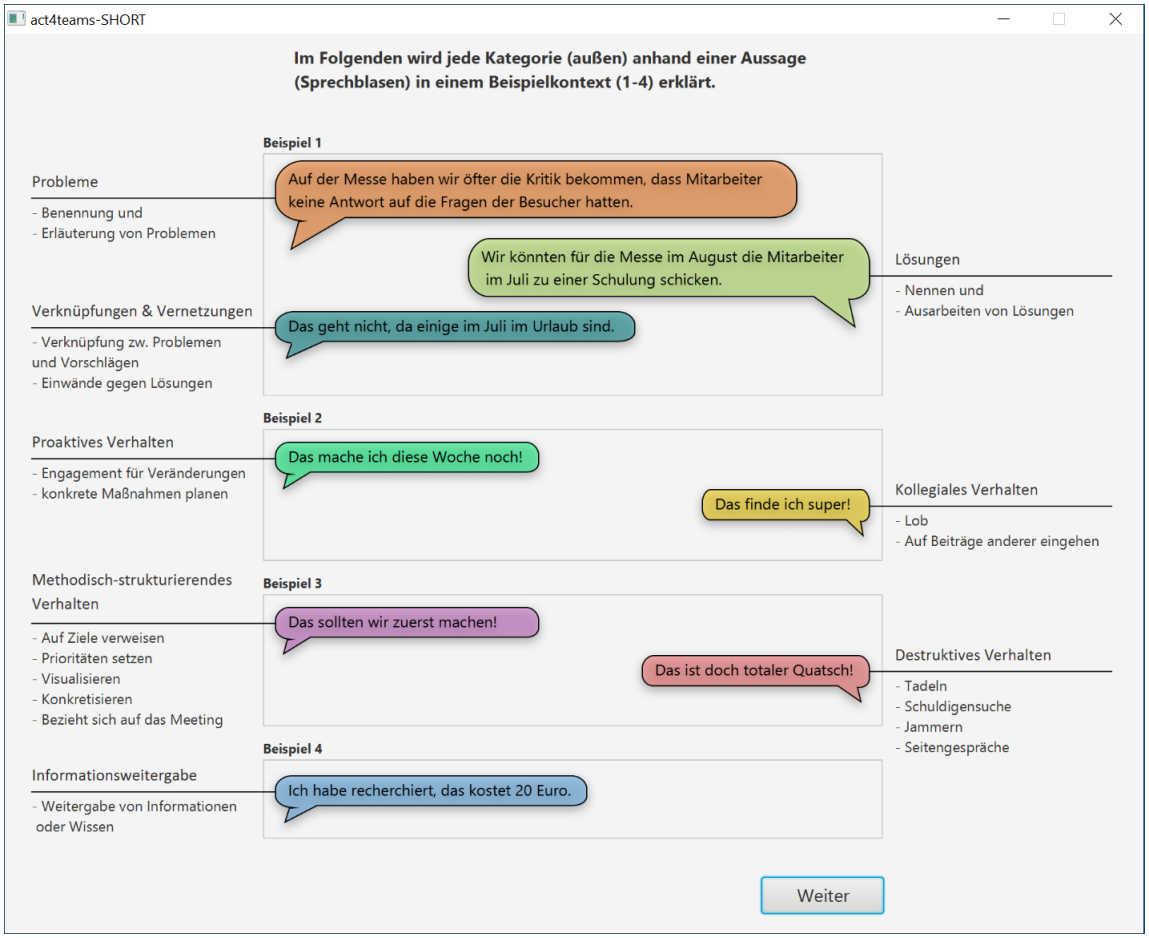

Abbildung 4.6: Erklärungen zu den Kategorien

4.3.2 Tutorials

Nachdem die Übersicht verinnerlicht wurde, folgt im Anschluss das Tutorial. Dazu wird entweder das *Quiz*-Tutorial oder das Tutorial *Vorcodierte Meetinganalyse* durchgeführt.

Vorcodierte Meetinganalyse

(a) (b)

Abbildung 4.7: Interface des Tutorials *Vorcodierte Meetinganalyse*

Dieses Tutorial beinhaltet einen Meetingausschnitt mit bereits integrierter Referenzcodierung. Das Interface des Tutorials wird in Abbildung 4.7 (a) dargestellt. Zuerst hört sich der Nutzer den Meetingausschnitt bis zum Schluss komplett an. Dabei kann er die Referenzcodierung anhand des Zeitstrahls unterhalb des Sliders ablesen. Dieser beinhaltet jeden Tag, aufgeführt an der Stelle auf dem Strahl, an dem dieser zeitlich in der Audiodatei vorkommt. Zu Beginn eines Tags leuchtet die entsprechende Kategorie in voller Farbe auf. Danach verblasst das Aufleuchten stückweise bis das Ende des Tags erreicht ist. Abbildung 4.7 (a) weist aktuell auf einen Tag der Kategorie *Informationsweitergabe Wissenstransfer* hin. Weiterhin kann der Nutzer, während der auditive Meetingausschnitt abgespielt wird, das Transkript dessen mitlesen.
Nachdem der Meetingauschnitt durchgelaufen ist, werden dem Nutzer anhand von aufkommenden Tooltips die Interaktionsmöglichkeiten mit dem Interface erklärt. Dies ist in der Abbildung 4.7 (b) dargestellt. Der User hat die Möglichkeit sich anhand der Pfeil-Buttons zum jeweiligen voherigen oder nächsten Tag zu navigieren. Des Weiteren kann er mit einem gezielten Klick auf einen Tag des Zeitstrahls direkt zu dessen Stelle in der Audiodatei gelangen, um sich diesen anzuhören. Die Tags des Zeitstrahls las-

sen sich zudem nach beliebigen Kategorien filtern. Um die Tags einer Kategorie nicht mehr auf dem Zeitstrahl anzeigen zu lassen, deselektiert man diese jeweils, indem die Checkbox auf dem jeweiligen Button der Kategorie angeklickt wird. Umgekehrt werden mit einem weiteren Klick die Tags der Kategorie wieder angezeigt. Neben der rechten Pfeiltaste wurden zusätzlich zwei Shortcuts hinzugefügt. Der obere selektiert mit einem Klick alle Kategorien und der untere deselektiert alle. Dies hat den Vorteil, dass wenn der User die Tags einer einzelnen Kategorie betrachten möchte, dieser nur einmal auf *Keine Kategorie* klickt und danach die gewünschte Checkbox anspricht. Somit wurden aus acht Klicks ohne Shortcut zwei Klicks mit Shortcut.

Quiz

Abbildung 4.8: Interface des *Quiz-Tutorials*

Dieses Tutorial beinhaltet die Durchführung eines Quiz. Abbildung 4.8 zeigt dessen implementierte Oberfläche. Um die Vergleichbarkeit der beiden Tutorials zu gewährleisten, wurden für die Kategorisierung Aussagen aus dem Meetingausschnitt des Tutorials *Vorcodierte Meetinganalyse* genommen. Oberhalb der Kategorienfläche kann die aktuell zu kategorisierende Aussage entnommen werden. Rechts oberhalb der

Aussage kann die Anzahl der bereits zugeordneten und die gesamte Anzahl der Aussagen abgelesen werden. Um zur nächsten Aussage zu gelangen, muss immer erst die aktuelle einsortiert werden. Wurde die letzte Aussage eingeordnet, besteht die Möglichkeit sich alle ein weiteres Mal anzuschauen.

4.3.3 Probeanalyse

Nachdem der User eines der beiden Tutorials durchgeführt hat, folgt im Anschluss die Probeanalyse. Das Interface zur Datenerhebung der Probeanalyse mit beispielhafter Codierung kann der Abbildung 4.9 entnommen werden.
Die Probeanalyse erfolgt anhand eines knapp über dreiminütigen auditiven Meetingausschnitts. Dabei versucht der Nutzer so gut wie es geht diesen zu codieren. Vernimmt der User eine Kategorie, so kann er einen Tag setzen, indem er die entsprechende Kategorie anklickt. Hat er aus Versehen einen falschen Tag gesetzt, kann er diesen anhand des kleinen grauen Minus-Buttons wieder zurücksetzen.

Abbildung 4.9: Interface zur Datenerhebung der Probeanalyse

4.3.4 Zeitstrahlgegenüberstellung

Abbildung 4.10 zeigt beispielhaft die Oberfläche zur Gegenüberstellung der Codierung der Probeanalyse des Nutzers und der der Referenzcodierung.

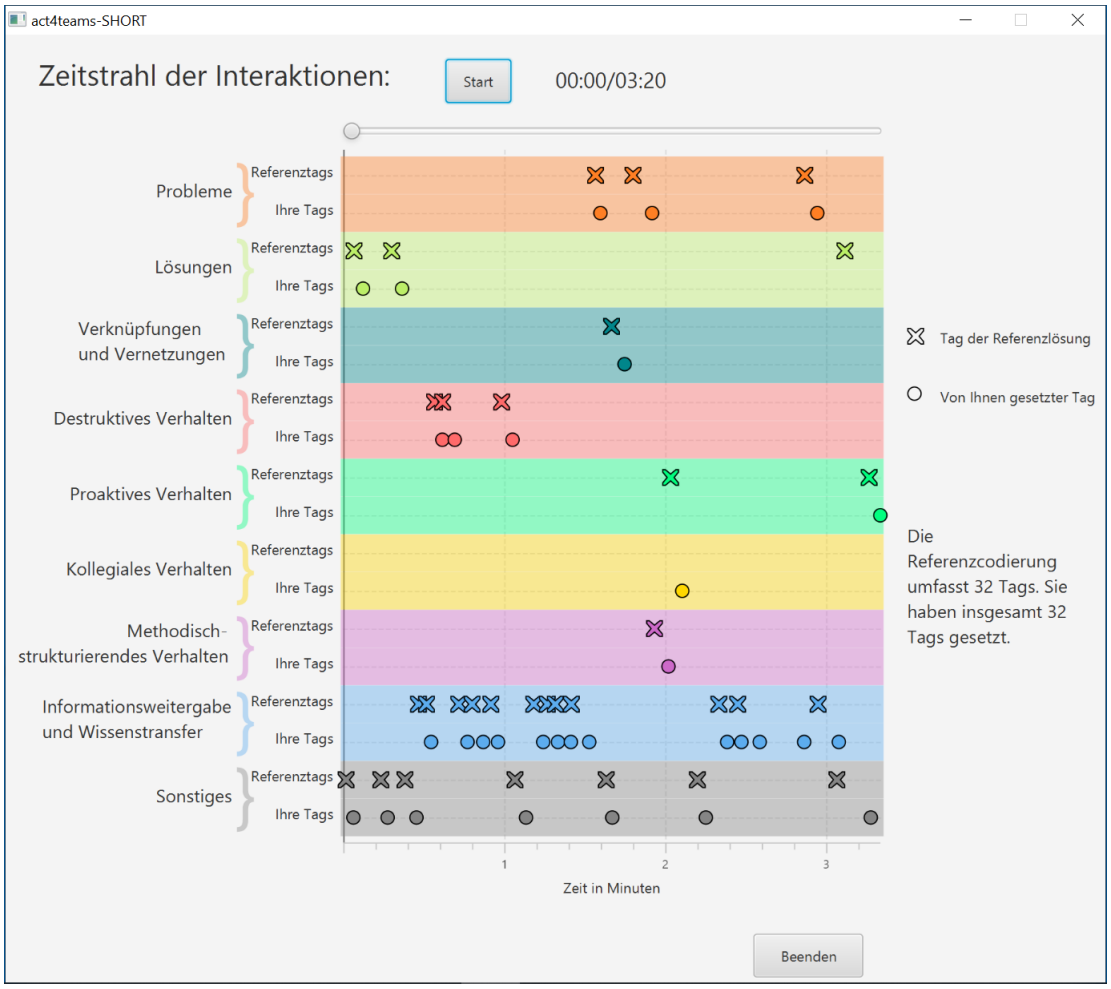

Abbildung 4.10: Gegenüberstellung vom Nutzer gesetzter Tags zur Referenzcodierung über die Zeit

Die Kreuze symbolisieren die Tags der Referenzcodierung. Diese sind auf den Linien zur entsprechenden Kategorie abgebildet. Jeweils unter diesen sind die Tags des Nutzers der gleichen Kategorie anhand von Kreisen abgebildet. Zudem ist die Audioatei des zur Probe analysierten Meetingausschnitts in der Oberfläche integriert. Dabei bewegt ein Klick auf einen Tag die Navigation zur entsprechenden Stelle in der Audiodatei. Bewegt der Nutzer den Cursor über einen Tag der Referenzcodierung, wird die zugehörige Aussage aufgezeigt. An der rechten Seite des Interfaces kann der Nutzer noch entnehmen,

wie viele Tags er gesetzt hat und wie viele Tags die Referenzcodierung enthält.
Nach der Untersuchung der Codierungen gelangt der Nutzer anhand des *Weiter*-Buttons zum Startmenü, wo er im Anschluss eine echte Meetinganalyse starten kann.

5 Evaluation

Um die Tutorials auf ihre Güte zu testen, wurden im Rahmen dieser Arbeit beide Konzepte in einem Experiment evaluiert.

5.1 Zielsetzung und Hypothesen

Dieses Kapitel beschäftigt sich mit der Experimentvorbereitung. Dafür wird zu Beginn das Ziel des Experimentes dargestellt und im Weiteren werden Hypothesen, die es zu prüfen gilt, herausgearbeitet und aufgeführt.

5.1.1 Zielsetzung

In der Vorbereitungsphase des Experimentes wurde als erster Schritt das Forschungsziel definiert. Zur Sicherstellung einer klaren Definition wurde das Ziel mittels des *Goal-Definition-Templates* von Wohlin et al. [29] beschrieben.

Forschungsziel

Analysiere die Konzepte der *act4teams-SHORT*-Tutorials
mit dem Ziel einen Lerneffekt sicherzustellen
in Bezug auf die Erzeugung besserer Meetinganalysen
aus der Sicht unerfahrener *act4teams-SHORT*-Nutzer
im Rahmen eines Experimentes mit Informatikstudenten.

Um Schwierigkeiten bei der anfänglichen Nutzung der *act4teams-SHORT*-Meetinganalyse zu reduzieren, soll ein Tutorial eingesetzt werden. Dieses soll idealerweise helfen, Begrifflichkeiten zu verstehen und eine Entscheidungsgrundlage für das Kategorisieren von Aussagen aufzubauen.
Das Ziel der Tutorials *Quiz* und *Vorcodierte Meetinganalyse* ist somit, bei unerfahrenen Nutzern einen messbaren Lerneffekt zu generieren, sodass diese bessere Meetinganalysen erzeugen als unerfahrene Beobachter, die zuvor kein Tutorial durchgeführt

haben.
Dabei sollen die Tutorials Aspekte der Usability berücksichtigen, um Akzeptanz und Zufriedenheit bei den Nutzern hervorzurufen [19].

5.1.2 Hypothesen

Das Hauptziel eines Tutorials ist, Nutzer auf die entsprechende Anwendung vorzubereiten [9]. Durch das Sammeln von Erfahrungen wird demzufolge ein Lerneffekt hervorgerufen. Bei der Anwendung des Tutorials *Vorcodierte Meetinganalyse* oder des *Quiz*-Tutorials sammelt der unerfahrene Meetingbeobachter Erfahrungen in der Meetinganalyse und baut folglich eine Entscheidungsbasis bei der Kategorisierung von Aussagen auf. Dementsprechend wird angenommen, dass die Durchführung von einem der beiden Tutorials vor der ersten Meetinganalyse mit *act4teams-SHORT* von einem unerfahrenen Meetinganalysten einen Einfluss auf dessen Ergebnis hat.

$H1_1$: Eine vorausgehende Durchführung des Tutorials *Vorcodierte Meetinganalyse* hat einen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.

$H2_1$: Eine vorausgehende Durchführung des *Quiz*-Tutorials hat einen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.

Die Tutorials beinhalten verschiedene Vorgehensweisen, um den Nutzer auf die Meetinganalyse mit *act4teams-SHORT* vorzubereiten. Dabei gilt es herauszufinden, welche der Vorgehensweisen effektiver ist.
Beide Tutorials halten Lernstrategien in unterschiedlicher Form und verschiedene Informationsaufbereitungen für den Nutzer bereit. Das Tutorial *Vorcodierte Meetinganalyse* setzt auf bimodale Präsentation der Lerninhalte. Die Codierung kann visuell verfolgt werden, während parallel zum einen eine visuelle Darstellung des zugehörigen Transkriptes auf dem Bildschirm abläuft und zum anderen das Skript als Tonaufnahme abspielt. Informationen, die bimodal aufbereitet sind, bleiben besser im Gedächtnis als ausschließlich visuell oder auditiv dargestellte [20].
Das *Quiz*-Tutorial hingegen kommuniziert die Inhalte rein visuell. Auch die Vorgehensweise unterscheidet sich vom Tutorial *Vorcodierte Meetinganalyse*. Statt sich mit einem bereits codierten Meeting zu beschäftigen, soll der unerfahrene Beobachter Aussagen selbst kategorisieren. Somit soll der Nutzer zum aktiven Überlegen und eigenständigen Nachdenken gebracht werden, was bereits durch Studien belegt zu einem Lerneffekt führt [26].

Daher wird angenommen, dass die unterschiedlichen Lernstrategien der Tutorials verschieden effektiv den Lernprozess der unerfahrenen Meetinganalysten beeinflussen. Die verschieden hohen Lerneffekte werden sich wiederum in unterschiedlich hohen Verbesserungen von Analyseergebnissen widerspiegeln.

$H3_1$: Welches der Tutorials durchgeführt wird, hat einen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.

Abgesehen von der Erzeugung eines objektiv messbaren Lerneffekts sollen die Tutorials auch die vom Nutzer wahrgenommene Komplexität in Bezug auf die Differenzierung der Kategorien reduzieren. Aufgrund der zuvor genannten Punkte in Bezug auf den Lerneffekt, wird angenommen, dass dieser somit zu einer Veränderung der subjektiven Komplexität in Bezug auf die Differenzierung der Kategorien führt.

$H4_1$: Die Durchführung eines Tutorials hat einen Einfluss auf die wahrgenommene Komplexität in Bezug auf die Unterscheidung der Kategorien von einem unerfahrenen Meetingbeobachter.

Wichtige Aspekte aus der Sicht der Nutzer sind die Konzepte *Ease of use* und *Joy of use* [19]. *Joy of use* drückt die Freude an der Nutzung aus. Empirische Studien belegen, dass positive Emotionen, wie Freude, die Lernmotivation sowie den Lerneffekt steigern können [7]. *Ease of use* steht für die einfache und verständliche Nutzung. Diese ist wichtig, um fokussiert den Inhalt der Tutorials aufnehmen zu können, ohne dabei von der Handhabung abgelenkt zu werden.
Beides sind wichtige Aspekte, die sich auf die Akzeptanz und die Zufriedenheit der Nutzer mit einer Anwendung auswirken [19].
Dazu wird zum einen untersuchen, welches der Tutorials aus der subjektiven Sicht der User einfacher anzuwenden ist, und zum anderen, welches den Usern besser gefällt. Aufgrund der unterschiedlichen Konzepte, wird angenommen, dass die Wahl des Tutorials einen Einfluss auf die wahrgenommene Einfachheit der Nutzung und das Gefallen an der Anwendung hat.

$H5_1$: Welches der Tutorials durchgeführt wird, hat einen Einfluss auf die vom unerfahrenen Meetingbeobachter wahrgenommene Einfachheit der Nutzbarkeit.

$H6_1$: Welches der Tutorials durchgeführt wird, hat einen Einfluss auf das Gefallen vom unerfahrenen Meetingbeobachter.

Die zuvor aufgeführten Hypothesen werden zwecks statistischer Analyse durch die folgenden Nullhypothesen kontrastiert:

$H1_0$: Eine vorausgehende Durchführung des Tutorials *Vorcodierte Meetinganalyse* hat keinen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeob-

achter und der Referenzcodierung.

$H2_0$: Eine vorausgehende Durchführung des *Quiz*-Tutorials hat keinen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.

$H3_0$: Welches der Tutorials durchgeführt wird, hat keinen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.

$H4_0$: Die Durchführung eines Tutorials hat keinen Einfluss auf die wahrgenommene Komplexität in Bezug auf die Unterscheidung der Kategorien von einem unerfahrenen Meetingbeobachter.

$H5_0$: Welches der Tutorials durchgeführt wird, hat keinen Einfluss auf die vom unerfahrenen Meetingbeobachter wahrgenommene Einfachheit der Nutzbarkeit.

$H6_0$: Welches der Tutorials durchgeführt wird, hat keinen Einfluss auf das Gefallen vom unerfahrenen Meetingbeobachter.

5.2 Planung des Experimentes

Dieser Abschnitt beschäftigt sich mit der Definition und Beschreibung der Studienparameter. Dafür werden zuerst die Experimentobjekte im Unterkapitel Meetingausschnitte vorgestellt. Darauffolgend wird im Abschnitt Experimentsubjekte die Zielgruppe adressiert und abschließend das Experimentdesign vorgestellt.
Das folgende methodische Vorgehen ist angelehnt an den Experimentprozess von Wohlin et al. [29].

5.2.1 Meetingausschnitte

Zwei simulierte Gespräche im Kontext eines Softwaremeetings wurden für die *Vorcodierte Meetinganalyse* sowie für die eigenständige Analyse eingesprochen und als Audiodatei in die Tutorials integriert. Die Kommunikationsinhalte wurden dabei von einem auf Video aufgenommenen Softwaremeeting entnommen. Dieses Meeting fand im Rahmen der Veranstaltung *Softwareprojekt* des Bachelorstudiums an der Leibniz Universität statt, an welchem Studierende der Informatik laut Regelstudienplan im fünften Semester teilnehmen. Mehrere Gründe befürworteten die Entscheidung zur simulierten Aufnahme anstelle der Verwendung des originalen Videomaterials (vgl. Kapitel 4.2). Dazu zählt beispielsweise eine Verbesserung der Tonqualität. Abhängig von der Distanz, die zwischen Meetingteilnehmer und Aufnahmemedium lag, können

einige Teilnehmer nur sehr erschwert verstanden werden. Hinzu führt das Original ein grundlegendes Rauschen mit sich, welches manche Aussagen vollkommen unkenntlich macht. Weiterhin sprechen teilweise mehrere Teilnehmer zur gleichen Zeit. Dies führt ebenfalls dazu, dass Aussagen unkenntlich sind und keine klare Differenzierung zwischen den sprechenden Personen möglich ist.

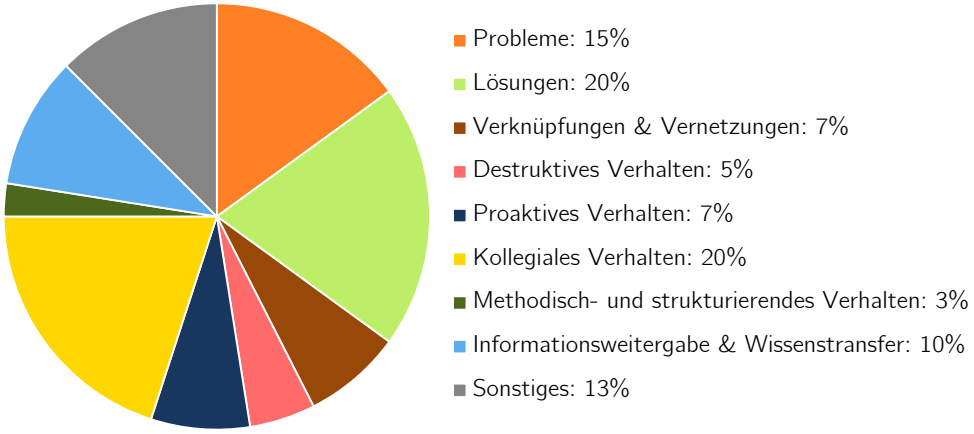

Abbildung 5.1: Referenzprofil des Meetingausschnitts der *Vorcodierten Meetinganalyse*

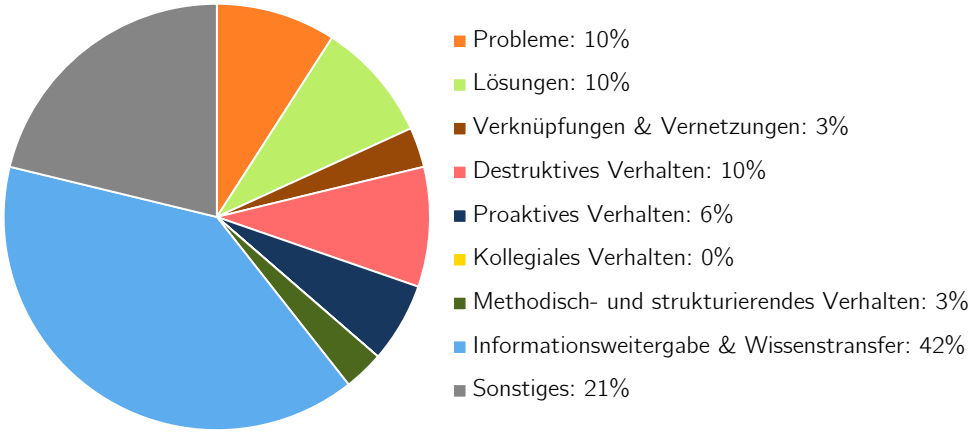

Abbildung 5.2: Referenzprofil des selbständig zu codierenden Meetingausschnitts

Der Meetingausschnitt für die *Vorcodierte Meetinganalyse* hat eine Länge von 03:16 Minuten. Bei der Aufnahme wurde besonders darauf geachtet, dass alle Kategorien in der Codierung vorkommen, sodass dem Probanden zum Lernen genügend Beispiele pro Kategorie zur Verfügung stehen. Dafür wurden Aussagensequenzen von unterschiedlichen Stellen des Meetingskriptes zusammengestellt. Das Profil des Meetingausschnitts kann aus der Abbildung 5.1 entnommen werden. Um eine Vergleichbarkeit

zu gewährleisten, wurden die gleichen Inhalte nur abgewandelt in schriftlicher Form bei dem *Quiz*-Tutorial als zu kategorisierende Aussagen verwendet.
Der Meetingausschnitt für die selbstständige Codierung hat eine Länge von 03:20 Minuten. Bei dieser Aufnahme wurde speziell darauf geachtet, dass es sich nahezu um eine Aussagensequenz des Meetingskriptes handelt. Das hat zur Folge, dass einige Kategorien wesentlich öfter auftreten als andere, so wie es in einem echten Meeting auch der Fall ist. Das Meetingprofil ist in Abbildung 5.2 dargestellt. Bei Betrachtung fällt der hohe Anteil der Kategorie *Informationsweitergabe und Wissenstransfer* ins Auge. Dies ist darauf zurückzuführen, dass es sich bei diesem Meeting um eines der initialen Meetings des Projektes handelt. Es werden Anforderungen besprochen und Informationen untereinander ausgetauscht. *Kollegiales Verhalten* trat in diesem Ausschnitt an keiner Stelle auf.

5.2.2 Experimentsubjekte

Die Zielgruppe sind unerfahrene Meetingbeobachter mit Ahnung von Softwareprojekten. Diese Attribute spiegeln sich in Masterstudenten der Informatik der Leibniz Universität Hannover wider, sowie auch in kürzlich absolvierten Informatikstudenten der Leibniz Universität Hannover. Diese Personen eignen sich sehr gut als Repräsentanten der Zielgruppe, da sie keine bis wenig Erfahrung im Analysieren von Meetings haben. Softwareprojekterfahrung sammeln Studierende der Informatik laut Regelstudienplan unter anderem im fünften Semester des Bachelorstudiums durch die Teilnahme am *Softwareprojekt*, sodass auch Bachelorstudenten nach erfolgreicher Teilnahme geeignete Repräsentanten verkörpern. Da die Veranstaltung verpflichtend ist, können bei allen zuvor genannten Personengruppen Kenntnisse diesbezüglich vorausgesetzt werden.
Ehemalige Informatikstudenten oder andere Personengruppen, die bereits seit längerem in der Wirtschaft tätig sind, werden als Probanden in diesem Experiment außen vor gelassen. Grund ist, dass gesammelte Erfahrungen und Vorkenntnisse der einzelnen Personen sehr variabel sein können und diese einen nicht quantifizierbaren Einfluss auf die Güte einer Meetinganalyse haben können. Eine mögliche Folge wäre, eine Verbesserung in der Meetinganalyse nicht uneingeschränkt dem zuvor durchgeführtem Tutorial zuordnen zu können, da diese auch das Resultat individueller Vorkenntnisse sein könnte.
Um eine Vergleichbarkeit zu gewährleisten, wurden somit nur Bachelorstudenten nach erfolgreicher Teilnahme an der Veranstaltung *Softwareprojekt*, Masterstudenten und kürzlich absolvierte Studenten der Informatik der Leibniz Universität Hannover als Probanden in Betracht gezogen.

Die Akquisition der Studienteilnehmer erfolgte über persönliche Kontakte.
Alle Probanden nahmen auf freiwilliger Basis und ohne Zahlung eines Entgeldes oder einer sonstigen Aufwandsentschädigung an dem Experiment teil.

5.2.3 Design des Experimentes

Das Experiment wurde kontrolliert durchgeführt. Es nahmen insgesamt 30 Probanden teil. Diese wurden in drei Gruppen aufgeteilt. Die erste und zweite Gruppe waren jeweils Treatmentgruppen. Die Bedeutungen der Gruppen sind zur Übersicht in der Tabelle 5.1 aufgeführt. Die erste Treamentgruppe, folgend als Treatmentgruppe 1 bezeichnet, beschäftigte sich im Vorfeld mit der *Vorcodierten Meetinganalyse* und die zweite, folgend als Treatmentgruppe 2 bezeichnet, führte das *Quiz*-Tutorial vor der Analyse durch. Die dritte Gruppe wird im Folgenden als Kontrollgruppe bezeichnet und ist jene, die die Meetinganalyse durchführte, ohne zuvor ein Tutorial gemacht zuhaben.

Gruppe	Bedeutung
Treatmentgruppe 1	Durchführung des Tutorials *Vorcodierte Meetinganalyse* vor der Analyse
Treatmentgruppe 2	Durchführung des *Quiz*-Tutorials vor der Analyse
Kontrollgruppe	Die Analyse wird ohne Tutorial gestartet

Tabelle 5.1: Bedeutung der Gruppen

Um Balancing zu gewährleisten, wurden jeder Gruppe gleich viele Teilnehmer zugewiesen. Die Einteilung erfolgte zufällig, indem die Probanden nach der Reihenfolge ihrer Teilnahme am Experiment in eine Gruppe einsortiert wurden. Tabelle 5.2 stellt dieses Vorgehen dar. Demzufolge wurde Proband 1 der Treatmentgruppe 1 zugeteilt, Proband 2 der Treatmentgruppe 2 und Proband 3 der Kontrollgruppe. Nach diesem Schema wurde verfahren, bis zuletzt Proband 30 in die Kontrollgruppe eingeteilt wurde.
Es handelt sich um ein Between-Group-Design. Jedes Subjekt dieser Studie wurde genau einer der drei Gruppen zugeteilt und führte genau ein Verfahren durch. Da es das Ziel dieses Experimentes war, einen Lerneffekt zu adressieren, war es nicht möglich, ein Subjekt zwei Verfahren durchführen zu lassen. Da beim ersten Durchlauf der Proband bereits Erfahrungen mit der *act4teams-SHORT*-Analyse sammelt, kann nach einem zweiten durchgeführten Verfahren ein Lerneffekt diesem Verfahren nicht eindeutig zugeordnet werden.

	Treatment-gruppe 1	Treatment-gruppe 2	Kontroll-gruppe
1. Proband	x		
2. Proband		x	
3. Proband			x
4. Proband	x		
5. Proband		x	
6. Proband			x
7. Proband	x		
8. Proband		x	
9. Proband			x
10. Proband	x		
11. Proband		x	
12. Proband			x
13. Proband	x		
14. Proband		x	
15. Proband			x
16. Proband	x		
17. Proband		x	
18. Proband			x
19. Proband	x		
20. Proband		x	
21. Proband			x
22. Proband	x		
23. Proband		x	
24. Proband			x
25. Proband	x		
26. Proband		x	
27. Proband			x
28. Proband	x		
29. Proband		x	
30. Proband			x

Tabelle 5.2: Gruppenzuteilung der Probanden

5.3 Durchführung des Experimentes

In diesem Kapitel wird die Durchführung des Experimentes beschrieben. Dafür wird zu Beginn auf die Experimentumgebung und deren Besonderheit eingegangen. Danach wird der Ablauf des Experimentes vorgestellt.

5.3.1 Experimentumgebung

Aufgrund der COVID-19-Pandemie und dem damit verbundenen Erlass des Kontaktverbots wurde das Experiment mit den Probanden online durchgeführt.

Voraussetzung für die Teilnahme am eigenen Computer war ein Windows-Betriebssystem und eine Java Version 7 (ab dem Update 6[1]), 8, 9 oder 10[2]. Beim Vorhandensein einer neueren Version von Java wurde zusätzlich die Standalone JavaFX Runtime[3] benötigt, da JavaFX ab dem JDK 11 nicht mehr inbegriffen ist[4]. Über Dropbox[5] wurden die Jar-Datei mitsamt der anderen Unterlagen verteilt und zur Kommunikation wurde Skype[6] genutzt.
Als ersten Schritt ließ ich den Teilnehmern die Jar-Datei der *act4teams-SHORT*-Anwendung mit integrierten Tutorials via Dropbox zukommen. Auch die Übersicht, Einverständniserklärung und Fragebögen wurden via Dropbox dem Nutzer zugesandt. Die Durchführung des Experimentes fand einzeln mit jedem Teilnehmer über Skype statt. Durch die Skype-Audioverbindung und einem zusätzlich geteiltem Bildschirm seitens des Teilnehmers konnte ich die Interaktionen verfolgen und aufkommende Fragen beantworten.

5.3.2 Ablauf des Experimentes

Die Abbildung 5.3 zeigt eine Übersicht über den sequentiellen Ablauf des Experimentes für jede der drei Gruppen. Diese ist eine Zusammenfassung der im folgenden ausführlich beschriebenen Schritte der Studie.

Zu Beginn des Experimentes wurden den Teilnehmern eine Einverständniserklärung und eine Übersicht (vgl. Anhang S.90-93) des Studienablaufs vorgelegt. In der Einverständniserklärung wurden den Probanden die Art der Datenaufnahme- und verarbeitung, sowie die Risiken und Vorzüge, Vergütung, Vertraulichkeit und die Möglichkeit des jederzeitigen Rücktritts vom Experiment vorgestellt. In der Übersicht wurde der konkrete Ablauf der Studie beschrieben. Um es den Teilnehmern zu erleichtern, sich in die Situation der zukünftigen Nutzer hineinzuversetzen, wurde ein Bezug zum Kontext der Masterarbeit gegeben. Nachdem die Teilnehmer diese Informationen zur Kenntnis genommen haben und alle aufkommenden Fragen geklärt wurden, folgte die Unterzeichnung der Einverständniserklärung.

[1] Update Release Notes finden sich unter https://www.oracle.com/technetwork/java/javase/7u6-relnotes-1729681.html.
[2] https://www.java.com/de/download/
[3] https://gluonhq.com/products/javafx/
[4] Informationen zur Auslagerung von JavaFX aus dem JDK 11 finden sich unter https://blogs.oracle.com/java-platform-group/the-future-of-javafx-and-other-java-client-roadmap-updates.
[5] https://www.dropbox.com/install
[6] https://www.skype.com/de/get-skype/

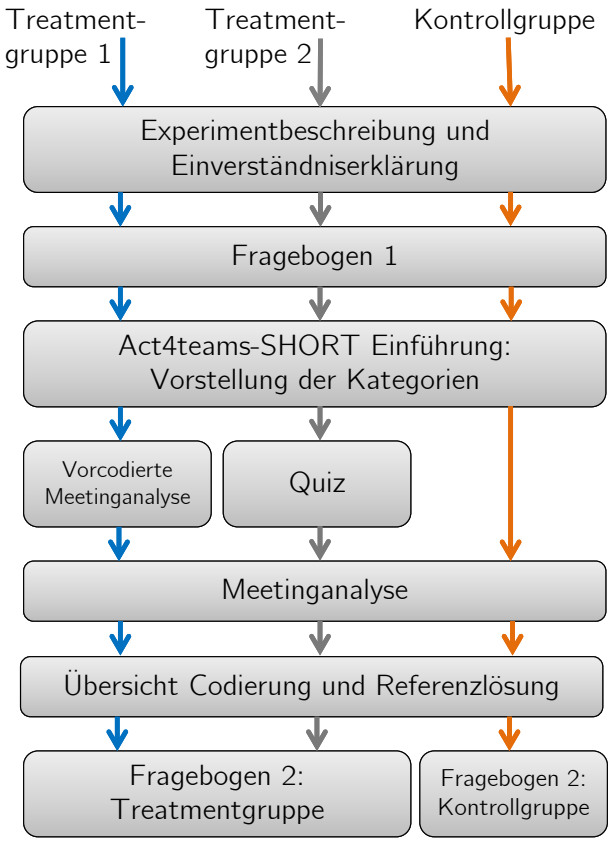

Abbildung 5.3: Experimentablauf der drei Gruppen

Bevor das Experiment startete, wurden die Teilnehmer gebeten einige Informationen auf dem Fragebogen 1 (vgl. Anhang S.94) anzugeben. Wie auch schon zu Beginn der Studie in der Übersicht zum Experimentablauf (vgl. Anhang S.90-92) erwähnt, wurden die Probanden ein zweites Mal darauf hingewiesen, dass die Bereitstellung dieser Informationen über ihre Person freiwillig sind und alle Angaben anonym behandelt werden. Ziel des Fragebogens war es, Kenntnisse über den Erfahrungsstand in der Softwareentwicklung, Teamarbeit, Softwareentwicklung im Team und in der Meetinganalyse der Probanden zu gewinnen. Diese Informationen halfen bei der Interpretation der Ergebnisse Rückschlüsse von Experimentergebnissen auf Vorkenntnisse der Probanden zu ziehen.
Hinzu wurde auf dem Fragebogen zur Angabe der Muttersprache gebeten, da der zu analysierende Meetingausschnitt nur in deutscher Sprache zur Verfügung steht. Für Teilnehmer, deren Muttersprache nicht Deutsch ist, hätte dies eine zusätzliche Schwierigkeit bedeuten können. Durch längere Überlegungen bei der Kategorisierung einer Aussage wäre eine mögliche Auswirkung die verspätete Setzung eines Tags. Ein

guter Sprachgebrauch ist eine nicht zu unterschätzende Eigenschaft für die korrekte Interpretation einer Aussage.[5]

Im nächsten Schritt begann die Interaktion mit dem *act4teams-SHORT*-Tool. Die Teilnehmer wählten das ihnen eingeteilte Verfahren aus. Unabhängig davon, welches Verfahren ausgewählt wurde, bekamen die Probanden eine Übersicht über die *act4teams-SHORT*-Kategorien, ausgenommen der Kategorie *Sonstiges*. Die aufgeführten Kategorien wurden den Teilnehmern anhand von Buzzword-Definitionen vorgestellt und mithilfe eines Beispiels veranschaulicht. Dafür konnten sich die Probanden bis zu fünf Minuten Zeit nehmen.

Der nächste Schritt wurde in Abhängigkeit zur Gruppeneinteilung ausgeführt.

- **Treatmentgruppe 1: Durchführung des Tutorials *Vorcodierte Meetinganalyse***
  Die Teilnehmer der Treatmentgruppe 1 führten das Tutorial *Vorcodierte Meetinganalyse* durch. Dafür hörten diese sich zu Beginn einen auditiven Meetingausschnitt an. Die Codierung konnte währenddessen dem Aufleuchten der Kategorien entnommen werden. Nach dem Ablauf der Audiodatei bekamen die Probanden bis zu fünf Minuten Zeit, sich selbständig mit den gesetzten Tags zu beschäftigen.
- **Treatmentgruppe 2: Durchführung des Tutorials *Quiz***
  Die Teilnehmer der Treatmentgruppe 2 führten das Tutorial *Quiz* durch. Insgesamt kategorisierten die Teilnehmer 30 Aussagen. Dafür wurden den Probanden eine Aussage nach der anderen vorgestellt. Jede Aussage sortierten die Teilnehmer nach eigenen Überlegungen in eine der neun Kategorien ein. Nach Eingabe einer Antwort wurde ihnen die entsprechende Lösung dazu aufgezeigt. Nachdem alle Aussagen kategorisiert wurden, bekamen die Probanden bis zu fünf Minuten Zeit, sich diese ein weiteres Mal anzuschauen.
- **Kontrollgruppe: Kein Tutorial**
  Die Teilnehmer der Kontrollgruppe bekamen kein Tutorial zugewiesen. Sie mussten als nächsten Schritt direkt mit der Meetinganalyse fortfahren.

Der nächste Schritt wurde wieder von allen Gruppen ausgeführt. Ein Meetingausschnitt der Länge 03:20 Minuten wurde von den Probanden selbständig codiert.

Nach der Meetinganalyse konnten die Probanden ihre Ergebnisse mit der Referenzlösung, visualisiert in einem Diagramm, vergleichen (vgl. Kapitel 4.3). Dazu wurden die von dem Probanden gesetzten Tags einer Kategorie unter denen der Referenzlösung

über die Zeit dargestellt. Durch die Verknüpfung von Diagramm und Audiodatei des Meetingausschnitts konnten sich die Probanden von einem Tag direkt zur Stelle in der Datei navigieren. Die Teilnehmer hatten Gelegenheit, bis zu fünf Minuten die Ergebnisse miteinander zu vergleichen und diese in der Audiodatei nachzuhören.

Abschließend wurden die Teilnehmer gebeten einen zweiten Fragebogen ausfüllen (vgl. Anhang S.95). Der Fragebogen wurde dafür in zwei Versionen angefertigt, eine Version für die Kontrollgruppe und die andere Version für die Treatmentgruppen. Beide Versionen können der Abbildung 5.4 entnommen werden.

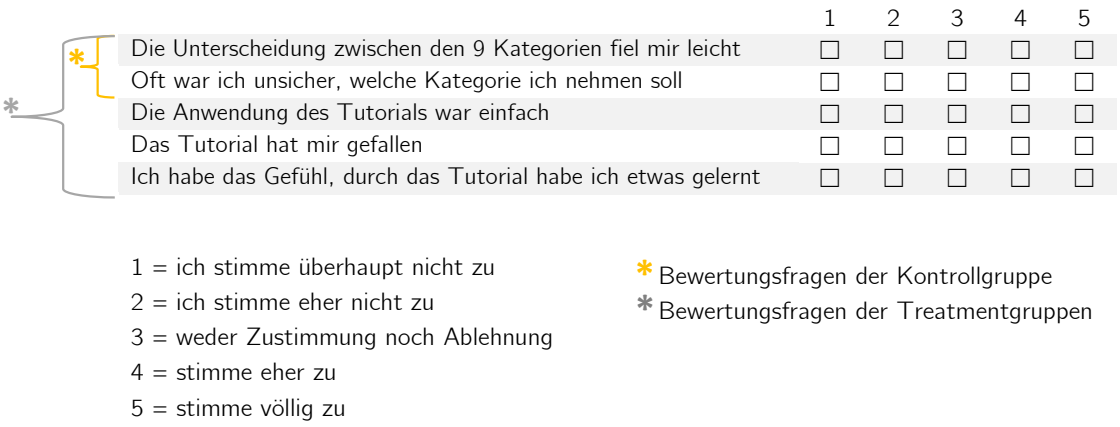

Abbildung 5.4: Bewertungsfragen des Fragebogen 2

- **Fragebogen 2 - Treatmentgruppen**
  Die Teilnehmer der Treatmentgruppen wurden gebeten einzuschätzen, wie sicher sie sich im Umgang mit der Kategorisierung der Aussagen gefühlt haben. Zudem sollten sie mitteilen, ob sie das Tutorial als einfach empfanden und ob es ihnen gefallen hat.
- **Fragebogen 2 - Kontrollgruppe**
  Die Teilnehmer der Kontrollgruppe wurden gebeten, einzuschätzen, wie sicher sie sich im Umgang mit der Kategorisierung der Aussagen gefühlt haben.

5.3.3 Zeitraum und Dauer

Die Studie wurde im Mai 2020 durchgeführt. Die durchschnittliche Durchführungsdauer betrug 30 Minuten je Proband der Treatmentgruppe 1 mit dem Tutorial *Vorcodierte Meetinganalyse*, 35 Minuten je Proband der Treatmentgruppe 2 mit dem *Quiz*-Tutorial und 20 Minuten je Teilnehmer der Kontrollgruppe ohne Tutorial.

5.4 Analyse und Interpretation

In diesem Kapitel wird die Analyse der Daten und die Interpretation der Ergebnisse beschrieben. Dafür wird zuerst anhand von deskriptiven Statistiken die Population beschrieben. Danach wird darauf eingegangen, wie die Daten nach der Experimentdurchführung aufbereitet wurden. Im Anschluss wird beschrieben, wie die Datenanalyse erfolgte, und zuletzt werden die durch die Datenanalyse resultierten Ergebnisse präsentiert.

5.4.1 Deskriptive Statistiken der Probanden

Unter den 30 Probanden nahmen 19 Masterstudierende der Informatik der Leibniz Universität Hannover und 11 Masterabsolventen der Informatik der Leibniz Universität Hannover, die kürzlichen ihren Abschluss gemacht haben, teil. Alle Teilnehmer gaben Deutsch als ihre Muttersprache an.
Aufgrund der zufälligen Zuordnung der Probanden zu einer der Experimentgruppen nach der Reihenfolge ihrer Teilnahme gab es keinen Einfluss auf die gleichmäßige Anzahl von Masterstudierenden und Absolventen in einer Gruppe. Die zufällige Aufteilung kann der Tabelle 5.3 entnommen werden.

	Treatmentgruppe 1	Treatmentgruppe 2	Kontrollgruppe
Absolventen	4	3	4
Masterstudierende	6	7	6
insgesamt	10	10	10

Tabelle 5.3: Anzahl der Absolventen und Masterstudierenden pro Experimentgruppe

Dabei ist zu erkennen, dass trotz der Aufteilung nach der Teilnahmereihenfolge keine großen Unterschiede in der Verteilung vorliegen. Treatmentgruppe 1 und die Kontrollgruppe beinhalten jeweils vier Absolventen und sechs Masterstudierende. Die Treatmentgruppe 2 hingegen drei Absolventen und 6 Masterstudierende.
Tabelle 5.4 und 5.5 listen die Aspekte der Likert-Skala des Fragebogen 1 auf und zeigen die Mediane der Antworten der Likert-Skala gruppiert nach Experimentgruppen (vgl. Abbildung 5.5) oder Masterstudierende, Absolventen und der gesamten Population (vgl. Abbildung 5.4) pro Aspekt.

Der Median über alle Probanden zeigt, dass diese nach eigenen Einschätzungen weder viel noch wenig Projekterfahrung haben (Zustimmungsgrad 3 der Likert-Skala). Werden nur die Aussagen der Masterstudierenden betrachtet, deckt sich der Median mit dem der

Aspekt	Median insgesamt	Master Studierende	Absolventen
Ich habe viel Projekterfahrung	3	3	4
Ich habe schon oft im Team gearbeitet	4	3	4
Ich habe schon Software im Team entwickelt	4	4	4
Ich habe noch nie ein Meeting analysiert	5	5	5

Tabelle 5.4: Mediane der Likert-Skala Einschätzungen sortiert nach Abschluss

gesamten Population. Die Absolventen hingegen stimmten der Aussage „Ich habe viel Projekterfahrung" eher zu (Zustimmungsgrad 4 der Likert-Skala). Bei der Aussage „Ich habe schon oft im Team gearbeitet" sehen die Einschätzungen der Probanden ähnlich aus. Auch dieser stimmten die Absolventen eher zu (4) und die Masterstudierenden lehnten diese weder ab noch gaben dieser Zustimmung (3). Auf die gesamte Population betrachtet, wurde der Aussage eher zugestimmt (4). Die Mediane über alle Probanden, der Masterstudierenden und Absolventen decken sich in den letzten beiden Aussagen. Bei der Aussage „Ich habe schon oft Software im Team entwickelt" lag der Median beim Zustimmungsgrad 4 (stimme eher zu). Der Aussage „Ich habe noch nie ein Meeting analysiert" stimmten alle Probanden voll zu (5).

Die Tabelle 5.5 zeigt die Mediane der Antworten pro Aspekt gruppiert nach Treatmentgruppe 1, Treatmentgruppe 2 und der Kontrollgruppe.

Aspekt	Median Treatment-gruppe 1	Treatment-gruppe 2	Kontroll-gruppe
Ich habe viel Projekterfahrung	3	4	3,5
Ich habe schon oft im Team gearbeitet	4	4	3,5
Ich habe schon Software im Team entwickelt	4	4	4
Ich habe noch nie ein Meeting analysiert	4,5	5	5

Tabelle 5.5: Mediane der Likert-Skala Einschätzungen sortiert nach Experimentgruppe

Dabei fällt auf, dass die Probanden der Treatmentgruppe 1 nach eigenen Angaben weder besonders viel noch besonders wenig Projekterfahrung haben (3). Die Teilnehmer der Treatmentgruppe 2 stimmten dieser Aussage eher zu (4), obwohl in dieser ein Absolvent weniger vertreten war, und die Probanden in der Kontrollgruppe befinden sich mit ihren Einschätzungen zwischen dem Zustimmungsgrad 3 und 4. Die Aussage „Ich habe schon oft im Team gearbeitet" wurde von den Teilnehmern der Treatmentgruppen

mit dem Zustimmungsgrad 4 bewertet. Der Median über den Antworten der Personen der Kontrollgruppe lag bei dieser Aussage bei 3,5. Die Mediane über den Antworten im Bezug auf die Softwareentwicklung im Team decken sich bei allen Experimentgruppen. Dieser liegt bei den Probanden der Treatmentgruppe 1, Treatmentgruppe 2, sowie bei denen der Kontrolllgruppe bei dem Zustimmungsgrad 4. Der Aussage „Ich habe noch nie ein Meeting analysiert" stimmten die Probanden der Treatmentgruppe 2 und der Kontrollgruppe voll zu (5). Der Median über die Einschätzungen der Probanden der Treatmentgruppe 1 liegt bei 4,5.

5.4.2 Datenaufbereitung

Studienauswertung:			
Probleme:	1	0.25	
Lösungen:	1	0.25	
Verknüpfungen & Vernetzungen:	0	0.0	
Destruktives Verhalten:	0	0.0	
Proaktives Verhalten:	1	0.25	
Kollegiales Verhalten:	0	0.0	
Methodisch-strukturierendes Verhalten:	0	0.0	
Informationsweitergabe & Wissenstransfer:	1	0.25	
Sonstiges:	0	0.0	
Event	Hours	Minutes	Seconds
Informationsweitergabe & Wissenstransfer	0	0	24
Probleme	0	1	17
Lösungen	0	1	48
Proaktives Verhalten	0	2	24

Abbildung 5.5: Ausschnitt einer csv-Datei mit beispielhaft gesetzten Tags

Die Daten der Einschätzungen der Erfahrungen sowie der subjektiven Eindrücke der Meetinganalyse und der Tutorials wurden anhand von Fragebögen gesammelt. Die Daten wurden manuell extrahiert und entsprechend inhaltlich gruppiert. Die Meeting-codierung der Probanden wurde nach Abschluss der Meetinganalyse automatisch in einer csv-Datei abgelegt. In dieser befindet sich zum einen die totale Anzahl der Häufigkeiten jeder Kategorie, zum anderen eine Auflistung jedes Tags mit entsprechendem Zeitstempel chronologisch nach dessen zeitlichem Auftreten sortiert. Eine beispielhafte Darstellung kann der Abbildung 5.5 entnommen werden. Zu Demonstrationszwecken wurden nur vier Sinneinheiten während des Meetings codiert, beispielsweise wurde

zum Zeitpunkt 2 Minuten 24 Sekunden proaktives Verhalten festgehalten.
Für die anschließende Untersuchung nach einer Verbesserung in den Meetinganalysen wurden die totalen Häufigkeiten der Kategorien den Dateien manuell entnommen.

5.4.3 Datenanalyse

Zur Untersuchung, ob die Tutorials zu einer Verbesserung der Meetinganalyse von unerfahrenen Beobachtern führen, und zur Bestimmung des effektivsten Tutorials (H1, H2 und H3) wurden die von Probanden erhobenen Meetinganalysen miteinander verglichen. Eine Veranschaulichung des Vorgehens und insbesondere der dabei verwendeten statistischen Tests werden anhand der Abbildungen 5.6 und 5.7 dargestellt.

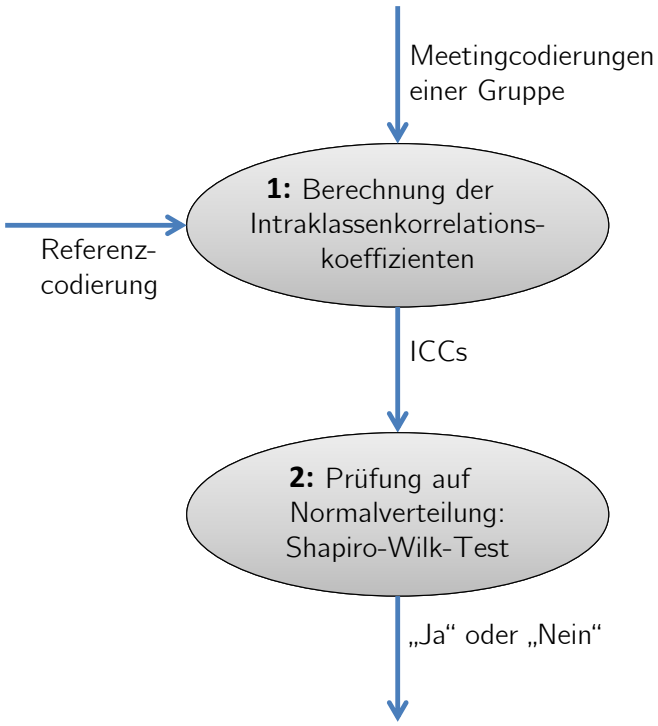

Abbildung 5.6: Part 1 zum Ablauf der Analyse von H1 - H3

Zuerst wurden mithilfe der Statistiksoftware IBM SPSS Statistics[7] die Intraklassenkorrelationskoeffizienten (ICCs) jeder Meetinganalyse mit der Referenzcodierung berechnet (Abbildung 5.6 Schritt 1). Die Referenzcodierung wurde von einer *act4teams-SHORT*-Expertin angefertigt. Der ICC bestimmt dabei die Übereinstimmung über alle Kategorien.

[7] https://www.ibm.com/de-de/analytics/spss-statistics-software

Anhand der ICCs lassen sich erste Aussagen über die Güte der Meetingcodierungen treffen. Zur Interpretation dieser wurde die Skala von Cicchetti [2] verwendet:

- ICC < 0,40 : schwach
- 0,40 <= ICC <= 0,59 : fair
- 0,60 <= ICC <= 0,74 : gut
- 0,75 < ICC : exzellent

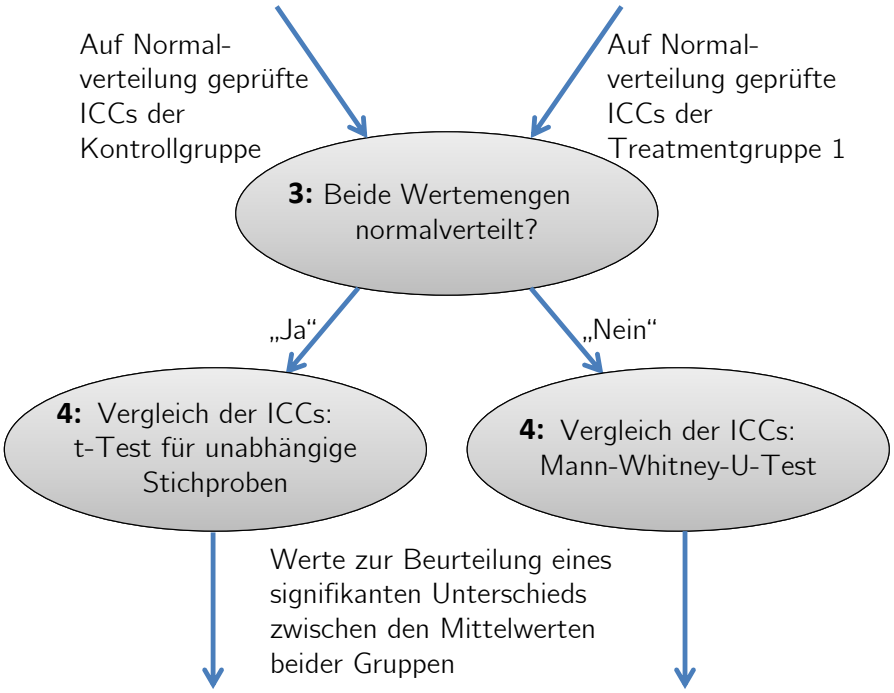

Abbildung 5.7: Part 2 zum Ablauf der Analyse am Beispiel von H1

Im Anschluss wurden die ICCs innerhalb der Experimentgruppen auf Normalverteilung mithilfe des Shapiro-Wilk-Tests[8] überprüft (Abbildung 5.6 Schritt 2). Diese Überprüfung war notwendig, um im nächsten Schritt den geeigneten Test zum Vergleich der ICC-Mengen zu bestimmen (Abbildung 5.7 Schritt 3). Für den Vergleich zweier normalverteilter ICC-Mengen wurde der t-Test für unabhängige Stichproben[9] gewählt. Zum Vergleich zweier ICC-Mengen, von denen mindestens eine keine Normalverteilung aufweist, wurde der Mann-Whitney-U-Test[10] verwendet (Abbildung 5.7 Schritt 4).

[8] Der Shapiro-Wilk-Test ist ein statistischer Signifikanztest, der die zugrundeliegende Grundgesamtheit einer Stichprobe auf Normalverteilung prüft [27].

[9] Der t-Test für unabhängige Stichproben testet, ob die Mittelwerte zweier unabhängiger Stichproben verschieden sind.

[10] Der Mann-Whitney-U-Test testet, ob die zentralen Tendenzen zweier unabhängiger Stichproben ver-

Um die Lerneffekte aus subjektiver Sicht zu analysieren, wurde das wahrgenommene Verständnis der Probanden in Bezug auf die Kategorien zwischen den Gruppen verglichen. Dazu wurde zur Angabe eines Zustimmungsgrades für die Aussage „Die Unterscheidung zwischen den 9 Kategorien fiel mir leicht" und „Oft war ich unsicher, welche Kategorie ich nehmen soll" auf einer Likert-Skala gebeten (H4). Hinzu wurden die Treatmentgruppen gebeten, drei weitere Angaben in Bezug auf die Tutorials zu machen. Dabei wurde die Zustimmung zu den Aussagen „Die Anwendung des Tutorials war einfach", „Das Tutorial hat mir gefallen" und „Ich habe das Gefühl, durch das Tutorial habe ich etwas gelernt" abgefragt. Die Angaben dienten als Grundlage, um die Tutorials aus subjektiver Sicht der Probanden bewerten zu können (H5, H6). Für die Analyse der Likert-Skala Bewertungen wurden zunächst alle Angaben gruppiert nach Aspekt und Experimentgruppe auf eine Normalverteilung mithilfe des Shapiro-Wilk-Tests überprüft. In Abhängigkeit davon wurden die Bewertungen zweier Gruppen entweder mit dem t-Test für unabhängige Stichproben oder dem Mann-Whitney-U-Test verglichen.

5.4.4 Ergebnisse

Im Folgenden werden die aus der zuvor beschriebenen Analyse resultierenden Ergebnisse vorgestellt.

5.4.5 Verbesserte Meetinganalysen

Die Tabelle 5.6 zeigt die errechneten Intraklassenkorrelationskoeffizienten zwischen den Analysen der Probanden und der Referenzcodierung gruppiert nach den Experimentgruppen.
Nach der Skala von Cicchetti [2] sind acht Werte der Kontrollgruppe als schwach einzuordnen. Lediglich ein Wert ist als fair und einer als gut einzustufen.
Dies impliziert, dass es den unerfahrenen Beobachtern große Schwierigkeiten bereitete, den Meetingausschnitt zu codieren. Bei der Betrachtung der Werte der Treatmentgruppe 2 in der Tabelle 5.6 wird eine Verbesserung der Meetinganlysen bereits anhand der farblichen Unterschiede deutlich. Nach der Skala von Cicchetti [2] sind nur zwei Werte als schwach zu bewerten. Hingegen indizieren vier Werte faire und weitere vier Werte gute Übereinstimmungen zwischen Probanden und Referenzcodierung. Auch die Werte der Treatmentgruppe 1 scheinen einen Lerneffekt der Probanden durch das Tutorial aufzuzeigen. Nach der Skala von Cicchetti [2] sind nur drei Werte als schwach zu verzeichnen.

schieden sind.

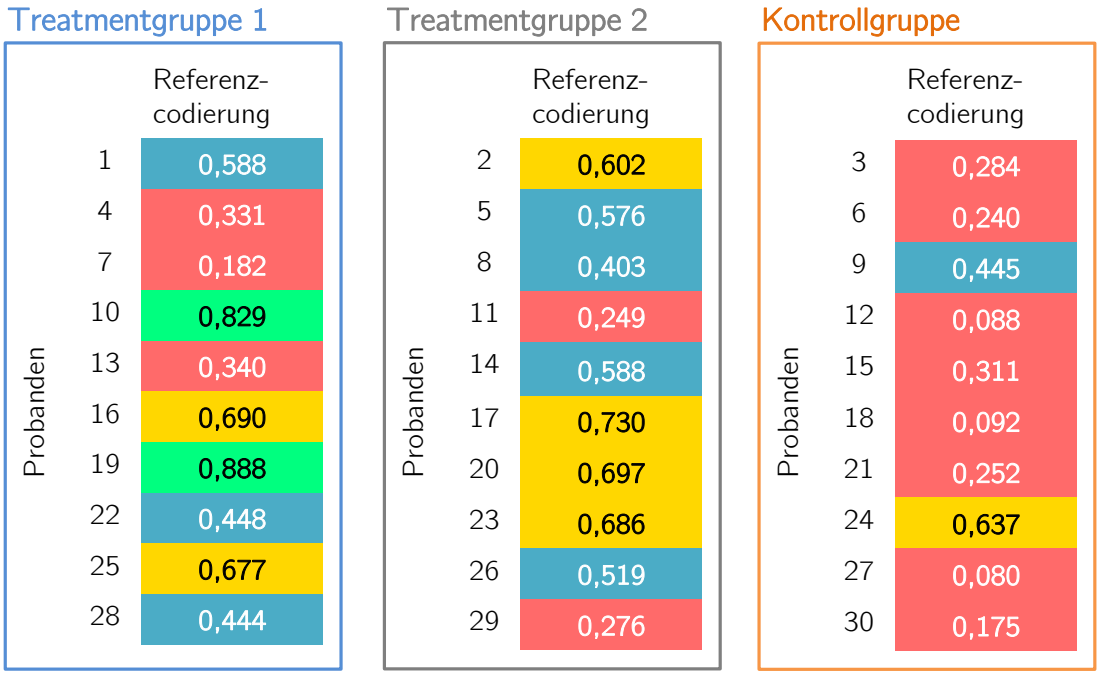

Tabelle 5.6: Werte der Inter-Rater-Reliabilität zwischen der Referenzcodierung und der Probanden der Treatmentgruppe 1 und 2 und der Kontrollgruppe

Drei Werte können als fair, zwei Werte als gut und sogar zwei Werte als exzellent eingestuft werden. Im Vergleich zur Treatmentgruppe 1 gibt es zwar einen schwachen Wert mehr, jedoch hat diese Gruppe keine exzellenten Werte vorzuweisen. Um die Daten auch quantitativ auszuwerten, wurden die Werte der Experimentgruppen zunächst mit dem Shapiro-Wilk-Test auf Normalverteilung untersucht.

	ICCs der Treatment-gruppe 1	ICCs der Treatment-gruppe 2	ICCs der Kontroll-gruppe
W	0,963	0,899	0,893
p	0,815	0,213	0,184
Normalverteilt?	Ja	Ja	Ja

Tabelle 5.7: Ergebnisse des Shapiro-Wilk-Tests

Die Ergebnisse können der Tabelle 5.7 entnommen werden. Daraus ist abzulesen, dass alle drei Wertemengen eine Normalverteilung aufweisen und so wurde folgend für die statistischen Analysen der Hypothesen $H1_0$, $H2_0$ und $H3_0$ der t-Test für unabhängige Stichproben angewendet, um die ICCs der Experimentgruppen untereinander auf signifikante Unterschiede der Mittelwerte zu untersuchen. Eine übersichtliche Darstellung

der Testergebnisse zeigt Tabelle 5.8. Für die Hypothese $H1_0$ wurden die ICCs der

	Vergleich der ICCs mit dem t-Test für unabhängige Stichproben		
	KG und TG 1	KG und TG 2	TG 1 und TG 2
Teststatistik t	-3,071	-3,514	0,1
Signifikanzwert p	0,007	0,002	0,921
Signifikanter Unterschied?	Ja	Ja	Nein

Tabelle 5.8: Ergebnisse des t-Tests für unabhängige Stichproben

Kontrollgruppe und die ICCs der Treatmentgruppe 1 mit dem t-Test für unabhängige Stichproben untersucht. Dieser ergab eine Teststatistik von t = -3,071 und dem dazugehörigen Signifikanzwert p = 0,007 und zeigte damit einen signifikanten Unterschied zwischen den Mittelwerten. Folglich kann die Hypothese $H1_0$ verworfen werden und $H1_1$ angenommen werden. Demnach hat die Durchführung des Tutorials *Vorcodierte Meetinganalyse* einen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung. Durch die Betrachtung der Tabelle 5.4 lässt sich eine Vermutung über die Art des Einflusses aufstellen. Nach der Skala von Cicchetti [2] sind acht ICCs der Kontrollgruppe als schwach einzustufen, bei der Treatmentgruppe 1 jedoch nur drei. Der Median der ICCs der Kontrollgruppe liegt bei 0,246 und der der Treatmentgruppe 1 bei 0,518. Daraus lässt sich ableiten, dass das Tutorial sich positiv auf die Übereinstimmung der Meetinganalysen der Probanden ausgewirkt hat.

Erkenntnis: Durch die Verwendung des Tutorials *Vorcodierte Meetinganalyse* werden bessere anfängliche Meetinganalysen erhoben, als wenn kein Tutorial verwendet wird.

Für die Überprüfung der Hypothese $H2_0$ wurde der t-Test für unabhängige Stichproben auf die ICCs der Kontrollgruppe und die ICCs der Treatmentgruppe 2 angewendet. Das Testergebnis ergab die Teststatistik t = -3,514 und den Signifikanzwert p = 0,002. Auch hier lässt sich ein signifikanter Unterschied zwischen den Mittelwerten feststellen, sodass folglich die Hypothese $H2_0$ abgelehnt und somit $H2_1$ entsprechend angenommen werden kann. Das bedeutet, dass eine vorausgehende Durchführung des *Quiz*-Tutorials einen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung hat. Der Median der ICCs der Kontrollgruppe liegt bei 0,246 und der der Treatmentgruppe 2 bei 0,582. Damit ist der Median der Treatmentgruppe 2 fast doppelt so hoch. Daher ist davon auszugehen, dass das Tutorial einen positiven Einfluss erwirkt.

Erkenntnis: Durch die Verwendung des *Quiz*-Tutorials werden bessere anfängli-

che Meetinganalysen erhoben, als wenn kein Tutorial verwendet wird.
Auch zur Untersuchung der Hypothese $H3_0$ wurde der t-Test für unabhängige Stichproben verwendet. Dieser wurde dafür auf die ICCs der Treatmentgruppe 1 und der der Treatmentgruppe 2 angewendet. Eine Teststatistik von t = 0,1 und ein Signifikanzwert von p = 0,921 resultierte. Den Testergebnissen zur Folge lässt sich kein signifikanter Unterschied zwischen den Mittelwerten feststellen und Hypothese $H3_0$ kann nicht verworfen werden.
Die Ergebnisse des t-Tests für unabhängige Stichproben, angewendet auf die beiden Treatmentgruppen, indizieren, dass die Wahl des Tutorials keinen Einfluss auf die Übereinstimmung zwischen einem unerfahrenen Beobachter und der Referenzcodierung hat. Bei der Betrachtung der Mediane fällt auf, dass der Median der ICCs der Probanden der Treatmentgruppe 1 bei 0,518 liegt und der der ICCs der Teilnehmer der Treatmentgruppe 2 bei 0,582 liegt. Letzterer liegt somit etwas über dem der Treatmentgruppe 1.
Erkenntnis: Die Verwendung des Tutorials *Vorcodierte Meetinganalyse* und des *Quiz*-Tutorials bewirken anscheinend eine Verbesserung der anfänglichen Meetinganalyse im gleichen Maße.

5.4.6 Vergleich der Tutorials

Zum Vergleich der subjektiven Wahrnehmung der Probanden in Bezug auf die Tutorials wurden zunächst deren Zustimmungsgrade in Bezug auf die Aussagen der Aspekte der Likert-Skala gruppiert nach Experimentgruppe und Aspekt mithilfe des Shapiro-Wilk-Tests auf Normalverteilung untersucht. Die Ergebnisse liefert die Tabelle 5.9.

Aspekt	Likert-Skala Werte TG1		TG 2		KG	
	p	n?	p	n?	p	n?
Die Unterscheidung zwischen den 9 Kategorien fiel mir leicht	0,008	Nein	0,000	Nein	0,015	Nein
Oft war ich unsicher, welche Kategorie ich nehmen soll	0,000	Nein	0,015	Nein	0,287	Ja
Die Anwendung des Tutorials war einfach	0,000	Nein	0,000	Nein		
Das Tutorial hat mir gefallen	0,036	Nein	0,008	Nein		
Ich habe das Gefühl, durch das Tutorial habe ich etwas gelernt	0,000	Nein	0,000	Nein		

Tabelle 5.9: Überprüfung auf Normalverteilung der Likert-Skala-Einschätzungen je Aspekt

Der Tabelle kann entnommen werden, dass alle Einschätzungswertemengen nicht normalverteilt sind mit Ausnahme der Menge zur Aussage „Oft war ich unsicher, welche Kategorie ich nehmen soll" der Kontrollgruppe. Da jedoch die Mengen zum gleichen Aspekt der Treatmentgruppen nicht normalverteilt sind, wurde folgend zur Überprüfung

von Unterschieden der zentralen Tendenzen zweier unabhängiger Stichproben ausschließlich der Mann-Whitney-U-Test verwendet. Im Folgenden werden die Ergebnisse des Mann-Whitney-U-Tests zwischen den jeweiligen Gruppen pro Aspekt beschrieben.

Aspekt 1: *Die Unterscheidung zwischen den 9 Kategorien fiel mir leicht.*
Tabelle 5.10 liefert die Ergebnisse des Mann-Whitney-U-Tests für den Aspekt 1.

	Vergleich der Likert-Skala Werte des Aspekts 1 mit dem Mann-Whitney-U-Test		
	KG und TG 1	KG und TG 2	TG 1 und TG 2
Z-Statistik	-2.496	-2.690	-0.171
Signifikanzwert p	0.019	0.011	0.912
Signifikanter Unterschied?	Ja	Ja	Nein

Tabelle 5.10: Ergebnisse des Mann-Whitney-U-Tests des Aspekts 1

Diesen Ergebnissen ist zu entnehmen, dass ein signifikanter Unterschied zwischen den wahrgenommen Schwierigkeiten in Bezug auf die Differenzierung der 9 Kategorien zwischen der Kontrollgruppe und der Treatmentgruppe 1 sowie der Kontrollgruppe und der Treatmentgruppe 2 besteht. Anhand eines Vergleiches der Mediane der Einschätzungen der drei Gruppen kann geschlussfolgert werden, dass sowohl durch die Durchführung des *Quiz*-Tutorials als auch durch die Durchführung des Tutorials *Vorcodierte Meetinganalyse* die Unterscheidung den Probanden zwischen den 9 Kategorien leichter fiel als denen, die kein Tutorial durchführten (Mediane: Kontrollgruppe: 2 ; Treatmentgruppe 1: 2,5 ; Treatmentgruppe 2: 3). Hierbei fällt jedoch auf, dass obwohl die Probanden durch die Tutorials etwas weniger Schwierigkeiten bei der Unterscheidung hatten, es ihnen jedoch trotzdem nicht leicht fiel.
Die Ergebnisse des Mann-Whitnes-U-Tests indizieren, dass die Wahl des Tutorials keinen Einfluss darauf hat, wie schwierig die Probanden die Unterscheidung der Kategorien fiel. Bei der Betrachtung der Mediane fällt auf, dass die Probanden der Treatmentgruppe 2 weder der Aussage zustimmten noch sie ablehnten. Dabei liegt der Median der Teilnehmer der Treatmentgruppe 1 zwischen dem Zustimmungsgrad 2 und 3 (*weder Zustimmung noch Ablehnung* und *ich stimme eher nicht zu*). Somit liegt der Median der Probanden der Treatmentgruppe 2 wenig höher, als der der Probanden der Treatmentgruppe 1.
Erkenntnis: Die Verwendung des Tutorials *Vorcodierte Meetinganalyse* oder des *Quiz*-Tutorials bewirkt, dass die Unterscheidung zwischen den Kategorien den unerfahrenen Beobachtern leichter fällt, als jenen ohne Verwendung eines Tu-

torials. Beide Tutorials scheinen dabei eine Erleichterung im gleichen Maße zu erwirken.

Aspekt 2: *Oft war ich unsicher, welche Kategorie ich nehmen soll.*
Tabelle 5.11 liefert die Ergebnisse des Mann-Whitney-U-Tests für den Aspekt 2.

	Vergleich der Likert-Skala Werte des Aspekts 2 mit dem Mann-Whitney-U-Test		
	KG und TG 1	KG und TG 2	TG 1 und TG 2
Z-Statistik	-2.463	-2.649	- 0.122
Signifikanzwert p	0.019	0.011	0.912
Signifikanter Unterschied?	Ja	Ja	Nein

Tabelle 5.11: Ergebnisse des Mann-Whitney-U-Tests des Aspekts 2

Hierbei wurden sehr ähnliche Ergebnisse wie zuvor bei der Analyse von Aspekt 1 erzeugt. Einen signifikanten Unterschied liefert der Test für die Kontrollgruppe und der Treatmentgruppe 1 sowie für die Kontrollgruppe und der Treatmentgruppe 2. Durch die Betrachtung der Mediane fällt auf, dass sich die Probanden durch die Durchführung eines Tutorials etwas sicherer gefühlt haben bei der Wahl einer Kategorie (Mediane: Kontrollgruppe: 4 ; Treatmentgruppe 1: 3 ; Treatmentgruppe 2: 3). Jedoch auch durch die Durchführung eines der Tutorials waren sich die Probanden nicht immer sicher, welche Kategorie die richtige ist. Der Test angewendet auf der Treatmentgruppe 1 und Treatmentgruppe 2 lieferte keinen signifikanten Unterschied der zentralen Tendenzen.

Sowohl anhand der Analyse von Aspekt 1 als auch anhand der Analyse von Aspekt 2 konnte gezeigt werden, dass die Durchführung eines Tutorials einen Einfluss auf die wahrgenommene Komplexität in Bezug auf die Unterscheidung der Kategorien beim unerfahrenen Nutzer hat. Die Hypothese $H4_0$ kann somit verworfen werden und $H4_1$ entsprechend angenommen werden.
Erkenntnis: Durch die Verwendung des Tutorials *Vorcodierte Meetingnanalyse* oder des *Quiz*-Tutorials sinkt bei den unerfahrenen Meetingbeobachtern die Unsicherheit, eine Aussage einer Kategorie zuzuordnen. Beide Tutorials scheinen dabei die Sicherheit im gleichen Maße zu fördern.

Tabelle 5.12 liefert die Ergebnisse des Mann-Whitney-U-Tests für die Aspekte 3, 4 und 5.

	Vergleich der Likert-Skala Werte zwischen der TG 1 und TG 2 mit dem Mann-Whitney-U-Test pro Aspekt		
	Aspekt 3	Aspekt 4	Aspekt 5
Z-Statistik	0.000	-0.630	-0.503
Signifikanzwert p	1.000	0.579	0.739
Signifikanter Unterschied?	Nein	Nein	Nein

Tabelle 5.12: Ergebnisse des Mann-Whitney-U-Tests für die Aspekte 3, 4 und 5

Aspekt 3: *Die Anwendung des Tutorials war einfach.*

Die Untersuchung der Likert-Skala-Werte der Treatmentgruppe 1 und der Treatmentgruppe 2 mit dem Mann-Whitney-U-Test lieferte keinen signifikanten Unterschied der zentralen Tendenzen beider Mengen. Folglich kann die Hypothese $H5_0$ nicht verworfen werden. Das bedeutet, dass die Wahl des Tutorials vermutlich keinen Einfluss auf die wahrgenommene Einfachheit der Nutzbarkeit hat. Die Mediane beider Wertemengen liegen bei 4 (stimme eher zu).
Erkenntnis: Beide Tutorials sind einfach anzuwenden.

Aspekt 4: *Das Tutorial hat mir gefallen.*

Die Ergebnisse liefern keinen signifikanten Unterschied der zentralen Tendenzen. Folglich kann die Hypothese $H6_0$ nicht verworfen werden. Das bedeutet, dass die Wahl des Tutorials vermutlich keinen Einfluss auf das Gefallen des Probanden hat. Beide Tutorials gefallen den Probanden somit vermutlich gleich gut (Mediane: Treatmentgruppe 1: 4 ; Treatmentgruppe 2: 4.5).
Erkenntnis: Beide Tutorials gefallen den Nutzern.

Aspekt 5: *Ich habe das Gefühl, durch das Tutorial habe ich etwas gelernt.*

Die Probanden wurden zusätzlich gefragt, ob sie das Gefühl haben, dass sie durch das jeweilige Tutorial etwas gelernt haben. Beide Gruppen stimmten dieser Aussage eher zu (Mediane: Treatmentgruppe 1: 4 ; Treatmentgruppe 2: 4). Einen signifikanten Unterschied der zentralen Tendenzen konnte durch die Durchführung des Mann-Whitney-U-Test nicht festgestellt werden. Die Ergebnisse indizieren somit, dass der wahrgenommene Lerneffekt der Probanden bei beiden Tutorials gleich hoch ist.
Erkenntnis: Nutzer beider Tutorials haben das Gefühl, durch diese etwas gelernt zu haben.

5.4.7 Zusammenfassung der Ergebnisse der Hypothesenprüfung

Die Tabelle 5.13 fasst alle Ergebnisse der formalen Hypothesenprüfung in einer Übersicht zusammen und zeigt, welche Rückschlüsse folglich gezogen werden können. Hypothese $H1_0$ konnte abgelehnt werden, sodass dementsprechend Hypothese $H1_1$ angenommen werden konnte. Gleiches gilt für die Hypothesen $H2_0$ und $H2_1$. Die Hypothesen $H3_0$, $H4_0$, $H5_0$ und $H6_0$ konnten nicht abgelehnt werden.

Hypothese	Abgelehnt ?	Schlussfolgerung
$H1_0$	✓	Eine vorausgehende Durchführung des Tutorials *Vorcodierte Meetinganalyse* **hat einen Einfluss** auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.
$H2_0$	✓	Eine vorausgehende Durchführung des *Quiz* Tutorials **hat einen Einfluss** auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.
$H3_0$	x	Welches der Tutorials durchgeführt wird, **hat vermutlich keinen Einfluss** auf die Übereinstimmung zwischen einem unerfahrenen Meetingbeobachter und der Referenzcodierung.
$H4_0$	x	Die Durchführung eines Tutorials **hat vermutlich keinen Einfluss** auf die wahrgenommene Komplexität in Bezug auf die Unterscheidung der Kategorien von einem unerfahrenen Meetingbeobachter.
$H5_0$	x	Welches der Tutorials durchgeführt wird, **hat vermutlich keinen Einfluss** auf die vom unerfahrenen Meetingbeobachter wahrgenommene Einfachheit der Nutzbarkeit.
$H6_0$	x	Welches der Tutorials durchgeführt wird, **hat vermutlich keinen Einfluss** auf das Gefallen vom unerfahrenen Meetingbeobachter.

Tabelle 5.13: Übersicht der Analyseergebnisse der Hypothesen

5.5 Einschränkungen der Validität

In diesem Abschnitt wurde die Bedrohung der Validität untersucht. Es wird darauf eingegangen, welche Aspekte möglicherweise die Validität der Ergebnisse einschränken und welche Maßnahmen ergriffen wurden, um den Bedrohungen der Validität entgegenzuwirken. Nachfolgend werden conclusion validity, internal validity, construct validity und external validity, aufgeteilt nach Wohlin et al. [29], untersucht.

5.5.1 Conclusion validity

Die Validität der Schlussfolgerung oder auch statistische Validität genannt befasst sich damit, ob die gewonnen Erkenntnisse aus einer statistisch korrekten Vorgehensweise

resultieren und zu welchem Grad diese somit annehmbar sind. Zu betrachtende Aspekte sind dabei unter anderem die Eignung der verwendeten statistischen Tests, sowie die Einhaltung der Voraussetzungen zur Verwendung der Tests. Des Weiteren spielt auch die Stichprobengröße eine Rolle. Eine Bedrohung für die statistische Validität könnten auch zufällige Irrelevanzen in der Experimentumgebung sein, wie beispielsweise Geräusche von außerhalb des Raumes oder plötzliche Unterbrechungen. [29]
Das Experiment wurde aufgrund des durch die COVID-19-Pandemie erlassenen Kontaktverbots online mit den Probanden durchgeführt. Dazu befanden sich die Teilnehmer zur Durchführung des Experimentes am heimischen Computer. Diese wurden gebeten sich während der Durchführung des Experimentes in einem ruhigen und von äußeren Einflüssen geschützten Raum aufzuhalten. Während des Experimentes war ich über eine Skype-Audioverbindung mit den Probanden verbunden. Dabei konnte ich feststellen, dass die Teilnehmer von keinen anderen Personen gestört worden sind. Mögliche störende Hintergrundgeräusche kann ich jedoch nicht ausschließen, sodass die Beeinflussung der Probanden durch zufällige Irrelevanzen in der Experimentumgebung in Betracht gezogen werden muss.
Durch die geringe Stichprobengröße sind die Ergebnisse nicht uneingeschränkt verallgemeinerbar. Da die Tutorials jedoch signifikante Effekte bei den Probanden aufwiesen, kann angenommen werden, dass es sinnvoll ist, ein Tutorial vor der ersten Meetinganalyse durchzuführen.
Des Weiteren wurden, um der Bedrohung der statistischen Validität entgegenzuwirken, für alle verwendeten Signifikanzstests die Erfüllung der Voraussetzung im Vorfeld geprüft und eingehalten. Somit wurde sichergestellt, dass die resultierenden Erkenntnisse auf statistisch korrekte Weise gewonnen wurden.

5.5.2 Internal validity

Bedrohungen der internen Validität sind Einflüsse, die sich auf die Ursächlichkeit einer unabhängigen Variable auswirken. Diese Einflüsse bedrohen die Schlussfolgerungen über eine mögliche kausale Beziehung zwischen Treatment und Ergebnis. [29]
Um die gemessenen Effekte tatsächlich der voherigen Durchführung eines Tutorials zuordnen zu können, wurde ein Between-Group-Design gewählt. Somit konnten Lerneffekte, die auf ein zuvor durchgeführtes Tutorial basieren, ausgeschlossen werden. Darüber hinaus gab es zu den Treatmentgruppen auch eine Kontrollgruppe, die eine neutrale Überprüfung der Ergebnisse ermöglichte.
Bei der Wahl der Probanden wurden nur bestimmte Personengruppen in Betracht gezogen, um Unterschiede in den Vorkenntnissen gering zu halten. Dennoch konnten

Unterschiede des Erfahrungsstandes zwischen Masterstudierenden und Absolventen, die kürzlich ihren Abschluss machten, ausgemacht werden. Die Absolventen stimmten den Aussagen ”Ich habe viel Projekterfahrung” und ”Ich habe schon oft im Team gearbeitet” eher zu, während die Masterstudenten beiden Aussagen weder zustimmten noch ablehnten (Median über alle Absolventen: 4 ; Median über alle Masterstudenten: 3). Aufgrund der zufälligen Zuordnung von Probanden anhand der Teilnahmereihenfolge zu einer Gruppe, gab es keinen Einfluss auf die gleichmäßige Verteilung von Masterstudierenden und Absolventen. Dies hat zur Folge, dass der Erfahrungsstand der Experimentgruppen nicht exakt übereinstimmt. So liegt beispielsweise der Median über alle Antworten der Treatmentgruppe 1 bei 4,5 (zwischen *stimme eher zu* und *stimme völlig zu*) zu der Aussage ”Ich habe noch nie ein Meeting analysiert” und der der beiden anderen Gruppen bei 5 (*stimme völlig zu*). Weitere Differenzen in den Medianen zu anderen Bewertungen von Aussagen zwischen den Gruppen sind Tabelle 5.3 zu entnehmen. Es handelt sich zwar um geringe Unterschiede, jedoch müssen diese bei der Bewertung der Ergebnisse berücksichtigt werden.

5.5.3 Construct validity

Die Konstruktvalidität befasst sich mit der Generalisierung der Experimentergebnisse auf das Konzept oder auf die Theorie hinter dem Experiment [29].
Ein Aspekt der Konstruktvalidität ist der *mono-method bias*. Dieser tritt auf, wenn nur eine einzige Methode zur Erhebung und Analyse von Daten verwendet wird. Dabei besteht das Risiko eines zweckwidrigen Experimentes, falls diese Methode einen Messfehler mit sich führt. Dem *mono-method bias* wurde entgegengewirkt, in dem zur Analyse sowohl objektive Daten aus der Meetinganalyse verwendet wurden, als auch Daten aus Fragebögen.
Ein weiterer Aspekt ist der *mono-operation bias*. Dieser tritt möglicherweise in diesem Experiment auf, da nur ein einziger Meetingausschnitt zur Analyse verwendet wurde. Dies war jedoch notwendig, um eine Vergleichbarkeit der Analysen sicherzustellen.
Der Aspekt *Interaction of different treatments* tritt auf, wenn ein Teilnehmer mehrere Treatments durchführt und somit Effekte nicht mehr uneingeschränkt einem Treatment zugeordnet werden können. Diesem wurde entgegengewirkt, indem jeder Proband nur einer Gruppe zugeordnet wurde.
Durch die Skype-Audioverbindung und dem zusätzlich geteilten Bildschirm seitens des Probanden konnte ich die Interaktionen verfolgen und es bestand die Möglichkeit, aufkommende Fragen zu stellen. Fragen wurden während der Experimentdurchführung nicht gestellt. Jedoch besteht die Möglichkeit, durch meinen direkten Einblick in das

Interaktionsgeschehen auf die Teilnehmer zusätzlichen Druck ausgeübt zu haben. Zudem könnten die Probanden auch bei der Ausfüllung der Fragebögen beeinflusst worden sein, da diese in einer direkten Verbindung zu mir zurückgesandt wurden und mir somit Rückschlüsse von den Daten auf die Personen möglich gewesen wären. Dieser Beeinflussung wurde mit der Zusicherung der anonymen Speicherung entgegengewirkt.

5.5.4 External validity

Die externe Validität befasst sich mit der Generalisierbarkeit der gewonnen Erkenntnisse. Eine Bedrohung der externen Validität ist dann gegeben, wenn sich die Bedingungen im Experiment und die realen Bedingungen unterscheiden.
Spätere Anwender sind Softwareentwickler, Teamleiter und Projektmanager mit unterschiedlichen Kenntnissen und Erfahrungen. Die Wahl der Probanden dieser Studie fiel aber auf Informatikstudenten und kürzlich absolvierte Informatikstudenten, da diese zum einen wenig Vorkenntnisse und Erfahrungen im Analysieren von Meetings haben und zum anderen vergleichbare Vorkenntnisse im Bereich der Softwareentwicklung und Teamarbeit haben. Damit unterscheiden sich die Probanden und die spätere Zielgruppe. Dies wurde jedoch so gewählt, damit Lerneffekte auf die Tutorials zurückgeführt werden konnten und nicht durch variable Erfahrungsmerkmale der Probanden unterstützt wurden. Für den späteren Praxiseinsatz sind unterschiedliche Vorkenntnisse jedoch kein Nachteil bei der Durchführung der Tutorials und so können diese uneingeschränkt von den Praktikern eingesetzt werden.

5.6 Zusammenfassung

Anhand des Experimentes mit 30 Probanden konnte die Anwendbarkeit und die Güte der Tutorialkonzepte gezeigt werden. **Es konnte eine deutliche Verbesserung der Meetinganalysen von Probanden, die zuvor ein Tutorial durchgeführt haben, festgestellt werden. Dabei führte sowohl das *Quiz*-Tutorial sowie auch das Tutorial *Vorcodierte Meetinganalyse* zu einem messbaren Lerneffekt. Während die Probanden in der Kontrollgruppe acht von zehn schwache Meetinganalysen erzeugten, so waren es bei der Treatmentgruppe 1 nur drei und bei der Treatmentgruppe 2 nur zwei.** Diese Erkenntnisse motivieren den Einsatz der Tutorials in der Praxis. **Jedoch fällt bei der Betrachtung der Analysen der Treatmentgruppen auch auf, dass lediglich zwei als exzellent zu bewerten sind.** Dazu können weitere Überlegungen vonnöten sein, um die Konzepte so zu optimeren, dass höhere Lerneffekte

hervorgerufen werden können, um damit mehr exzellente Meetinganalysen zu erzielen. **Die Probanden empfanden beide Konzepte als gut anwendbar und hatten Gefallen an diesen. Auch ein subjektiver Lerneffekt durch das jeweilige Tutorial konnte bei den Probanden verzeichnet werden.** Da im Vergleich beide Konzepte gleich gut bei den Probanden abschnitten, ist eine Kombinierung aus beiden denkbar.

5.7 Implikationen für die Anwendung der Tutorials

Die Ergebnisse des Experimentes motivieren den Einsatz der Tutorials in der Praxis. Um den Lerneffekt weiter zu erhöhen und dadurch noch bessere anfängliche Meetinganalysen hervorzurufen, ist eine Anpassung der Tutorials denkbar. Dabei können diese zum Beispiel um weitere Meetingausschnitte zur Übung erweitert werden. Eine andere Möglichkeit ist es, dass Nutzer beide Tutorials vor der ersten Meetinganalyse durchführen, in dem die Inhalte eines Tutorials abgeändert werden. Somit haben die Nutzer mehr Lernmaterial zur Verfügung, welches sich potenziell positiv auf die Erhöhung des Lerneffektes ausüben kann.
Auch wenn bei den Konzepten noch Verbesserungspotenzial da ist, können diese bereits in der aktuellen Version in der Praxis eingesetzt werden, da anhand der Experimentergebnisse subjektiv sowie auch objektiv ein Lerneffekt bei den Probanden nachgewiesen werden konnte.

6 Zusammenfassung und Ausblick

In diesem Kapitel werden die Vorgehensweise und Ergebnisse zusammengefasst. Hinzu werden die Forschungsfragen beantwortet. Abschließend werden im Abschnitt *Ausblick* Verbesserungsmöglichkeiten aufgezeigt.

6.1 Zusammenfassung

Das Ziel dieser Arbeit war es, zwei Tutorialkonzepte herauszuarbeiten, um die Schwierigkeiten bei der anfänglichen Analyse anhand des *act4teams-SHORT*-Tools zu reduzieren, sodass unerfahrene Meetingbeobachter nach Durchführung eines der Tutorials bessere Meetinganalysen erzeugen als jene unerfahrene Meetingbeobachter, die zuvor kein Tutorial durchgeführt haben. Zudem galt es herauszufinden, welches der beiden Tutorialkonzepte in Bezug auf den Lerneffekt sowie das subjektive Empfinden der Nutzer besser abschneidet.
Dazu wurden zu Beginn die Schwierigkeiten adressiert, indem Nutzer der *act4teams-SHORT*-Analyse nach diesen befragt wurden. Im Anschluss wurde jede Schwierigkeit untersucht und aus dieser passende Maßnahmen zur Vorbeugung abgeleitet. Aus diesen Maßnahmen wurden zwei Tutorialkonzepte entwickelt. Daraus ergibt sich die Antwort auf die *Forschungsfrage 1*, die untersucht, welche Art von Tutorials sich eignen, um Schwierigkeiten bei der anfänglichen Nutzung der *act4teams-SHORT*-Analyse zu reduzieren.

Antwort auf Forschungsfrage 1

Um Schwierigkeiten bei der anfänglichen Nutzung der *act4teams-SHORT*-Analyse zu reduzieren, eignet sich zum einen ein Tutorial, dass einen mit der Referenzcodierung angereicherten Softwareprojekt-Meetingausschnitt dem Nutzer präsentiert. Zum anderen eignet sich ein Quiz-Tutorial, in dem der User beispielhafte Aussagen eines Softwareprojekt-Meetings kategorisiert.

Nach der Erstellung der Tutorialkonzepte wurden beide Konzepte in die *act4teams-*

SHORT-Anwendung integriert. Folgend wurde ein Experiment mit 30 Probanden durchgeführt, die alle einen Softwareprojekt-Meetingausschnitt mit dem *act4teams-SHORT*-Tool zur Probe analysierten. Dafür wurden die Teilnehmer in drei Gruppen aufgeteilt. Zehn Probanden führten vor der Probeanalyse das Tutorial *Vorcodierte Meetinganalyse* durch, weitere zehn das *Quiz*-Tutorial und zehn führten die Probeanalyse durch, ohne zuvor ein Tutorial gemacht zu haben. Danach wurden die Meetingcodierungen mithilfe von statistischen Tests analysiert und die Analyseergebnisse der Gruppen miteinander verglichen. Anhand des Vergleiches kann eine Antwort auf die *Forschungsfrage 2* gegeben werden. Diese beschäftigt sich damit, wie ein Tutorial vor der ersten Interaktionsanalyse die resultierende Codierung eines unerfahrenen Beobachters beeinflussen kann.

Antwort auf Forschungsfrage 2

Beide Tutorialkonzepte haben die resultierende Codierung der ersten Interaktionsanalyse der Probanden positiv beeinflusst. Dabei konnte durch die Durchführung eines Tutorials in beiden Treatmentgruppen ein messbarer Lerneffekt bei den Probanden identifiziert werden.

Die *Forschungsfrage 3* beinhaltet die Fragestellung, welches der beiden Tutorials in Bezug auf den Lerneffekt und das subjektive Empfinden des Nutzers besser ist. Diese Aspekte wurden ebenso mithilfe des Experimentes untersucht und die resultierenden Ergebnisse analysiert und miteinander verglichen. Anhand des Vergleiches lässt sich die *Forschungsfrage 3* wie folgt beantworten:

Antwort auf Forschungsfrage 3

Beide Tutorials erzielten ähnlich gute Ergebnisse in Bezug auf den Lerneffekt, sowie auf das subjektive Empfinden der Nutzer.

Beide Tutorials erwiesen sich in allen untersuchten Punkten als hilfreich, jedoch kann keines der beiden präferiert werden.

6.2 Ausblick

Diese Arbeit diente dazu einen Ansatz zu entwickeln, wie Meetinganalysen von unerfahrenen *act4teams-SHORT*-Nutzern verbessert werden können. Die Ergebnisse

zeigen, dass durch beide entwickelten Tutorialkonzepte eine Verbesserung verzeichnet werden kann. Werden jedoch die einzelnen Übereinstimmungen der Analysen mit der Refernzcodierung betrachtet, fällt auf, dass lediglich zwei Analysen exzellente Übereinstimmungen aufweisen. Die Ergebnisse der 18 weiteren Probanden, die zuvor eines der beiden Tutorials durchführten, indizieren, dass noch weitere Lerneinheiten vonnöten sind. Von diesen 18 Ergebnissen weisen sogar fünf nur eine schwache Übereinstimmung mit der Referenzcodierung auf. Auch das subjektive Empfinden der Nutzer in Bezug auf die Unsicherheiten während der Codierung weist auf einen Bedarf nach mehr Lerninhalten hin. Die Verbesserungen der Tutorialkonzepte bieten daher Potenzial für weitere Forschungen.

Eine mögliche Verbesserung könnte die Kombination aus beiden Tutorials ergeben, indem die Inhalte eines Tutorials abgeändert werden würden. Dadurch würden mehr Lernmethoden auf eine Person angewendet werden, was potenziell zu einem höheren Lerneffekt führen könnte. Des Weiteren könnte das Tutorial *Vorcodierte Meetinganalyse* um weitere bereits analysierte Meetingausschnitte erweitert werden, um den Nutzern mehr Beispiele zu kategorisierten Aussagen präsentieren zu können. Bei dem *Quiz*-Tutorial besteht die Möglichkeit, die in schriftlicher Form dargestellten Aussagen durch eine auditive Unterstützung zu erweitern. So könnte der Nutzer sich jede Aussage vorsprechen lassen, um beispielsweise die Betonung in den Interpretationsprozess mit einfließen zu lassen. Eine weitere Möglichkeit wäre es, bei falscher Kategorisierung dem Nutzer Feedback zu geben aus welchen Gründen die Aussage zu der entsprechend richtigen Kategorie zuzuordnen ist. Beide Tutorials könnten hinzu noch in dem Maß erweitert werden, dass durch das Hovern mit der Maus über eine Kategorie in der Codierungsfläche eine Erklärung zu dieser angezeigt wird. Des Weiteren könnten beiden Tutorials durch zusätzliche Probeanalysen erweitert werden.

Literaturverzeichnis

[1] R. Beale und M. Sharples, *Design Guide for Developers of Educational Software*. Becta, 2002.

[2] D. V. Cicchetti, “Guidelines, Criteria, and Rules of Thumb for Evaluating Normed and Standardized Assessment Instrument in Psychology”, *Psychological Assessment*, Jg. 6, Nr. 4, S. 284–290, 1994.

[3] V. Elzen-Rump, D. Leutner und J. Wirth, “Lernstrategien im Unterrichtsalltag”, in *Diagnostizieren und Fördern in der Sekundarstufe I*, S. Kliemann, Hrsg. Cornelson Verlag, 2008, S.101–111.

[4] A.-C. Haak, *Analyse von Interaktionen in Softwareprojekt-Meetings*, Leibniz Universität Hannover, Fachgebiet Software Engineering, Bachelorarbeit, 2017.

[5] E. Hallsteinsdóttir, *Das Verstehen idiomatischer Phraseologismen in der Fremdsprache Deutsch*. Verlag Dr. Kovac, 2001.

[6] M. Hasselhorn und A. S. Labuhn, “Lernstrategien”, in *Pädagogische Interventionsforschung*, B. Hascher T. und Schmitz, Hrsg. Juventa Verlag, 2010, S. 73–84.

[7] S. Heidig, J. Muller und M. Reichelt, “Emotional design in multimedia learning: Differentiation on relevant design features and their effects on emotions and learning”, *Computers in Human Behavior*, Jg. 44, S. 81–95, 2015.

[8] D. Hewitt, *Understanding Effective Learning: Strategies for the Classroom*. Open University Press, 2008.

[9] H. Iwata, J. Shirogane und Y. Fukazawa, “Automatic Generation of Tutorial Systems from Development Specification”, in *Fundamental Approaches to Software Engineering*, L. Baresi und R. Heckel, Hrsg. Springer Berlin Heidelberg, 2006, S. 79–92.

[10] J. Jacobsen und L. Meyer, *Praxisbuch Usability und UX*. Rheinwerk Computing, 2019.

[11] S. Kauffeld und N. Lehmann-Willenbrock, “Meetings Matter Effects of Team Meetings on Team and Organizational Success”, *Small Group Research*, Jg. 43, Nr. 2, S. 130–158, 2012.

[12] S. Kauffeld, G. Tiscar-Lorenzo, K. Montasem und N. Lehmann-Willenbrock, “act4teams®: die nächste Generation der Teamentwicklung”, in *Handbuch Kompetenzentwicklung*. Schäffer-Poeschel, 2009, S. 191–215.

[13] J. Klünder, *Analyse der Zusammenarbeit in Softwareprojekten mittels Informationsflüssen und Interaktionen in Meetings*. Logos Verlag Berlin, 2019.

[14] J. Klünder, N. Prenner, A.-K. Windmann, M. Stess, M. Nolting, F. Kortum, L. Handke, K. Schneider und S. Kauffeld, “Do You Just Discuss or Do You Solve? Meeting Analysis in a Software Project at Early Stages”, in *IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ACM, 2020.

[15] J. Klünder, C. Unger-Windeler, F. Kortum und K. Schneider, “Team meetings and their relevance for the software development process over time”, in *Proceedings of the 43rd Euromicro Conference on Software Engineering and Advanced Applications*, IEEE, 2017, S. 313–320.

[16] N. Lehmann-Willenbrock, J. A. Allen und S. Kauffeld, “A Sequential Analysis of Procedural Meeting Communication: How Teams Facilitate Their Meetings”, *Journal of Applied Communication Research*, Jg. 41, 2013.

[17] N. Lehmann-Willenbrock, R. A. Meyers, S. Kauffeld, A. Neininger und A. Henschel, “Verbal Interaction Sequences and Group Mood: Exploring the Role of Team Planning Communication”, *Small Group Research*, Jg. 42, S. 639–668, 2011.

[18] A. Luong und S. Rogelberg, “Meetings and More Meetings: The Relationship Between Meeting Load and the Daily Well-Being of Employees”, *Group Dynamics: Theory, Research, and Practice*, Jg. 9, S. 58–67, 2005.

[19] S. Möller, *Quality Engineering*. Springer Berlin Heidelberg, 2017.

[20] W. Nasser D. ; McEwen, “The Impact of Alternative Media Channels: Recall and Involvement with Messages”, in *AV Communication Review*. Springer, 1976, Bd. 24, S.263–272, Nr.3.

[21] J. Nielsen, *Usability Engineering*. Morgan Kaufmann, 1994.

[22] G. M. Olson, J. S. Olson, M. R. Carter und M. Storrosten, “Small Group Design Meetings: An Analysis of Collaboration”, *Human–Computer Interaction*, Jg. 7, Nr. 4, S. 347–374, 1992.

[23] N. Prenner, *Analyse von direkter und dokumentenbasierter Kommunikation in agilen Entwicklerteams*, Leibniz Universität Hannover, Fachgebiet Software Engineering, Masterarbeit, 2017.

[24] N. Prenner, J. Klünder und K. Schneider, “Making meeting success measurable by participants’ feedback”, in *Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering*, Ser. SEmotion ’18, ACM, 2018, S. 25–31.

[25] K. Schneider, J. Klünder, F. Kortum, L. Handke, J. Straube und S. Kauffeld, “Positive affect through interactions in meetings: The role of proactive and supportive statements”, *Journal of Systems and Software*, Jg. 143, S. 59–70, 2018.

[26] N. M. Seel, “Learning and Thinking”, in *Encyclopedia of the Sciences of Learning*. Springer US, 2012, S. 1797–1799.

[27] S. S. Shapiro und M. B. Wilk, “An analysis of variance test for normality (complete samples”, *Biometrika*, Jg. 52, Nr. 3/4, S. 591–611, 1965.

[28] P. M. Tobia und M. C. Becker, “Making the Most of Meeting Time”, *Training and Development Journal*, Jg. 44, Nr. 8, 1990.

[29] C. Wohlin, P. Runeson, M. Höst, M. Ohlsson, B. Regnell und A. Wesslén, *Experimentation in Software Engineering*. Springer Berlin Heidelberg, 2012.

Abbildungsverzeichnis

2.1 Menü 11
2.2 Oberfläche der Datenerhebung 12
2.3 Kreisdiagramm einer Beispielanalyse 13
2.4 Zeitstrahl einer Beispielanalyse 13

4.1 Technische vs. Soft Skills 30
4.2 Papierprototyps des Probeanalyse-Interfaces 35
4.3 Papierprototyps des Vergleichs zwischen der Codierung des Nutzers und der Referenzcodierung anhand eines Zeitstrahls 36
4.4 Papierprototyp eines Beispielmeetingausschnitts mit Referenzcodierung . 37
4.5 Papierprototyps Quiz 39
4.6 Erklärungen zu den Kategorien 41
4.7 Interface des Tutorials *Vorcodierte Meetinganalyse* 42
4.8 Interface des *Quiz-Tutorials* 43
4.9 Interface zur Datenerhebung der Probeanalyse 44
4.10 Gegenüberstellung vom Nutzer gesetzter Tags zur Referenzcodierung über die Zeit 45

5.1 Referenzprofil des Meetingausschnitts der *Vorcodierten Meetinganalyse* . 51
5.2 Referenzprofil des selbständig zu codierenden Meetingausschnitts 51
5.3 Experimentablauf der drei Gruppen 56
5.4 Bewertungsfragen des Fragebogen 2 58
5.5 Ausschnitt einer csv-Datei mit beispielhaft gesetzten Tags 61
5.6 Part 1 zum Ablauf der Analyse von H1 - H3 62
5.7 Part 2 zum Ablauf der Analyse am Beispiel von H1 63

Tabellenverzeichnis

2.1 Zusammenhang zwischen den Kompetenzbereichen und -aspekten und Kategorien von *act4teams* [4], nach [11] 6
2.2 Einflüsse der *act4teams* Kategorien auf die Meetingergebnisse [4], nach [11] 8
2.3 Inhalt der Iterationen (in Anlehnung an [13]) 10

4.1 Schwierigkeiten bei der anfänglichen *act4teams-SHORT*-Nutzung mit entsprechenden Lösungsvorschlägen 32

5.1 Bedeutung der Gruppen 53
5.2 Gruppenzuteilung der Probanden 54
5.3 Anzahl der Absolventen und Masterstudierenden pro Experimentgruppe . 59
5.4 Mediane der Likert-Skala Einschätzungen sortiert nach Abschluss 60
5.5 Mediane der Likert-Skala Einschätzungen sortiert nach Experimentgruppe 60
5.6 Werte der Inter-Rater-Reliabilität zwischen der Referenzcodierung und der Probanden der Treatmentgruppe 1 und 2 und der Kontrollgruppe 65
5.7 Ergebnisse des Shapiro-Wilk-Tests 65
5.8 Ergebnisse des t-Tests für unabhängige Stichproben 66
5.9 Überprüfung auf Normalverteilung der Likert-Skala-Einschätzungen je Aspekt 67
5.10 Ergebnisse des Mann-Whitney-U-Tests des Aspekts 1 68
5.11 Ergebnisse des Mann-Whitney-U-Tests des Aspekts 2 69
5.12 Ergebnisse des Mann-Whitney-U-Tests für die Aspekte 3, 4 und 5 70
5.13 Übersicht der Analyseergebnisse der Hypothesen 71

Anhang

Übersicht zur Teilnahme am Experiment – Treatmentgruppe 1

Einsatz eines Tutorials zur besseren Meetinganalyse

Zuerst einmal möchte ich mich bei dir für deine Teilnahme an diesem Experiment bedanken. Im Rahmen meiner Masterarbeit mit dem Titel "Verbesserung der Meetinganalyse in Entwicklerteams unter Verwendung von praktischen Tutorials" initiiere ich dieses Experiment, um zwei von mir entwickelte Konzepte zur Anleitung zur Meetinganalyse auf ihre Güte und Anwendbarkeit zu prüfen. Alle erfassten Daten während des gesamten Experimentes werden anonym behandelt. Es wird nicht möglich sein von dir getätigte Angaben auf deine Person zurückzuführen. Damit du einen Überblick bekommst was dich während des Experimentes erwartet, folgt nun der Ablauf.

Zu Beginn würde ich gerne ein paar persönliche Daten von dir erfassen. Die Fragen richten sich dabei größtenteils auf deine bisherigen Erfahrungen in der Softwareentwicklung und im Analysieren von Meetings. Diese Angaben sind freiwillig auszufüllen, wären für meine spätere Auswertung aber von großem Interesse. So wird es mir möglich sein Rückschlüsse von Ergebnissen auf Erfahrungsmerkmale zu ziehen.

Das Experiment startet mit einer Übersicht über die act4teams-SHORT Analyse. Es werden dir acht Kategorien anhand von Buzzword Definitionen erklärt und mithilfe eines kurzen Beispiels veranschaulicht. Anhand dieser acht Kategorien plus Kategorie "Sonstiges" werden Aussagen während der Meetinganalyse klassifiziert.

Als nächstes startet ein Tutorial. Ein auditiver Meetingausschnitt wird abgespielt. Während du dir dieses anhörst, kannst du visuell die Codierung verfolgen. Dafür zeigt das Interface alle neun Kategorien, welche zum Zeitpunkt einer Klassifizierung farblich aufleuchten. Nach Ablauf der Audiodatei hast du Zeit beliebige Stellen der Audiodatei erneut abzuspielen. Unter dem Slider für die Audiodatei befindet sich ein Zeitstrahl, der alle Tags farblich entsprechend der Klassifizierung beinhaltet. Durch einen Klick auf einen Tag kannst du dich direkt zu der Stelle in der Audiodatei navigieren lassen. Zudem kannst du die Tags auf dem Zeitstrahl auch filtern, indem du Kategorien selektierst oder deselektierst. Du kannst dir bis zu fünf Minuten Zeit nehmen dafür.

Nach Beendigung des Tutorials wirst du selber einen auditiven Meetingausschnitt mit act4teams-SHORT kodieren. Jeder Tag, den du während des Codings setzt, wird mitgeloggt und in eine csv-Datei gespeichert.

Im Anschluss kannst du dir deine Lösung im Vergleich zu der Musterlösung in einem Diagramm anschauen. Dazu stehen die von dir gesetzten Tags einer Kategorie und die aus der Musterlösung direkt untereinander. Die Tags im Diagramm sind ebenfalls mit der Audiodatei verknüpft, sodass ein Klick auf einen Tag dich direkt zu der Stelle in der Audiodatei führt.

Abschließend werde ich dir einen zweiten Fragebogen aushändigen, mithilfe dessen ich Feedback zu dem Tutorial sammeln möchte.
Die Dauer des Experimentes beschränkt sich auf 30 Minuten.

Übersicht zur Teilnahme am Experiment – Treatmentgruppe 2

Einsatz eines Tutorials zur besseren Meetinganalyse

Zuerst einmal möchte ich mich bei dir für deine Teilnahme an diesem Experiment bedanken. Im Rahmen meiner Masterarbeit mit dem Titel "Verbesserung der Meetinganalyse in Entwicklerteams unter Verwendung von praktischen Tutorials" initiiere ich dieses Experiment, um zwei von mir entwickelte Konzepte zur Anleitung zur Meetinganalyse auf ihre Güte und Anwendbarkeit zu prüfen. Alle erfassten Daten während des gesamten Experimentes werden anonym behandelt. Es wird nicht möglich sein von dir getätigte Angaben auf deine Person zurückzuführen. Damit du einen Überblick bekommst was dich während des Experimentes erwartet, folgt nun der Ablauf.

Zu Beginn würde ich gerne ein paar persönliche Daten von dir erfassen. Die Fragen richten sich dabei größtenteils auf deine bisherigen Erfahrungen in der Softwareentwicklung und im Analysieren von Meetings. Diese Angaben sind freiwillig auszufüllen, wären für meine spätere Auswertung aber von großem Interesse. So wird es mir möglich sein Rückschlüsse von Ergebnissen auf Erfahrungsmerkmale zu ziehen.

Das Experiment startet mit einer Übersicht über die act4teams-SHORT Analyse. Es werden dir acht Kategorien anhand von Buzzword Definitionen erklärt und mithilfe eines kurzen Beispiels veranschaulicht. Anhand dieser acht Kategorien plus Kategorie "Sonstiges" werden Aussagen während der Meetinganalyse klassifiziert.

Als nächstes startet ein Tutorial. Es wird dir eine Aussage auf dem Screen präsentiert. Darunter stehen dir alle Kategorien als Antwortmöglichkeiten zur Verfügung. Sobald du die Aussage einer der Kategorien zugeordnet hast, wird dir die Lösung angezeigt. Als nächsten Schritt kannst du auf 'Weiter' klicken und die nächste Aussage erscheint. Insgesamt sind es 30 Aussagen zu kategorisieren. Wenn du damit fertig bist, hast du bis zu fünf Minuten Zeit alle Aussagen inklusive Lösungen dir ein weiteres Mal anzuschauen.

Nach Beendigung des Tutorials wirst du einen auditiven Meetingausschnitt mit act4teams-SHORT kodieren. Jeder Tag, den du während des Codings setzt, wird mitgeloggt und in eine csv-Datei gespeichert.

Im Anschluss kannst du dir deine Lösung im Vergleich zu der Musterlösung in einem Diagramm anschauen. Dazu stehen die von dir gesetzten Tags einer Kategorie und die aus der Musterlösung direkt untereinander. Die Tags im Diagramm sind mit der Audiodatei verknüpft, sodass ein Klick auf einen Tag dich direkt zu der Stelle in der Audiodatei führt.

Abschließend werde ich dir einen zweiten Fragebogen aushändigen, mithilfe dessen ich Feedback zu dem Tutorial sammeln möchte.

Die Dauer des Experimentes beschränkt sich auf 30 Minuten.

Übersicht zur Teilnahme am Experiment - Kontrollgruppe

Einsatz eines Tutorials zur besseren Meetinganalyse

Zuerst einmal möchte ich mich bei dir für deine Teilnahme an diesem Experiment bedanken. Im Rahmen meiner Masterarbeit mit dem Titel "Verbesserung der Meetinganalyse in Entwicklerteams unter Verwendung von praktischen Tutorials" initiiere ich dieses Experiment, um zwei von mir entwickelte Konzepte zur Anleitung zur Meetinganalyse auf ihre Güte und Anwendbarkeit zu prüfen. Alle erfassten Daten während des gesamten Experimentes werden anonym behandelt. Es wird nicht möglich sein von dir getätigte Angaben auf deine Person zurückzuführen. Damit du einen Überblick bekommst was dich während des Experimentes erwartet, folgt nun der Ablauf.

Zu Beginn würde ich gerne ein paar persönliche Daten von dir erfassen. Die Fragen richten sich dabei größtenteils auf deine bisherigen Erfahrungen in der Softwareentwicklung und im Analysieren von Meetings. Diese Angaben sind freiwillig auszufüllen, wären für meine spätere Auswertung aber von großem Interesse. So wird es mir möglich sein Rückschlüsse von Ergebnissen auf Erfahrungsmerkmale zu ziehen.

Das Experiment startet mit einer Übersicht über die act4teams-SHORT Analyse. Es werden dir acht Kategorien anhand von Buzzword Definitionen erklärt und mithilfe eines kurzen Beispiels veranschaulicht. Anhand dieser acht Kategorien plus Kategorie "Sonstiges" werden Aussagen während der Meetinganalyse klassifiziert.

Im nächsten Schritt wirst du anhand der act4teams-SHORT Kategorie ein auditives Meeting codieren. Dabei wird jeder Tag, den du während des Codings setzt, mitgeloggt und in eine csv-Datei gespeichert.

Im Anschluss kannst du dir deine Lösung im Vergleich zu der Musterlösung in einem Diagramm anschauen. Dazu stehen die von dir gesetzten Tags einer Kategorie und die aus der Musterlösung direkt untereinander. Die Tags im Diagramm sind mit der Audiodatei verknüpft, sodass ein Klick auf einen Tag dich direkt zu der Stelle in der Audiodatei führt.

Abschließend werde ich dir einen zweiten Fragebogen aushändigen, mithilfe dessen ich Feedback zur Analyse sammeln möchte.

Die Dauer des Experimentes beschränkt sich auf 20 Minuten.

Einverständniserklärung zur Teilnahme am Experiment

Einsatz eines Tutorials zur besseren Meetinganalyse

Um an der Studie teilzunehmen, lies dir bitte die folgenden Punkte sorgfältig durch. Bei Fragen wende dich gerne an mich.

Beschreibung: Die Beschreibung des Experimentes ist Inhalt der Übersicht zur Teilnahme am Experiment.

Datenaufnahme und –verarbeitung: Während der Meetinganalyse werden gesetzte Tags mitgeloggt und in einer Datei gespeichert. Hinzu werden Feedback und personenbezogene Daten mithilfe zweier Fragebögen erfasst. Alle Daten werden anonym behandelt, sodass anhand der Daten keine Rückschlüsse auf den Teilnehmer möglich sind. Der Teilnehmer stimmt der Erfassung, Speicherung und Verarbeitung der Meetingcodierung, sowie der Antworten der Fragebögen zu.

Risiken und Vorzüge: Das Experiment birgt weder Risiken noch entstehen für den Teilnehmer Vorzüge.

Vergütung: Die Teilnahme am Experiment erfolgt freiwillig. Es wird keine Vergütung für entstandenen Zeitaufwand gezahlt.

Vertraulichkeit: Die Daten werden anonym behandelt und verarbeitet. Der Teilnehmer ist zur Geheimhaltung der Studieninhalte verpflichtet.

Rücktritt vom Experiment: Der Abbruch des Experimentes ist jederzeit und ohne Angabe von Gründen möglich. Dadurch entsteht kein Nachteil für den Teilnehmer.

Ich habe die oben aufgeführten Punkte gelesen und verstanden. Mit meiner Unterschrift akzeptiere ich alle Inhalte der Einverständniserklärung und bestätige, dass ich an der Studie teilnehmen möchte.

Nachname, Vorname	Ort, Datum, Unterschrift

Fragebogen 1 (vor Experimentbeginn)

Vielen Dank für deine Teilnahme!
Zu Beginn würde ich dir gerne ein paar Fragen zu deinem Hintergrund stellen. Diese Angaben sind freiwillig, werden für mich aber einen großen Mehrwert bei der Interpretation der Studienergebnisse haben. Alle erfassten Daten werden anonym behandelt und verarbeitet.

Dein Studiengang? ☐ Informatik BSc., Fachsemester:_____
☐ Informatik MSc., Fachsemester:_____
☐ Ich bin kein Student mehr, Abschluss:_____

Ich habe folgende Vorlesungen/Veranstaltungen besucht:

☐ Softwaretechnik	☐ Softwarequalität
☐ Softwareprojekt	☐ Requirements Engineering
☐ Erweitertes Softwareprojekt	☐ Labor: Agile Software-Entwicklung

Deine Muttersprache?_____

Inwieweit treffen folgende Aussagen auf dich zu. Kreuze bitte an.

	1	2	3	4	5
Ich habe viel Projekterfahrung	☐	☐	☐	☐	☐
Ich habe schon oft im Team gearbeitet	☐	☐	☐	☐	☐
Ich habe schon Software im Team entwickelt	☐	☐	☐	☐	☐
Ich habe noch nie ein Meeting analysiert	☐	☐	☐	☐	☐

1 = ich stimme überhaupt nicht zu
2 = ich stimme eher nicht zu
3 = weder Zustimmung noch Ablehnung
4 = stimme eher zu
5 = stimme völlig zu

Fragebogen 2 - Treatmentgruppe (nach Experimentende)

Inwieweit treffen folgende Aussagen auf dich zu. Kreuze bitte an.

	1	2	3	4	5
Die Unterscheidung zwischen den 9 Kategorien fiel mir leicht	☐	☐	☐	☐	☐
Oft war ich unsicher, welche Kategorie ich nehmen soll	☐	☐	☐	☐	☐
Die Anwendung des Tutorials war einfach	☐	☐	☐	☐	☐
Das Tutorial hat mir gefallen	☐	☐	☐	☐	☐
Ich habe das Gefühl, durch das Tutorial habe ich etwas gelernt	☐	☐	☐	☐	☐

1 = ich stimme überhaupt nicht zu
2 = ich stimme eher nicht zu
3 = weder Zustimmung noch Ablehnung
4 = stimme eher zu
5 = stimme völlig zu

Fragebogen 2 - Kontrollgruppe (nach Experimentende)

Inwieweit treffen folgende Aussagen auf dich zu. Kreuze bitte an.

	1	2	3	4	5
Die Unterscheidung zwischen den 9 Kategorien fiel mir leicht	☐	☐	☐	☐	☐
Oft war ich unsicher, welche Kategorie ich nehmen soll	☐	☐	☐	☐	☐

1 = ich stimme überhaupt nicht zu
2 = ich stimme eher nicht zu
3 = weder Zustimmung noch Ablehnung
4 = stimme eher zu
5 = stimme völlig zu

Erklärung der Selbstständigkeit

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

Hannover, den 09.06.2020

_____

Ann-Christin Haak