

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

**Design einer Ontologie-basierten  
Visualisierungstechnik für  
Annotationen in Vision Videos**

**Bachelorarbeit**

im Studiengang Informatik

von

**Julia Christoph**

**Prüfer: Prof. Dr. rer. nat. Kurt Schneider**

**Zweitprüfer: Dr. Oliver Karras**

**Betreuer: Lukas Nagel**

**Hannover, 16.09.2022**



# Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit 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 16.09.2022

---

Julia Christoph



# Zusammenfassung

In Softwareprojekten ist es wichtig, dass alle Stakeholder das gleiche Verständnis über die Anforderungen an das Produkt besitzen. Um Missverständnisse zu vermeiden eignen sich Videoclips sogenannte Visionvideos, welche konkrete Interaktionen des Nutzers mit dem System darstellen. Sie bieten die Vorteile, dass sie effektiver und weniger zweideutig sind im Vergleich zu textuell beschriebenen Szenarien. Damit sich die Motivation und Aufmerksamkeit der Stakeholder beim Schauen des Videos erhöht, können interaktive Elemente in Form von Annotationen eingebunden werden. Annotationen sind Anmerkungen auf dem Videobild, welche zusätzliche Informationen wie z.B. Texte beinhalten. Sie ermöglichen dem Nutzer eine aktive Erforschung der im Video gezeigten Objekte.

Da die Annotationen im Kontext von Visionvideos Informationen über die im Videobild dargestellten Objekte beinhalten, werden sie im Rahmen dieser Bachelorarbeit als Ontologien interpretiert. Ontologien sind explizit beschriebene Modelle, welche Informationen und Zusammenhänge der realen Welt abbilden. Um beliebig komplexe Strukturen von Annotationen in Visionvideos trotzdem übersichtlich darstellen zu können, werden ontologiespezifische Visualisierungstechniken genutzt. In dieser Arbeit werden die Annotationen bzw. Ontologien als kräftebasierte Graphen dargestellt, wobei die Knoten sich durch den kräftebasierten Algorithmus übersichtlich positionieren.

Für die Evaluation der ontologiebasierten Konzepte wird eine Studie durchgeführt, welche sich auf die Beurteilung der Usability konzentriert. Dafür wird untersucht wie potentielle Stakeholder mit den implementierten Konzepten umgehen und wie effektiv und effizient sie beim Suchen von Informationen sind. Zusätzlich wird die subjektiv empfundene Übersichtlichkeit der Ansicht untersucht.



# Abstract

In software projects, it is important that all stakeholders have the same understanding of the requirements of the product. In order to avoid misunderstandings, video clips so-called vision, which represent concrete interactions between the user and the system, are helpful. They offer the advantages of being more effective and less ambiguous compared to textually described scenarios. To increase the motivation and attention of the stakeholders when watching the video, interactive elements can be integrated in form of annotations. Annotations are markings on the video image that contain additional information such as texts. They allow the user to actively explore the objects shown in the video.

Since the annotations in the context of vision videos contain information about the objects represented in the video image, they are interpreted as ontologies in this bachelor's thesis. Ontologies are explicitly described models that depict information and relations of the real world. In order to be able to display any complex structure of annotations in vision videos, ontology-specific visualization techniques are used. In this work, the annotations and ontologies are represented as force-based graphs, whereby the nodes are clearly positioned by the force-based algorithm.

For the evaluation of ontology-based concepts, a study is conducted, which focuses on the assessment of usability. For this purpose, it is investigated how potential stakeholders deal with the implemented concepts and how effective and efficient they are when searching for information. In addition, the subjectively perceived clarity of the view is investigated.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Problemstellung . . . . .	1
1.3	Lösungsansatz . . . . .	2
1.4	Struktur der Arbeit . . . . .	2
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Requirements Engineering . . . . .	3
2.1.1	Requirements Analysis . . . . .	4
2.1.2	Requirements Management . . . . .	5
2.1.3	Visionvideos . . . . .	6
2.2	Ontologien . . . . .	6
2.2.1	Bestandteile einer Ontologie . . . . .	7
2.2.2	Ontologiearten . . . . .	8
2.3	Graphen . . . . .	8
2.3.1	Gerichtete Graphen . . . . .	9
2.3.2	Kräftebasierte Graphen . . . . .	9
2.4	Usability . . . . .	10
2.5	GQM - Goal Question Metric . . . . .	10
<b>3</b>	<b>Verwandte Arbeiten</b>	<b>13</b>
3.1	Interaktivität in Visionvideos . . . . .	13
3.2	Visualisierung von Ontologien . . . . .	16
3.3	Abgrenzung der eigenen Arbeit . . . . .	18
<b>4</b>	<b>Konzeption</b>	<b>21</b>
4.1	Allgemeine Designentscheidungen . . . . .	21
4.2	Traditionelles Design . . . . .	22
4.2.1	Annotationen im Videobild . . . . .	23
4.2.2	Diskussionen . . . . .	24
4.3	Ontologien mit bisherigen Konzepten . . . . .	25
4.4	Ontologiebasiertes Design . . . . .	26
4.4.1	Annotationsarten im Videobild . . . . .	26

4.4.2	Spezielle Funktionalitäten . . . . .	26
4.4.3	Darstellung als Graph . . . . .	27
4.4.4	Strukturierung der Diskussionen . . . . .	30
<b>5</b>	<b>Implementierung</b>	<b>31</b>
5.1	Programmierungsumgebung . . . . .	31
5.2	Datenformat der Ontologie . . . . .	31
5.3	Textbasierte Suche . . . . .	32
5.4	Aufgetretene Probleme . . . . .	32
5.5	Grenzen des Prototypen . . . . .	33
<b>6</b>	<b>Evaluation</b>	<b>35</b>
6.1	Verwendetes Beispiel . . . . .	35
6.2	GQM . . . . .	36
6.3	Aufbau der Studie . . . . .	37
6.3.1	Gewählte Probanden . . . . .	37
6.3.2	Testgruppen der Studie . . . . .	38
6.3.3	Ablauf der Studie . . . . .	38
6.3.4	Gestellte Aufgaben . . . . .	39
6.4	Ergebnisse . . . . .	39
6.4.1	Allgemeines Design . . . . .	40
6.4.2	Unterschiede zwischen den Designs . . . . .	40
6.4.3	Ontologiespezifische Ergebnisse . . . . .	43
6.4.4	Verbesserungsvorschläge . . . . .	44
<b>7</b>	<b>Diskussion</b>	<b>47</b>
7.1	Interpretation der Ergebnisse . . . . .	47
7.1.1	Allgemeines Design . . . . .	47
7.1.2	Effektivität und Effizienz . . . . .	47
7.1.3	Subjektiv wahrgenommene Übersichtlichkeit . . . . .	48
7.1.4	Darstellung von Zusammenhängen . . . . .	49
7.1.5	Zusammenfassung . . . . .	50
7.2	Bedrohung der Validität . . . . .	51
<b>8</b>	<b>Zusammenfassung und Ausblick</b>	<b>53</b>
8.1	Zusammenfassung . . . . .	53
8.2	Ausblick . . . . .	54
<b>A</b>	<b>Ein Anhang</b>	<b>57</b>

# Kapitel 1

## Einleitung

### 1.1 Motivation

In Softwareprojekten, ist das gemeinsame Verständnis der beteiligten Personen (auch Stakeholder genannt) essentiell für den Erfolg des Projektes [18]. Gemeinsames Verständnis bedeutet, dass es zwischen den Stakeholdern keine Missverständnisse oder unterschiedlich verstandene Anforderungen des zu entwickelnden Systems gibt. Um Kosten, welche durch falsch verstandene Anforderungen entstehen können, zu minimieren, werden im Requirements Engineering die Anforderungen an ein Produkt vor seiner Entwicklung geklärt [6]. Konkrete Szenarien sind ein hilfreiches Mittel für die Spezifikation, da sie die Sicht des Nutzers beschreiben und eine intuitive Diskussionsgrundlage bieten [17]. Sogenannte Visionvideos eignen sich besonders gut für die Darstellung von Szenarien, da sie effektiv und wenig zweideutig sind [3][4]. Durch die Möglichkeit mit Visionvideos zu interagieren, schauen sich die Stakeholder das Video mit einer höheren Konzentration und Motivation an [31]. Es gibt unterschiedliche Optionen Interaktionen in Visionvideos umzusetzen [27]. Dazu gehören Annotationen, welche Anmerkungen im Videobild sind. Sie werden dazu genutzt, wichtige Aspekte des Videos zu betonen oder um zusätzliche Informationen zu den im Bild sichtbaren Objekte zu liefern [27] [20].

### 1.2 Problemstellung

In vielen verwandten Arbeiten werden Annotationen als unabhängige und festgesetzte Punkte im Videobild dargestellt [27] [26] [23]. Verbindungen zwischen Annotationen werden in diesen verwandten Arbeiten nicht thematisiert. Dadurch werden Informationen über Zusammenhänge nicht explizit visualisiert bzw. vermittelt und gehen für den Zuschauer womöglich verloren.

Eine verwandte Arbeit löst dieses Problem, in dem eine Visualisierung von Verbindungen zwischen Annotationen implementiert wird [20].

Allerdings werden mögliche einhergehende Problematiken der Ansicht nicht genauer diskutiert oder gelöst. Beispielsweise könnte es passieren, dass relativ viele Annotationen und Verbindungen im Videobild existieren, so dass die Ansicht unübersichtlich und chaotisch wird.

Eine gute Usability der Annotationen ist jedoch essentiell für die positive Einstellung des Nutzers. Eine schlechte Übersichtlichkeit kann zu einer Frustration des Nutzers führen, so dass er die Annotationen meidet und ihr Potential im Requirements Engineering nicht ausgeschöpft wird.

### 1.3 Lösungsansatz

Ontologien werden genutzt, um beliebige Informationen und Zusammenhänge über einen Bereich abzubilden. Im Kontext von Visionvideos stellen Annotationen zusätzliche Informationen über die im Video sichtbaren Objekte bereit und können deshalb als Ontologien interpretiert werden. In dieser Bachelorarbeit werden darauf aufbauend unterschiedliche Visualisierungstechniken von Ontologien genutzt, wobei lose Annotationen eine besonders große Rolle spielen. Sie haben, im Gegensatz zu den bisher bekannten und festgesetzten Annotationen, den Vorteil, dass sie vom Programm automatisch und übersichtlich platziert werden können.

Um beurteilen zu können, welchen Nutzen die in dieser Arbeit umgesetzten Konzepte haben, wurde eine Studie mit insgesamt 16 Probanden durchgeführt. Der Fokus lag dabei auf der Untersuchung der Übersichtlichkeit, Effektivität und Effizienz zum Suchen von Informationen in Annotationen.

### 1.4 Struktur der Arbeit

Die Arbeit ist in acht Kapitel strukturiert. Kapitel 2 stellt die Grundlagen vor. Daraufhin werden in Kapitel 3 verwandte Arbeiten zu Annotationen in Visionvideos sowie der Visualisierung von Ontologien vorgestellt. In Kapitel 4 werden die implementierten Konzepte erläutert. Im folgenden Kapitel 5 werden Eigenschaften und besondere Problematiken zur Implementierung beschrieben. Das Kapitel 6 beschreibt den Aufbau und die Ergebnisse der Evaluation, so dass in Kapitel 7 eine Interpretation und Auswertung dieser beschrieben wird. Im letzten Kapitel gibt es eine Zusammenfassung der Arbeit sowie einen Ausblick für zukünftige Forschungsarbeiten.

# Kapitel 2

## Grundlagen

Dieses Kapitel beschreibt und erklärt spezifische Informationen um die Inhalte dieser Arbeit besser zu verstehen.

### 2.1 Requirements Engineering

Das Requirements Engineering ist ein wichtiger Bestandteil eines Projektes und dient dazu, Anforderungen an ein Produkt zu identifizieren, modellieren, kommunizieren und dokumentieren [28]. Es sorgt für ein umfangreiches und detailliertes Verständnis eines Produkts bevor dieses entwickelt wird. Dadurch können Kosten gespart werden, da dadurch weniger Zeit und Aufwand in die nachträgliche Korrektur von falsch verstandenen Anforderungen investiert wird. Außerdem werden mögliche problematische Anforderungen vor der Entwicklung identifiziert und aufgelöst. Problematische Anforderungen können widersprüchliche oder fehlende Anforderungen sein, welche den Erfolg des Projektes gefährden könnten. In der ISO/IEC/IEEE 24765:20 [29] sind folgende Anforderungsarten an ein Produkt zusammengefasst:

- *Funktionale Anforderungen* sind Anforderungen an die Funktionalitäten eines Produkts. Diese zeigen auf, was ein Produkt können soll.
- *Nicht-funktionale Anforderungen* sind Anforderungen an bestimmte Eigenschaften des Produkts und beschreiben wie das Produkt etwas können soll. Dazu gehören Qualitätsanforderungen wie die Schnelligkeit, Benutzerfreundlichkeit oder Sicherheit des Produkts.

Im informationstechnischen Kontext ist das zu entwickelnde Produkt meist eine Software. Deshalb wird im folgenden Kontext des Requirements Engineering stets von Software gesprochen.

In Abbildung 2.1 sind die Bestandteile vom Requirements Engineering zusammengefasst. Dabei müssen die Phasen nicht zwangsweise streng hintereinander ablaufen, sondern können ebenfalls gleichzeitig oder wiederholend

durchgeführt werden. In den nächsten Unterkapiteln werden die Einzelbestandteile genauer erläutert. Die Erklärungen beruhen auf Inhalten der Arbeiten von Herrmann [20] und Paetsch et al. [28].

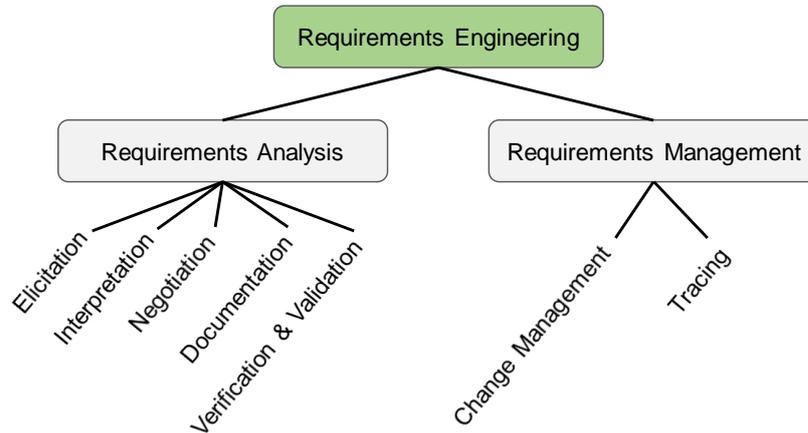


Abbildung 2.1: Requirements Engineering Referenzmodell laut Börger et al. [6].

### 2.1.1 Requirements Analysis

#### Elicitation

Diese Phase ist ganz am Anfang vom Requirements Engineering. In ihr sollen die Anforderungen an eine Software sowie die Grenzen des Systems gesammelt werden. Zuerst werden alle Personen, die mittelbar und unmittelbar mit der Software in Kontakt kommen, identifiziert. Diese Personen nennt man *Stakeholder* und zu ihnen zählen z.B. Kunden, die die Software in Auftrag gegeben haben sowie Endnutzer, welche die fertige Software benutzen werden. Um ein umfassendes Verständnis über die Anforderungen an die Software zu erhalten, ist es wichtig alle Perspektiven und Wünsche der Stakeholder zu erfassen. Dafür werden beispielsweise Interviews, Workshops oder Brainstorming-Techniken verwendet.

#### Interpretation

Bei der Interpretation werden die Anforderungen inhaltlich interpretiert um ihre genauen Bedeutungen herauszufinden. Gleiche oder ähnliche Anforderungen werden in Gruppen oder Strukturen zusammengefasst. Des Weiteren werden abstrakte Anforderungen konkretisiert. Diese Phase dient der Ordnung der Anforderungen sowie dem Identifizieren von Konflikten

zwischen diesen. Dadurch wird eine Grundlage für die nächste Phase geschaffen.

### **Negotiation**

In der dritten Phase werden Anforderungen verhandelt. Dafür werden identifizierte Widersprüche und Inkonsistenzen zwischen Anforderungen diskutiert und aufgelöst. Außerdem werden die Anforderungen priorisiert und Abhängigkeiten zwischen ihnen geklärt. Prototypen können zum Diskutieren der Anforderungen hilfreich sein und ein gleiches Grundverständnis zwischen den verhandelnden Parteien schaffen. Zum Ende dieser Phase sollte die Vision der Software feststehen.

### **Documentation**

Die Anforderungen sowie weitere Daten zu Annahmen, Kontaktpersonen, Zielen o.Ä. müssen möglichst unmissverständlich festgehalten werden, meist passiert dies in textueller Form. Die Dokumentation von Anforderungen dient der Kommunikation zwischen Stakeholdern und Entwicklern. Außerdem kann dadurch der zukünftige Fortschritt des Projekts beurteilt und mit der Zeit veränderte Anforderungen können festgestellt werden.

### **Verification and Validation**

Um zu prüfen, ob die dokumentierten Anforderungen korrekt sind, wird die fixierte Dokumentation validiert und verifiziert. Mithilfe der Validierung wird geklärt, ob die dokumentierten Anforderungen wirklich dem aktuellen Wunsch und Verständnis des Kunden entsprechen. Dafür können Gespräche und Befragungen geführt werden. Die Verifizierung prüft, ob die dokumentierten Anforderungen mit den anfänglichen Anforderungen aus der Elicitation-Phase vereinbar sind. Dafür werden formale Verfahren und Konsistenzprüfungen durchgeführt.

## **2.1.2 Requirements Management**

### **Change Management and Tracing**

Im Laufe des Projekts und der Implementierung können sich die Anforderungen verändern. Dafür ist es wichtig die Änderungen stets übersichtlich und nachvollziehbar zu verwalten. Dies beinhaltet Änderungs- und Versionskontrollen, Verfolgung der Status der Anforderungen sowie der Beziehungen zwischen den einzelnen Komponenten des Systems.

### 2.1.3 Visionvideos

*Shared understanding* (übersetzt: „Gemeinsames Verständnis“) über die Vision des zukünftigen Systems zwischen den Stakeholdern ist essentiell für den Erfolg des Projektes [18]. Es wird zwischen explizitem und implizitem *shared understanding* unterschieden. Explizites *shared understanding* ist das gemeinsame Verständnis über konkret besprochene Spezifikationen und Anforderungen. Implizites *shared understanding* ist das gemeinsame Verständnis über nicht konkret besprochenes Wissen, Annahmen und Meinungen.

Wenn die Personengruppen fälschlicherweise denken, dass sie das gleiche gemeinsame Verständnis über die Vision des Systems haben, nennt man dies *false shared understanding* [18]. Um Missverständnisse zu vermeiden, können Anforderungen mit konkreten Szenarien erweitert werden. *Scenarios* beschreiben das System aus der Perspektive des Nutzers und veranschaulichen dabei eine beispielhafte Interaktion mit dem System [17]. Sie können in Textform sowie in gedrehten Videoclips (sogenannten *Visionvideos*) dargestellt werden. Visionvideos können mehr Usabilityaspekte abdecken, einen besseren Überblick geben und sind weniger zweideutig im Vergleich zu textuell dargestellten Szenarios [3]. Sie sind in der *Elicitation*-Phase des Requirements Engineerings besonders nützlich [4].

Zusammengefasst definiert Karras [22] Visionvideos wie folgt:

*Ein Visionvideo ist ein Video, welches eine Vision oder Teile von ihr repräsentiert um ein gemeinsames Verständnis zwischen allen involvierten Parteien zu erreichen, indem deren mentale Modelle des zukünftigen Systems offenbart, diskutiert und angeglichen werden.*

## 2.2 Ontologien

Ontologien sind im Bereich der Informatik eine „explizite Spezifikation einer Konzeptualisierung“ [19]. Das bedeutet, dass die reale Welt oder Teile der realen Welt wie z.B. Gegenstände oder komplexe Zusammenhänge in einem abstrakten Modell festgehalten werden. Dieses Modell wird formal beschrieben und von einer Gruppe Menschen akzeptiert. Ontologien sind somit eine formale Beschreibung von Wissen über die reale Welt, welches von einer Gruppe Menschen geteilt wird.

Ontologien ermöglichen das Teilen und Austauschen von Informationen mit Anderen. Vor allem in der Softwareentwicklung können sie zur Spezifikation oder Identifikation von Anforderungen dienen, da sie Informationen und Zusammenhänge zwischen einzelnen Komponenten eines Systems abbilden. Dadurch können sie Wissenslücken füllen, Missverständnisse aufklären und

als Diskussionsgrundlage fungieren.

### 2.2.1 Bestandteile einer Ontologie

Eine Ontologie besteht grundsätzlich aus fünf Teilen: Klassen, Attribute, Relationen, Individuen und Axiome.

Die *Klassen* einer Ontologie sind vergleichbar mit *Interfaces* aus objektorientierten Programmiersprachen. *Klassen* können als „Sammlung“ semantisch ähnlicher Objekte verstanden werden. Sie beschreiben abstrakte Entitäten und deklarieren dazugehörige Attribute. Dabei sind *Attribute* Eigenschaften, welche zu einer Klasse gehören.

Die *Relationen* einer Ontologie beschreiben Zusammenhänge zwischen zwei Klassen.

In Abbildung 2.2 sind zwei Klassen abgebildet, welche mit der Relation „kauft“ zusammenhängen. Die Klasse „Person“ besitzt zusätzlich das Attribut „Alter“.

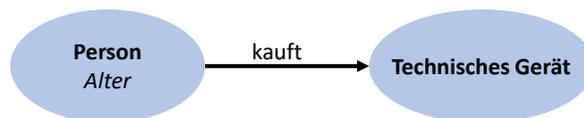


Abbildung 2.2: Ontologiebeispiel

*Individuen* sind konkrete Instanzen, welche von einer Klasse abgeleitet werden. Sie beinhalten *Attribute* mit konkreten Werten. Aufbauend auf Abbildung 2.2 werden in Abbildung 2.3 die Individuen „Käufer“ und „Smartphone“ von ihren Elternklassen „Person“ und „Technisches Gerät“ abgeleitet.

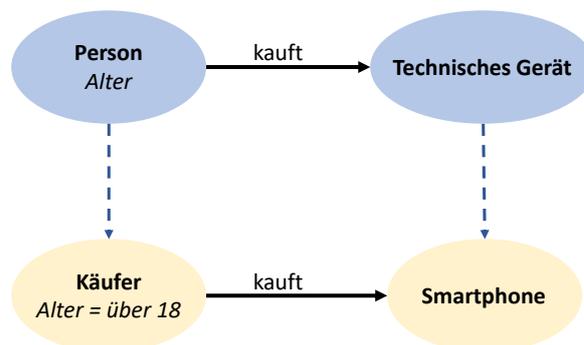


Abbildung 2.3: Ontologiebeispiel mit Individuen

*Axiome* sind die in der Ontologie dargestellten Zusammenhänge und

werden innerhalb dieser als wahr angenommen. Beispielsweise gilt in dem Beispiel aus Abbildung 2.3 das „Gesetz“, dass jeder Käufer eine Person über 18 ist und ein Smartphone kauft.

### 2.2.2 Ontologiearten

Ontologien können anhand ihrer Eigenschaften in unterschiedliche Arten eingeordnet werden [9]. Dabei werden zwei Dimensionen betrachtet, in denen Ontologien sich untereinander unterscheiden können: das Thema des Konzepts und die Reichhaltigkeit der internen Struktur. Im Rahmen dieser Arbeit spielen folgende Ontologieklassifikationen der Dimensionen eine Rolle:

#### 1. Dimension - Thema des Konzepts:

- *Wissen repräsentierende Ontologien*: Sie spezifizieren Wissen über einen definierten Bereich.
- *Anwendungsontologien*: Sie stellen Informationen über eine spezifische Anwendung dar. Oft erweitern sie *Wissen repräsentierende Ontologien*.

#### 2. Dimension - Reichhaltigkeit der Struktur:

- *Informelle Hierarchien*: Sie sind relativ flach hierarchisch aufgebaute Ontologien. Die Klassen der Ontologie sind mehr als thematische Ausrichtung zu verstehen und weniger als strikte Definition. Beispielsweise sind die Individuen „Bauarbeiter“ und „Kran“ von der Klasse „Baustelle“ abgeleitet, da sie beide als Schlüsselworte einer Baustelle zugeordnet werden können. *Informelle Hierarchien* zählen eher zu den leichtgewichtigen Ontologien (auch „lightweight ontologies“ genannt) und haben den Vorteil, dass sie eine simple Struktur besitzen und somit leicht verständlich sind. Sie beinhalten mehr Individuen als Klassen [13].

## 2.3 Graphen

Ein Graph  $G$  besteht aus einer Knotenmenge  $V$  und einer Kantenmenge  $E$  [12]:

$$V = \{v_1, v_2, \dots\}$$

$$E = \{e_1, e_2, \dots\} \text{ und } e_n \in V \times V$$

Dabei sind die Kanten eines Graphen Verbindungen zwischen zwei Knoten. Sie besitzen keine Richtung, so dass:

$$v_1, v_2 \in V, \quad e \in E \text{ und } e = (v_1, v_2) = (v_2, v_1)$$

### 2.3.1 Gerichtete Graphen

Wenn die Kanten eine Richtung besitzen, spricht man von einem gerichteten Graphen oder auch Digraph. So ist bei einer Kante  $e=(u,v)$   $u$  der Anfangsknoten und  $v$  der Endknoten.

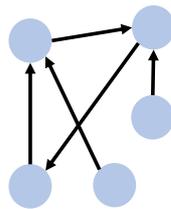


Abbildung 2.4: Beispiel eines gerichteten Graphen

### 2.3.2 Kräftebasierte Graphen

Wenn man Graphen visualisieren möchte, gibt es viele unterschiedliche Variationen wie die Knoten auf der vorhandenen Fläche positioniert werden können. Für eine ästhetisch ansprechende Visualisierung eines Graphen werden sogenannte „Spring Embedders“ bzw. (feder)kraftbasierte Algorithmen genutzt. Diese stellen Graphen besonders übersichtlich dar, da sie die Anzahl der sich überschneidenden Kanten minimieren und die Knoten symmetrisch auf der vorhanden Fläche verteilen.

Es gibt viele Varianten eines kräftebasierten Algorithmus, da sie sich in einigen Details unterscheiden. Im Kern ist das Prinzip jedoch gleich. Der Algorithmus von Fruchterman und Reingold [14] beschreibt die Knoten eines Graphen als Atomteilchen, welche sich gegenseitig abstoßen, und die Kanten eines Graphen als Anziehungskräfte. Die anfänglichen Startpositionen der Atomteilchen sind zufällig gewählt. Der Algorithmus funktioniert iterativ. In jedem Zeitschritt werden, basierend auf den einwirkenden Anziehungs- und Abstoßungskräfte zwischen den Knoten, die neuen Positionen berechnet. Das sogenannte „Simulierte Abkühlen“ sorgt dafür, dass die Bewegungen der Knoten langsamer werden, je optimierter ihre Positionen sind. Deshalb ist am Anfang der Simulation insgesamt mehr Bewegung als zum Ende. Die Abbildungen 2.5 und 2.6 vergleichen unterschiedliche Repräsentationen des gleichen Graphen.

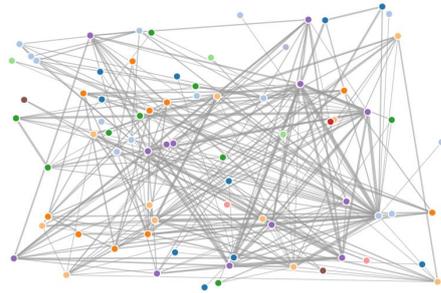


Abbildung 2.5: Graph mit zufälligen Knotenpositionen

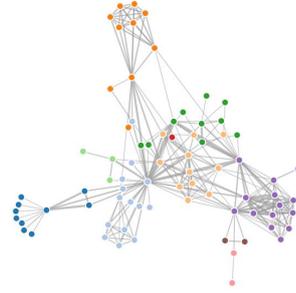


Abbildung 2.6: Kräftebasierter Graph

## 2.4 Usability

Usability ist ein wichtiger Aspekt der Qualität einer Software. Sie wird in der internationalen Norm ISO 9241-11 beschrieben als „der Grad, zu dem ein Produkt oder System durch definierte Benutzer verwendet werden kann, um spezielle Ziele in einem bestimmten Nutzungskontext zu erreichen“ [11].

Die folgenden drei Ziele sind wortwörtlich aus der ISO 9241-11 [11] entnommen:

**Effektivität:** Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen.

**Effizienz:** Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen.

**Zufriedenheit:** Freiheit von Beeinträchtigungen und positive Einstellungen gegenüber der Nutzung des Produkts.

## 2.5 GQM - Goal Question Metric

Um die Qualität oder einzelne Qualitätsaspekte wie z.B. die Usability einer Software zu bestimmen und bewerten, müssen Metriken definiert und interpretiert werden. Allerdings gibt es viele unterschiedliche Maße zum Messen von Software und nicht jedes ist zielführend für die Absichten der Messung.

Goal Question Metric (auch GQM genannt) ist eine Vorgehensweise um zielführende Metriken zu bestimmen. Das darin beschriebene Vorgehen besteht aus drei hierarchisch strukturierten Komponenten [8].

An erster Stelle stehen die Ziele und Absichten der Messung. Die Ziele können in Zielbäumen verfeinert beschrieben werden. Zu jedem dieser Ziele werden Fragen ausgearbeitet. Daraufhin werden Metriken bestimmt. Nach der Messung werden die Ergebnisse interpretiert, die Fragen dadurch beantwortet und die Ziele erreicht.

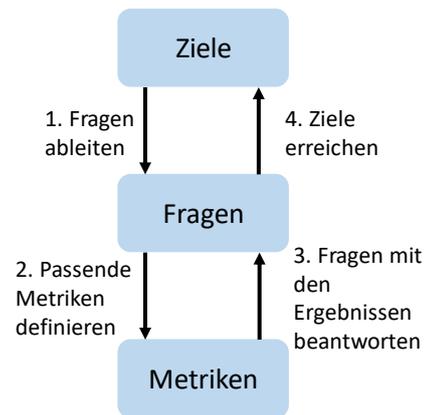


Abbildung 2.7: GQM Vorgehen



# Kapitel 3

## Verwandte Arbeiten

Dieses Kapitel stellt vorhandene verwandte Arbeiten zu interaktiven Visionvideos und der Visualisierung von Ontologien vor. Die darin vorgestellten Konzepte werden beschrieben und verglichen, bevor eine Abgrenzung dieser Bachelorarbeit erläutert wird.

### 3.1 Interaktivität in Visionvideos

Menschen tendieren dazu mehr Spaß und Aufmerksamkeit beim Lernen zu haben, wenn sie interaktiv erforschen statt lediglich passiv zu konsumieren [31]. Dieses Prinzip kann genau so auf das Schauen von Visionvideos übertragen werden. Deshalb gibt es unterschiedliche Arbeiten, die sich mit Interaktivität in Visionvideos beschäftigen und mehrere Konzepte vorstellen. In der Arbeit von Nagel und Karras [27] werden folgende Möglichkeiten zusammengefasst:

- **Verständnisfragen:** Beim Schauen des Videos werden zwischendurch Fragen gestellt, welche der Stakeholder beantwortet. Dadurch kann man prüfen, welches Verständnis der Stakeholder über das dargestellte Szenario hat und Missverständnisse vermeiden.
- **Szenengraph/alternative Pfade:** Der Stakeholder kann an bestimmten Stellen entscheiden, welche Aktion der Nutzer im Video als Nächstes ausführt und sich somit durch unterschiedliche Szenarien im Video navigieren. Dies ermöglicht ein breitgefächertes Verständnis über das Szenario.
- **Annotationen:** Beim Schauen des Videos erscheinen Annotationen. Sie sind in einem angegebenen Zeitraum und an einer festgelegten Position sichtbar und enthalten weitere Informationen über die im Bild dargestellten Objekte oder Unterstreichen dessen Wichtigkeit. Zusätzliche Informationen können in Form von Texten, Bildern, Hyperlinks, Dateien oder Audioaufnahmen repräsentiert werden [25].

Diese Bachelorarbeit beschäftigt sich hauptsächlich mit Annotationen in Visionvideos, so dass im Folgenden genauer auf sie eingegangen wird.

## Annotationen

In verwandten Arbeiten werden unterschiedliche Programme und Prototypen zum Erstellen und Darstellen von Annotationen in Videos vorgestellt. Allerdings weisen die umgesetzten Ansätze Unterschiede in einigen Details auf.

Beispielsweise werden im Programm *Siva Suite* Annotationen vorgestellt, welche das Video automatisch pausieren sobald sie angezeigt werden [26]. Dadurch wird die Wichtigkeit der Annotation hervorgehoben und ihre besonders wichtigen Informationen können nicht übersehen werden. Dahingegen werden Annotationen in der Arbeit von Nagel und Karras erst sichtbar, wenn der Stakeholder das Video manuell pausiert und die Annotationen bewusst im gestoppten Bild aktiviert [27].

Des Weiteren gibt es Unterschiede beim Anzeigen der Inhalte. In der Arbeit von Nagel und Karras [27] werden die in der Annotation enthaltenen Informationen erst angezeigt, wenn man mit einem Mausklick auf die entsprechende Annotation klickt. Daraufhin wird die Annotation ausgeklappt und die darin enthaltenen Informationen in einem kleinen Fenster angezeigt. Abbildung 3.1 zeigt ein Beispiel aus dem Paper von Nagel und Karras. Diese Vorgehensweise verhindert, dass das Videobild mit Inhalten von Annotationen überhäuft wird und somit für den Nutzer übersichtlich bleibt. Im *Siva Suite* [26] wiederum werden keine Interaktionen

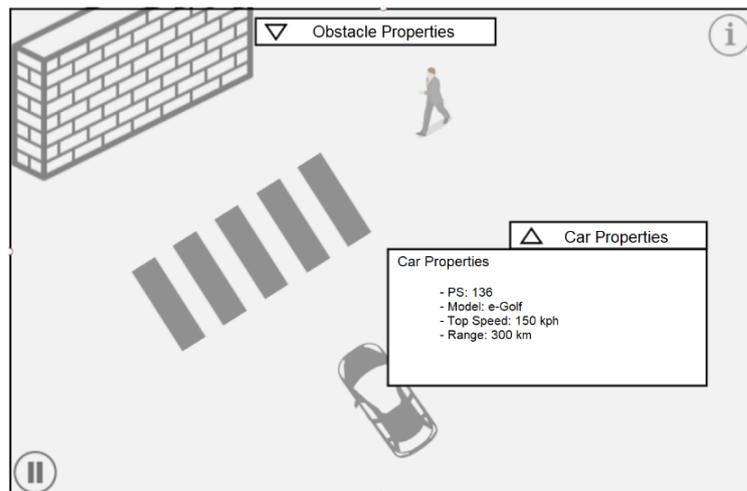


Abbildung 3.1: Die Annotation *Car Properties* im Videobild ist ausgeklappt. Die Grafik ist entnommen aus dem Paper von Nagel und Karras [27].

benötigt um die Inhalte der Annotation anzuzeigen. Sie werden direkt beim

Erscheinen der Annotation mit dargestellt.

Darüber hinaus werden im Prototypen *ReqVidA* aus der Arbeit von Karras et al. verschiedene Typen von Annotationen vorgestellt und beachtet [23]. Der Typ der Annotation muss bei der Erstellung explizit mit angegeben werden und kategorisiert sie nach ihrem Inhalt ein wie z.B. ob sie Anforderungen, Konflikte, Fragen oder Anderes repräsentiert. In der Arbeit von Nagel und Karras wiederum werden lediglich zwei Arten von Annotationen unterschieden. Die Unterscheidung bezieht sich nicht direkt auf den Inhalt der Annotation, sondern ob sie von dem Ersteller oder einem Zuschauer hinzugefügt wurde [27].

Ferner werden Annotationen in Abbildung 3.1 als viereckige Punkte abgebildet, welche weitere Informationen zu den im Bild gezeigten Objekten liefern. Demgegenüber werden Annotationen in der Masterarbeit von Herrmann [20] im Prototypen *ObViViAn* als durchscheinende Flächen dargestellt. Diese Flächen heben spezifische Teile des Bildes hervor und lenken somit zwangsläufig die Aufmerksamkeit des Stakeholders auf sich wie in Abbildung 3.2 gezeigt wird. Dieser Vergleich zeigt, dass Annotationen zu

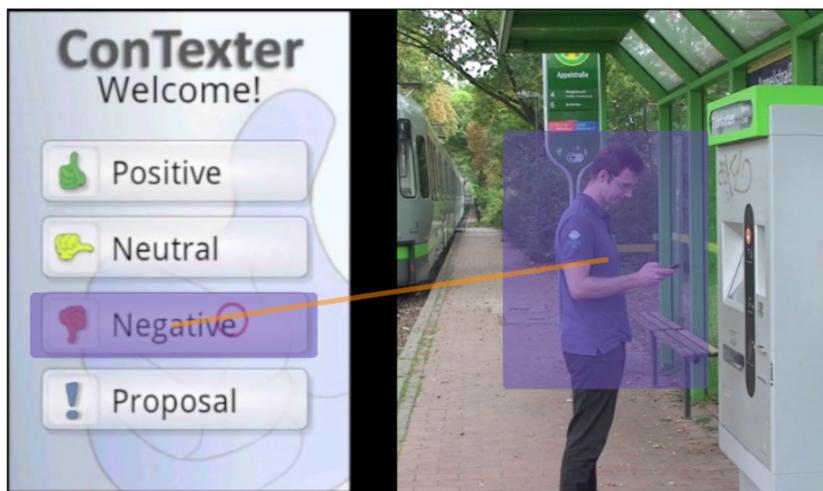


Abbildung 3.2: Annotationen im *ObViViAn* sind dargestellt als Flächen. Hier sind sie zusätzlich durch eine Linie verbunden. Die Grafik ist entnommen aus der Masterarbeit von Herrmann [20].

unterschiedlichen Zwecken genutzt werden können. Sie können als zusätzliche Informationsquelle mit textuellen Inhalten aber auch zum Hervorheben von relevanten Bildszenen eingesetzt werden.

Die beschriebenen beispielhaften Unterschiede zeigen die vielfältigen Optionen, Anforderungen oder Verständnisse eines Annotationsprogramms auf. Trotz der feinen Unterschiede haben die vorgestellten Arbeiten grundsätzliche Vorgehensweisen gemein. Sie beschäftigen sich alle mit einer

prototypischen Umsetzung zum Erstellen, Editieren und Anzeigen von Annotationen in Visionvideos. Jede der Arbeiten beschreibt, dass der Ersteller der Annotationen ihre Positionen und Start- und Endzeitpunkte der Sichtbarkeit angeben muss. Außerdem werden die Annotationen stets als vereinzelte Anmerkungen dargestellt, welche unabhängig voneinander im Bild zu einem gegebenen Zeitpunkt an einer festgelegten Position erscheinen oder zu sehen sind.

Lediglich die Masterarbeit von Herrmann[20] ermöglicht zusätzlich eine Repräsentation von Zusammenhängen zwischen Annotationen durch Linien wie in Abbildung 3.2 gezeigt wird. Dafür werden die Annotationen zu einer Klasse einer Ontologie zugeordnet. Sind zwei Klassen durch eine Relation miteinander verbunden, so wird eine sichtbare Linie zwischen den zwei Annotationen erstellt. Die Ontologie im Prototypen *ObViViAn* ist fest vordefiniert und beinhaltet die Klassen „Software“, „Hardware“, „Mensch“ und „Anderes“ mit den möglichen Relationen „liefert Informationen an“ und „erhält Informationen von“ [20].

Im Softwareentwicklungsprozess dienen Annotationen in Visionvideos vor allem zum Spezifizieren, Diskutieren und Dokumentieren von Anforderungen [23]. Der Ansatz Annotationen in Visionvideos mit Ontologien zu kombinieren, ermöglicht es Strukturen und Zusammenhänge verständlich und übersichtlich darzustellen. Aufgrund dessen erforscht diese Bachelorarbeit dieses Themengebiet detaillierter.

### 3.2 Visualisierung von Ontologien

Die Visualisierungen von Ontologien haben den Nutzen, dass sie die in der Ontologie beschriebenen Strukturen und Konzepte verdeutlichen [10]. Dabei ist einer der größten Vorteile, dass Ontologien das große Ganze zeigen während sie gleichzeitig Detailinformationen bereithalten [10].

Ein Survey von Dudás et al. [10] listet Funktionen auf, die dem Nutzer helfen können die Ontologie besser zu verstehen und zu erforschen. Im Rahmen dieser Bachelorarbeit spielen Folgende eine Rolle:

- **Pop-Up Fenster:** Informationen zu einer gewünschten Entität in einem Extrafenster anzeigen
- **Suchen und Hervorheben:** Textbasierte Suche innerhalb der Ontologie und Markierung der Entitäten, in denen Treffer gefunden wurden.
- **Filterung:** Nichtanzeigen von Teilen, die den Nutzer nicht interessieren, damit die Ansicht übersichtlich bleibt.
- **Drag&Drop:** Möglichkeit, dass der Nutzer die Entitäten bewegen kann und somit die Ansicht zu seinem eigenen Belieben anpassen und individualisieren kann.

Viele Visualisierungsprogramme für Ontologien stellen diese als kräftebasierte Graphen oder Hierarchien dar. Dabei fokussieren sich die meisten auf eine stufenweise Exploration und zeigen anfänglich nur eine ausgewählte Klasse, von dessen Ausgangspunkt man andere Klassen und Entitäten erforschen kann [24].

VOWL-2 ist ein Visualisierungsprogramm für Ontologien [24], welches demgegenüber auf Shneidermanns Grundgedanken „Überblick zuerst, dann Details auf Nachfrage“ [30] basiert. Es ermöglicht, dass der Nutzer von Beginn an den vollen Umfang und die Struktur der Ontologie erfährt. Gleichzeitig ist die Darstellung für jeden intuitiv verständlich und nicht nur für Ontologieexperten nutzbar [24]. VOWL-2 repräsentiert Ontologien als kräftebasierte Graphen, wobei die Klassen und Attribute die Knoten des Graphen sind und die Relationen durch Kanten des Graphen dargestellt werden.

### Ontologien als kräftebasierte Graphen

Da die Visualisierung einer Ontologie als kräftebasierter Graph intuitiv verständlich ist, beschäftigen sich mehrere verwandte Arbeiten mit den Spezifitäten dieser Darstellungsform.

Die Startpositionen der Knoten eines kräftebasierten Graphen beeinflussen das Layout des Endergebnisses. Wenn zufällige Startpositionen gewählt werden, kann der gleiche Graph am Ende trotzdem unterschiedlich aussehen. Abbildung 3.3 zeigt den gleichen Graphen und veranschaulicht den Einfluss der Startpositionen auf die Anzahl an Iterationsschritten und die Positionen des Endergebnisses.

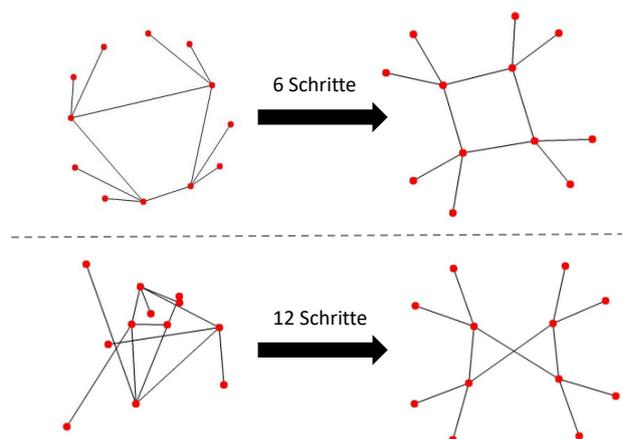


Abbildung 3.3: Der gleiche Graph mit unterschiedlichen Startpositionen und somit Endergebnissen. Das Beispiel stammt aus dem Paper von Toosi et al. [16]

Eine minimale Anzahl von sich kreuzenden Kanten wird als ästhetisch ansprechend und übersichtlich empfunden. Um das beste Endergebnis zu erhalten und die Anzahl an nötigen Iterationsschritten bis zum Endergebnis zu minimieren, beschreibt das Paper von Toosi et al. [16] einen Lösungsansatz. Darin wird beschrieben, dass die Startpositionen der Knoten nicht zufällig gewählt werden sollten. Der dafür beschriebene Algorithmus zielt darauf ab, den anfänglichen Abstand zwischen verbundenen Knoten zu minimieren und berechnet ihre Startpositionen. Gruppierte Knoten tendieren dazu näher aneinander zu starten, wodurch weniger sich kreuzende Kanten entstehen und insgesamt weniger Iterationsschritte bis zum optimalen Endergebnis nötig sind. Darauf aufbauend betont die Arbeit von Fluit et al. die Wichtigkeit der Platzierung der Knoten eines Graphen, da die geometrische Nähe zwischen Knoten auf eine semantische Ähnlichkeit der Klassen der Ontologie hindeutet. Aufgrund dessen wird vorgeschlagen, dass Entitäten, welche gemeinsame Klassen haben, gruppiert oder näher aneinander dargestellt werden sollten [13].

### 3.3 Abgrenzung der eigenen Arbeit

Annotationen können als Informationsquelle für die im Bild sichtbaren Objekte genutzt werden. Die Darstellung von Verbindungen ermöglicht es, weitere Informationen über Strukturen und Zusammenhänge zwischen ihnen visuell darzustellen und zu vermitteln. Deshalb wird in der Masterarbeit von Herrmann [20] die Darstellung von Verbindungen zwischen Annotationen ermöglicht, in dem die Annotationen mit einer vordefinierten Ontologie kombiniert werden.

In dieser Bachelorarbeit wird der Ansatz von Herrmann verwendet und erweitert, in dem eine übersichtliche Visualisierung für beliebig große und komplexe Ontologien konzeptioniert wird. Dafür werden im Kontext von Visionvideos erstmals ontologiespezifische Visualisierungstechniken vorgestellt.

Allerdings behandeln alle verwandten Arbeiten, einschließlich die Masterarbeit von Herrmann, lediglich im Bild fest positionierte Annotationen. Bei der Kombination mit Ontologien können bisher noch nicht diskutierte Problematiken im Bezug auf die Usability auftreten. Diese Risiken werden in dieser Bachelorarbeit aufgedeckt und durch eigene Konzepte aufgelöst. Dafür werden unter Anderem sogenannte lose Annotationen eingeführt, welche sich mithilfe eines kräftegesteuerten Algorithmus automatisch übersichtlich positionieren.

Außerdem betrachtet diese Bachelorarbeit die Option, dass die vom Nutzer hinzugefügten Annotationen meist Rückfragen, Kommentare oder Anregungen beinhalten. Deshalb werden diese Diskussionen von den Informationen der Annotationen getrennt dargestellt. Keine der verwandten Arbeiten geht darauf ein, dass vom Nutzer erstellte Annotationen grafisch

in einem anderen Bereich dargestellt oder in einem Forum separiert werden könnten.

Des Weiteren wurden bisher keine Studien durchgeführt, um die Darstellung von Zusammenhängen zwischen Annotationen im Kontext von Visionvideos zu evaluieren. Im Rahmen dieser Bachelorarbeit wird die Usability der umgesetzten ontologiespezifischen Konzepte untersucht und bewertet, in dem genauer auf die Effektivität, Effizienz und Übersichtlichkeit eingegangen wird.



# Kapitel 4

## Konzeption

In diesem Kapitel werden die im Prototypen implementierten Konzepte vorgestellt. Zuerst wird auf das allgemeine Design eingegangen und in den weiteren Unterkapiteln auf das interpretierte Design aus verwandten Arbeiten sowie das entwickelte ontologiespezifische Design.

### 4.1 Allgemeine Designentscheidungen

Die allgemeinen Designentscheidungen, welche in dieser Bachelorarbeit für die grafische Oberfläche getroffen wurden, basieren auf den Usability Guidelines aus den Arbeiten von Cabrera [7], Johnson [21] und Galitz [15].

Die Ansicht wird in drei Bereiche aufgeteilt: die Menüleiste, das Videobild mit den Annotationen und die separate Diskussionsleiste. Abbildung 4.1 zeigt die Aufteilung der grafischen Oberfläche. Es wird angenommen, dass das Videobild stets das Format 3:4 besitzt. Deshalb ist es ursprünglich 100% hoch und 75% breit. Allerdings nimmt das Menü bereits 5% der Höhe des Bildschirms ein, so dass das Videobild lediglich 95% der vorhandenen Höhe einnimmt. Damit das Videobild im Format 3:4 bestehen bleibt, ist es  $75\% * 0,95 = 71,25\%$  breit. Die Diskussionsleiste nimmt die restliche Breite von 28,75% des Bildschirms ein.

Die Diskussionsleiste hat zusätzlich zu der vertikalen Trennlinie einen hellblauen Hintergrund, so dass die Grenzen dieses Bereichs besser ersichtlicher werden. Außerdem wird das Videobild um 40% heller dargestellt, damit es für den Nutzer in den Hintergrund rückt und die Annotationen durch den weißen Kontrast farblich besser herausstechen und nicht übersehen werden können.

Damit die grafische Oberfläche intuitiv bedienbar ist, müssen alle Interaktionsmöglichkeiten für den Nutzer leicht erkennbar sein. Wenn das Programm auf die Nutzereingabe reagiert und ein visuelles Feedback gibt, wird eine Interaktionsmöglichkeit symbolisiert. Geht der Nutzer mit dem Mauszeiger

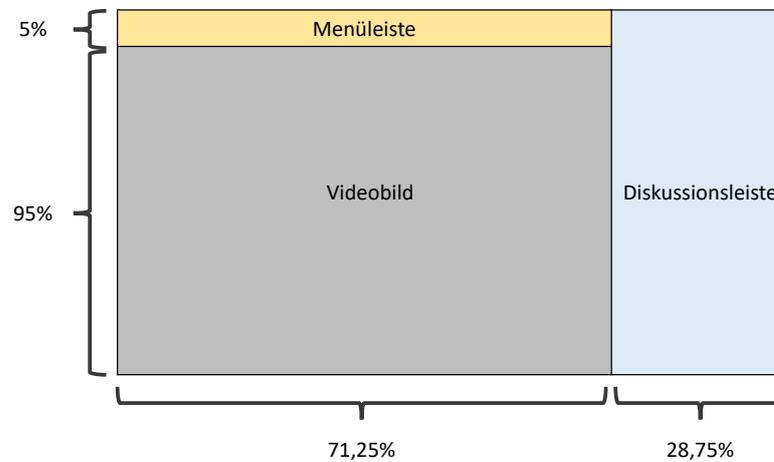


Abbildung 4.1: Aufteilung des Bildschirms mit ausgeklappter Diskussionsleiste

auf ein interaktives Element, färbt es sich deshalb dunkler und die Form des Mauszeigers verändert sich. Abbildung 4.2 und Abbildung 4.3 zeigen dazu ein Beispiel. Allgemein sind alle interaktiven Elemente konsistent in einem auffälligen aber trotzdem neutralen orange, so dass der Nutzer „lernt“, dass alle interaktiven Elemente die gleiche Farbe besitzen und sie auf einen Blick erkennen kann.



Abbildung 4.2: Button zum Öffnen der Diskussionsleiste



Abbildung 4.3: Button zum Öffnen der Diskussionsleiste symbolisiert Klickbarkeit.

## 4.2 Traditionelles Design

Im Abschnitt 3.1 werden verschiedene in verwandten Arbeiten entwickelte Konzepte zu Annotationen in Visionvideos und deren Unterschiede vorgestellt. Dieses Kapitel beschreibt, auf welchen der Konzepte diese Bachelorarbeit aufbaut. Der Einfachheit halber werden die Ansätze aus verwandten Arbeiten fortan als „traditionelles Design“ betitelt.

In der Arbeit von Nagel und Karras [27] wird zwischen zwei Arten von Annotationen unterschieden: jene, die vom Ersteller erstellt und jene,

die vom Nutzer hinzugefügt wurden. In dem im Rahmen des Papers vorgestellten Prototypen wird angenommen, dass die Annotationen, die vom Ersteller hinzugefügt wurden, zusätzliche Informationen beinhalten. Die Annotationen, die vom Zuschauer hinzugefügt wurden, bestehen meist aus Kommentaren und Fragen. Aufgrund dessen werden die vom Zuschauer hinzugefügten Annotationen in dieser Bachelorarbeit als Diskussionen interpretiert und separat in einer seitlichen Leiste dargestellt. Wenn in den folgenden Abschnitten Annotationen erwähnt werden, sind immer die Annotationen im Videobild gemeint, welche vom Ersteller hinzugefügt wurden und Informationen beinhalten.

### 4.2.1 Annotationen im Videobild

Diese Bachelorarbeit setzt bei den beschriebenen Ansätzen aus der Arbeit von Nagel und Karras [27] an. Die im Paper beschriebenen Konzepte wurden für die Konzeption des Prototypen dieser Bachelorarbeit übernommen. Demnach ist der Zweck der Annotationen zusätzliche Informationen zu den im Bild dargestellten Objekten zu bieten. Die Informationen werden für die Übersichtlichkeit nicht in reiner Textform dargestellt, sondern als Auflistung von sogenannten *Key-Value-Paaren*. Ein *Key-Value-Paar* ist eine Verknüpfung zweier Elemente, wobei der Key der Bezeichner der Information ist und der Value der dazugehörige konkrete Wert. Des Weiteren werden die Informationen der Annotationen erst bei einem Mausklick auf diese ausgeklappt und sichtbar. Abbildung 3.1 zeigt dazu ein Beispiel aus dem Paper von Nagel und Karras [27]. Abbildung 4.4 zeigt das gleiche Szenario. Allerdings stammt es aus dem traditionellen Design, welches im Rahmen dieser Bachelorarbeit entwickelt wurde. Trotz unterschiedlicher Designentscheidungen wie z.B. die Farben der Annotationen sind die Konzepte in beiden Abbildungen gleich.

Allerdings gibt es Details zur Konzeption, welche von Nagel und Karras [27] nicht genauer beschrieben werden. Deshalb werden die im Paper vorgestellten Konzepte durch Eigene erweitert:

Die Größe der Annotationen selbst und die Größe des ausgeklappten Fensters werden in dem Paper von Nagel und Karras [27] nicht genauer diskutiert. Allerdings können bereits durch das Aussehen einer Ansicht Informationen vermittelt werden. Ist eine Annotation größer als die anderen, würde sie vom Nutzer als wichtiger empfunden werden und schneller ins Auge stechen [7]. Deshalb sind alle Annotationen in dieser Bachelorarbeit gleich groß. Für den Fall, dass der Annotationsname zu lang ist und nicht mehr vollständig angezeigt werden kann, werden die vollständigen Bezeichnungen sichtbar, wenn die Maus auf der Annotation liegt. Ein Beispiel dafür ist die Annotation „Car Properties“ in Abbildung 4.4. Die Höhe und Breite des Informationsfensters ist konstant festgelegt. Wenn zu viele Informationen für diese Fläche vorhanden sind, kann darin mit der Maus „gescrollt“ werden.

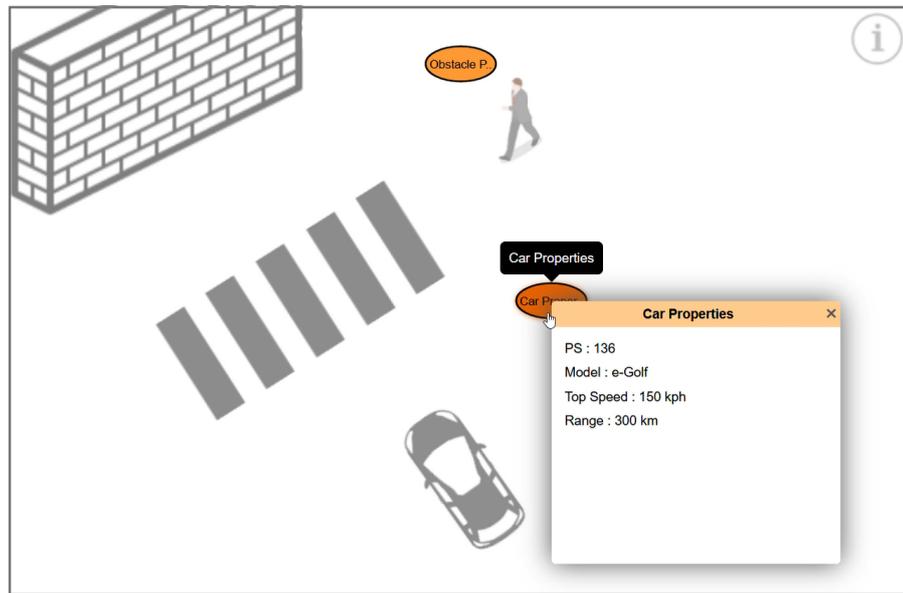


Abbildung 4.4: Videobild im traditionellen Design. Das Szenario ist äquivalent mit dem von Abbildung 3.1.

In dieser Bachelorarbeit werden jederzeit nur Informationen zu einer Annotation angezeigt und niemals mehrere Annotationen gleichzeitig ausgeklappt. Dies hat den Vorteil, dass das Videobild nicht durch ausgeklappte Informationsfenster überhäuft werden kann und gleichzeitig der Fokus jederzeit auf der für den Nutzer interessanten Annotation liegt.

### 4.2.2 Diskussionen

In der Arbeit von Amiri [1] werden unterschiedliche Darstellungsformen für asynchrone Kommunikation vorgestellt und verglichen. Ein Design wie bei *YouTube* schnitt bei den Ergebnissen einer Studie positiv ab, da es intuitiv schnell verständlich ist und die Diskussionen verschachtelt strukturiert sind. Aufgrund dessen werden die Diskussionen in dieser Bachelorarbeit ähnlich dargestellt. Abbildung 4.6 zeigt ein Beispiel der ähnlichen Visualisierung von Diskussionen. Genau wie bei *YouTube* werden alle Kommentare und Fragen an einer Stelle gesammelt angezeigt.



Abbildung 4.5: Zensiertes Beispiel einer ausgeklappten Diskussion in YouTube

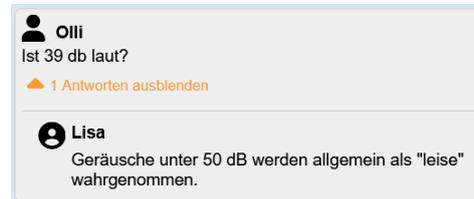


Abbildung 4.6: Beispiel einer ausgeklappten Diskussion im Prototypen

### 4.3 Ontologien mit bisherigen Konzepten

Im traditionellen Design sind Annotationen voneinander unabhängige Anmerkungen. Im Gegensatz dazu bieten Ontologien die Möglichkeit Annotationen mit Zusammenhängen und beliebigen Strukturen abzubilden. Da die Annotationen äquivalent zu den Individuen der Ontologie sind, werden diese Begriffe fortan gleichgesetzt.

Im Abschnitt 3.1 werden verwandte Arbeiten vorgestellt, welche sich mit Annotationen in Visionvideos beschäftigen. Einige der vorgestellten Konzepte sind in allen verwandten Arbeiten gleich. Die Annotationen werden jederzeit an einer festen Position dargestellt, da sie sich auf die im Bild sichtbaren Objekte beziehen. Dafür muss der Ersteller der Annotationen jede Annotation, die er erstellt, manuell positionieren und einen Zeitraum angeben, um die Dauer und den Zeitpunkt der Sichtbarkeit der Annotation zu definieren. Zusammenhänge zwischen diesen Annotationen können mittels einer Ontologie umgesetzt werden, in dem eine Relation erstellt und eine Linie visualisiert wird.

Die Kombination von den vorgestellten festgesetzten Annotationen mit Ontologien birgt einige Risiken:

- **Falscher oder vergessener Zeitraum:** Es ist möglich, dass der Ersteller eine Annotation erstellt, welche mit einer anderen verbunden ist, aber nicht die richtigen Zeiträume der Sichtbarkeit übernimmt. Die Annotationen sind zwar miteinander verbunden, aber dadurch, dass sie nie gleichzeitig sichtbar sind, wird kein Zusammenhang dargestellt oder erkenntlich.
- **Unübersichtlichkeit:** Wenn der Ersteller jede Annotation manuell positionieren muss, kann es passieren, dass die Ansicht unübersichtlich wird. Je mehr Annotationen und Zusammenhänge entstehen, desto besser müssen die Positionen der Annotationen gewählt werden.

- **Hoher Aufwand:** Je mehr Annotationen erstellt werden, desto höher ist der Aufwand des Erstellers. Wenn jede Annotation zusätzlich übersichtlich positioniert und ihre Zeiträume passend zu den verbundenen Annotationen definiert werden müssen, kann das Erstellen sehr aufwändig werden und viel Zeit und Konzentration benötigen.

Zusammengefasst muss der Ersteller einen guten Überblick über die erstellte Ontologie haben, damit keine unbeabsichtigten Effekte bei der Darstellung der Annotationen entstehen. Als Lösung für die Problematik werden in dieser Bachelorarbeit Annotationen ohne feste Position und ohne Zeitraum eingeführt.

## 4.4 Ontologiebasiertes Design

In diesem Abschnitt werden die umgesetzten ontologiespezifischen Konzepte erläutert.

### 4.4.1 Annotationsarten im Videobild

Im traditionellen Design besitzt jede Annotation eine feste Position und einen festen Zeitraum, in dem sie im Video sichtbar ist. Das ontologiebasierte Design erweitert diesen Ansatz und ermöglicht zusätzlich die Darstellung von losen Annotationen ohne Sichtbarkeitszeitraum. In der folgenden Tabelle 4.1 werden die Eigenschaften dieser beiden Annotationsarten gegenübergestellt.

Aspekt	Festgesetzte Annotationen	Lose Annotationen
Position	Fest angegeben	Iterativ neu berechnet durch einen kräftebasierten Algorithmus
Sichtbarkeit	Explizit angegeben durch einen Zeitraum	Rekursiv berechnet anhand der Verbindungen
Repräsentation im Prototypen	Dicke Umrandung	Ohne Umrandung und leicht durchsichtige Farbe

Tabelle 4.1: Annotationsarten und Eigenschaften

### 4.4.2 Spezielle Funktionalitäten

Im Abschnitt 3.2 werden Funktionen aufgelistet, welche die Übersichtlichkeit und Erforschung von Ontologien verbessern können. In dieser Bachelorarbeit sind folgende Funktionalitäten umgesetzt:

- **Filter:** Der Filter ermöglicht es, die Individuen nach ihren Klassen zu filtern. Das bedeutet wenn im Filter eine Klasse ausgewählt wird, werden alle Annotationen, welche diese Klasse nicht besitzen, unsichtbar. Wenn mehrere Kategorien im Filter aktiviert werden, werden diese mit einem logischen UND verknüpft.
- **Suchleiste:** In der Suchleiste kann eine textbasierte Suche durchgeführt werden. Es wird nicht zwischen Groß- und Kleinschreibung unterschieden und Leerzeichen vor und hinter dem eingegebenem Wort werden ignoriert. Dadurch ist die Suche weniger fehleranfällig. Die Suche sucht in den Inhalten der Annotationen und in den Inhalten der Diskussionen nach Treffern. Wenn ein Treffer gefunden wurde, wird die Annotation selbst und die Textstelle gelb markiert. Des Weiteren wird oben in der Diskussionleiste und im Informationsfenster angezeigt, wie viele Treffer gefunden wurden.
- **Drag&Drop:** Um die Ansicht zu verschönern, ist es möglich die losen Annotationen mit dem Mauszeiger zu verschieben. Die neue Position wird nicht dauerhaft gespeichert, so dass die Annotation beim nächsten Mal, wenn das Video abgespielt wird, sich wieder an der alten Stelle befindet. Außerdem kann man die losen Annotationen nie festsetzen. Das bedeutet, wenn eine lose Annotation verschoben wird, reagiert sie beim Loslassen auf die auf sie einwirkenden Kräfte und positioniert sich um.

#### 4.4.3 Darstellung als Graph

Die Annotationen im Videobild bzw. die Individuen einer Ontologie werden als gerichtete kräftebasierte Graphen abgebildet.

Die folgende Tabelle zeigt alle Komponenten einer Ontologie und bildet diese auf die Komponenten eines Graphen sowie die umgesetzte Visualisierung ab.

Ontologie-komponenten	Graph-komponenten	Darstellung
Klassen	-	Filterkategorien
Individuen	Knoten	Ellipsen
Relationen	Gerichtete Kanten	Pfeile (unbeschriftet)
Attribute	-	Informationen im Pop-Up Fenster

Tabelle 4.2: Mappingtabelle

## Berechnung der Knotenmenge des Graphen

Ob eine lose Annotation zu einem bestimmten Zeitpunkt des Videos angezeigt werden soll oder nicht, wird mit einem rekursiven Algorithmus berechnet. Der Grundgedanke ist dabei, dass zu einem bestimmten Zeitpunkt alle festgesetzten Annotationen angezeigt werden. Im ontologiebasierten Design werden daraufhin zusätzlich alle Knoten ermittelt, welche direkt oder indirekt mit diesen festgesetzten Annotationen verbunden sind. Dabei spielt die Richtung der Verbindung eine entscheidende Rolle. Ausgehend von den festen Annotationen, werden lediglich die ausgehenden Kanten und die damit verbundenen Kindknoten betrachtet.

Ein wichtiger Aspekt ist, dass der rekursive Algorithmus nicht berechnet, welche Knoten des Graphen sichtbar sind, sondern welche Knoten überhaupt in dem Graphen enthalten sind. Das bedeutet, dass alle Annotationen, welche zu einem bestimmten Zeitpunkt nicht festgesetzt oder mit dem rekursiven Algorithmus erfasst wurden, nicht im Graphen enthalten sind und keinen Einfluss auf den Layout-Algorithmus und somit die Positionierung der sichtbaren Annotationen haben.

Zusammengefasst gilt, dass die losen Kinder aufgrund von ihrer Verbindung zu einem festen Knoten zu dem kräftebasierten Graphen gehören und dadurch sichtbar werden. Obwohl der Ersteller nicht explizit angegeben hat, dass diese Annotationen zu dem Zeitpunkt sichtbar sein sollen, werden sie trotzdem visualisiert, weil sie vom rekursiven Algorithmus erfasst werden.

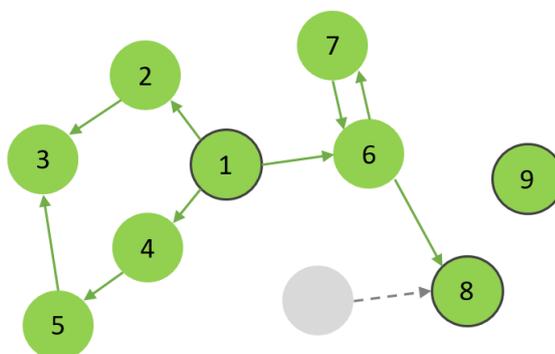


Abbildung 4.7: Rekursive Suche um alle sichtbaren Knoten zu berechnen. Knoten mit dickem Rand stellen festgesetzte Annotationen dar.

Zum Beispiel wird ein Visionvideo bei Sekunde 50 gestoppt und die Annotationen aktiviert. Der rekursive Algorithmus prüft ausnahmslos alle Individuen der Ontologie, indem er untersucht, ob das betrachtete Individuum zu Sekunde 50 festgesetzt ist. Wird eine festgesetzte Annotation gefunden, so werden rekursiv all ihre Kindknoten gesucht. In Abbildung 4.7 ist ein Beispiel, in dem zuerst der festgesetzte Knoten 1 gefunden wurde.

Daraufhin wurde sein Kindknoten 2 sichtbar. Dieser Knoten wiederum wird ebenfalls rekursiv auf seine Kindknoten untersucht, so dass Knoten 3 gefunden wird. Knoten 3 besitzt keine und Knoten 2 keine weiteren Kindknoten. Deshalb werden weitere Kindknoten des Knoten 1 überprüft. Daraufhin wird der Knoten 4 gefunden. Das beschriebene Vorgehen wiederholt sich bis alle Kindknoten und Kanten zur Sekunde 50 gefunden wurden. In der Ontologie aus der Abbildung 4.7 wird ein Knoten niemals sichtbar, da er kein sichtbares Elternteil besitzt und somit vom rekursiven Algorithmus nie erreicht wird. Dieser Knoten ist in der Abbildung in grauer Farbe markiert. Im Anhang befinden sich die Pseudocodes des rekursiven Algorithmus.

### Berechnung der Startpositionen der Knoten

Allgemein werden die festgesetzten Annotationen durch den kräftebasierten Algorithmus nicht beeinflusst, da ihre Position zum Videobild passen sollen und vom Ersteller festgelegt werden. Somit wirkt der kräftebasierte Algorithmus lediglich auf die losen Annotationen. Vor dem Start werden die Anfangspositionen der losen Knoten berechnet. Dafür werden für jeden losen Knoten rekursiv alle festgesetzten Elternknoten ermittelt. Dann wird der Mittelwert dieser Positionen berechnet, so dass jeder lose Knoten mittig von allen mit ihm verbundenen festen Elternknoten startet. Abbildung 4.8 zeigt eine Ontologie, in der lediglich der Knoten b lose ist. Um die Startposition des losen Knoten zu berechnen, werden rekursiv die festgesetzten Elternknoten ermittelt. In dem Beispiel sind es Knoten a und Knoten c. Abbildung 4.9 zeigt somit, die Startpositionen aller Knoten, wobei b genau zwischen seinen Elternknoten a und c startet. In Abschnitt 3.2 werden die Vorteile dieser Berechnung im Vergleich zu zufälligen Startpositionen genauer beschrieben.

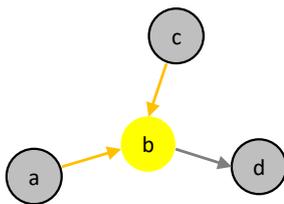


Abbildung 4.8: Endpositionen. Der Knoten b hat die Elternknoten a und c.

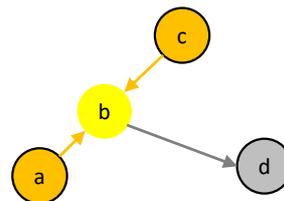


Abbildung 4.9: Startpositionen. b startet zwischen seinen Elternknoten a und c.

### Berechnung der Kantenlängen zwischen Knoten

Im Nutzungskontext von Visionvideos wird angenommen, dass die Ontologien die Struktur einer *informellen Hierarchie* haben. Das bedeutet, dass es insgesamt mehr Individuen als Klassen gibt, wodurch mehrere Individuen

mindestens eine gleiche Klasse haben. Im Rahmen dieser Arbeit wird angenommen, dass die Individuen maximal zu zwei Klassen zugeordnet sind und somit können sie maximal zwei gleiche Klassen mit anderen Individuen besitzen. Allerdings beruht diese Annahme nicht auf weiteren Quellen oder Untersuchungen.

Individuen, welche gemeinsame Klassen besitzen, werden im Graphen näher aneinander positioniert, da eine geometrische Nähe zwischen Knoten eine semantische Ähnlichkeit symbolisiert. Aufgrund dessen variiert die Länge der Kanten zwischen den Individuen je nach Anzahl gemeinsamer Klassen. Dabei gibt es einen größeren Unterschied zwischen Individuen, welche keine oder mindestens eine Klassenüberschneidung haben. Die Distanz zwischen den Knoten verändert sich deshalb wie folgt:

Anzahl gemeinsamer Klassen zwischen zwei Knoten	Länge der Kante
0	6 * radiusY
1	4 * radiusY
2	3 * radiusY
> 2	3 * radiusY

Tabelle 4.3: Kantenlängen

#### 4.4.4 Strukturierung der Diskussionen

Im traditionellen Design sind alle Diskussionen im Verlauf des kompletten Videos in der rechten Seitenleiste des Programms aufgelistet. Im ontologiebasierten Design jedoch sind die Diskussionen immer einem Annotationspunkt zugeordnet. Wenn man auf eine Annotation klickt, öffnet sich das Pop-Up Fenster mit den Informationen und die zu der Annotation dazugehörigen Diskussionen werden in der seitlichen Leiste angezeigt. Ist keine Annotation geöffnet, so ist die Diskussionsleiste leer.

Die Aufteilung der Diskussionen auf Annotationen hat den Vorteil, dass weniger Diskussionen angezeigt werden und sie dadurch insgesamt übersichtlicher sind. Gleichzeitig sind sie automatisch dem Oberthema der Annotation zugeordnet. Außerdem könnten sie als wichtiger empfunden werden, da sie interaktiver auf die Nutzereingaben reagieren als eine statische sich nicht verändernde Auflistung.

# Kapitel 5

## Implementierung

In diesem Kapitel wird die Programmierumgebung und das Format der Ontologie genauer beschrieben.

### 5.1 Programmierumgebung

Das Modul namens D3 (Data Driven Documents) wurde im Visualisierungsprogramm VOWL verwendet und bietet eine hohe Performanz sowie weitere Module, welche das Visualisieren der kräftebasierten Elemente vereinfacht [2]. Aufgrund dessen wurde dieses Modul ebenfalls in dem Prototypen dieser Bachelorarbeit eingesetzt, so dass die Implementierung in HTML, CSS und Javascript stattgefunden hat.

### 5.2 Datenformat der Ontologie

Die Ontologie wird in einem verschachtelten Objekt gespeichert, welches vom Prototypen ausgelesen und verwendet wird. Sie wird in dem Datenformat von *JavaScript Object Notation* (abgekürzt *JSON*) gespeichert. Die Ontologie beinhaltet alle Annotationen im Laufe des kompletten Visionvideos und besteht aus drei Oberkomponenten: eine Auflistung aller Klassen, aller Individuen und aller Relationen. Die Individuen besitzen die meisten Bestandteile, da sie eine ID, eine Beschriftung, alle Sichtbarkeitszeiträume und Positionen, Attribute, eine Auflistung der zugehörigen Klassen und alle zu ihr zugehörigen Diskussionsstränge enthalten. Die Relationen hingegen beinhalten lediglich die ID des Start- und Endknoten. Die Abbildung 5.2 zeigt den Aufbau des JSON-Objektes, welches die Ontologie repräsentiert. Die genaue Syntax des JSON-Objektes wird in einer Tabelle im Anhang erklärt.

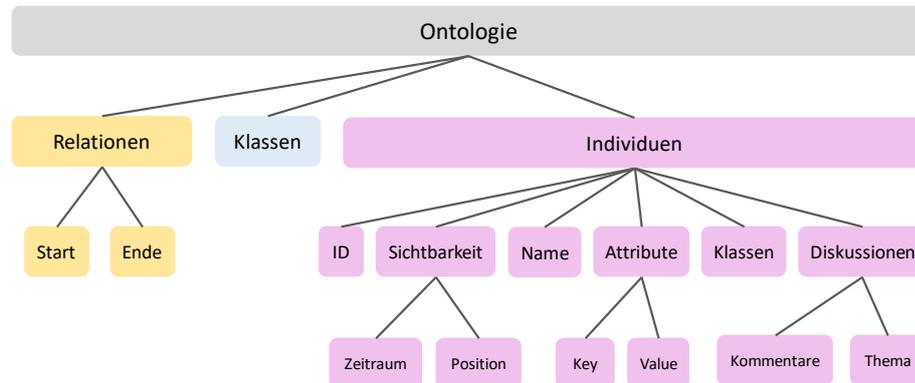


Abbildung 5.1: Aufbau des JSON-Objektes der Ontologie

### 5.3 Textbasierte Suche

In Abschnitt 4.4.2 werden zusätzliche Funktionen vorgestellt, welche im ontologiebasierten Design umgesetzt wurden. Im Rahmen dieser Bachelorarbeit wird eine Studie durchgeführt, welche unter Anderem einen Vergleich beider Designs ermöglicht. Um die Validität der Studie zu erhöhen, wurden deshalb die nicht-ontologiespezifischen Features ebenfalls im traditionellen Design umgesetzt. Somit besitzt das traditionelle Design ebenfalls eine Suchleiste, in der nach Stichworten gesucht werden kann. Die genauere Funktionsweise der Suchleiste ist in Abschnitt 4.4.2 beschrieben.

### 5.4 Aufgetretene Probleme

Während der Implementierung ist eine besondere aber essentielle Problematik aufgetreten. Bevor die Evaluation gestartet ist, wurden Generalproben unternommen. In diesen stellte sich heraus, dass die Ansicht des Prototypen mit der Auflösung des Bildschirms in Wechselwirkung steht. Das liegt daran, dass der Prototyp die Ansicht immer im Vollbildmodus anzeigt und sich an die Höhe und Breite des Monitors zusammen mit der Auflösung anpasst. Dadurch war z.B. die Schriftgröße auf einem Monitor mit einer höheren Auflösung kleiner als die Schriftgröße auf einem Monitor mit einer geringeren Auflösung. Dieses Problem wurde im Prototypen behoben, in dem die Ansicht normiert und dann mit einem Zoomfaktor an die vorhandene Bildschirmauflösung angepasst wird.

## 5.5 Grenzen des Prototypen

In erster Linie ist es wichtig zu erwähnen, dass der implementierte Prototyp keine Videos abspielen kann. Wenn der Prototyp jedoch in einem Videoprogramm eingebaut werden sollte, kann er nicht beliebig eingesetzt werden. Am besten ist er geeignet, wenn die Annotationen, wie in der Arbeit von Nagel und Karras beschrieben [27], nicht sichtbar sind während das Video läuft. Sie sollten nur in einem gestoppten Bild aktiviert werden können. Dies liegt daran, dass der Prototyp mithilfe eines rekursiven Algorithmus die Knotenmenge, welche in dem kräftebasierten Graphen enthalten ist, berechnet. Für die Berechnung muss der Zeitpunkt, in dem das Video sich befindet, angegeben werden. Wegen dieser Funktionsweise wird theoretisch zu jeder Sekunde des Videos die Knotenmenge des Graphen neu berechnet und der kräftebasierte Algorithmus startet von vorne. Würde das Video nun durchlaufen und die Annotationen aktiviert bleiben, würde der kräftebasierte Algorithmus jede Sekunde mit einer neuen Knotenmenge von vorne starten müssen. Es könnte keine konsistente Ansicht der Annotationen während des Videoablaufs angezeigt werden.

Außerdem ist keine Funktion implementiert, mit der Annotationen oder Kommentare hinzugefügt bzw. bearbeitet werden können. Des Weiteren ist das Programm in keiner Javaumgebung eingebaut und läuft lediglich im Browser. Dies hat den Nebeneffekt, dass die Datei, in der die Ontologie gespeichert ist, durch das Programm nicht modifiziert werden könnte, selbst wenn dies gewünscht gewesen wäre. Im Rahmen dieser Bachelorarbeit werden diese Funktionalitäten nicht benötigt, da die Evaluation der beiden Designs im Vordergrund stand und nicht die Bearbeitung und Erstellung von Annotationen.

Der Prototyp ist zusätzlich nicht gegen alle Fehler in der Ontologiedatei geschützt. Er prüft die Datei nicht auf ihre syntaktische und semantische Korrektheit und so kann es passieren, dass das Programm bei einer fehlerhaften Datei ohne Fehlermeldung abstürzt oder nicht die gewünschte Ontologie anzeigt. Im Rahmen dieser Bachelorarbeit wurde keine Validierungsfunktion im Programm implementiert, da für die Evaluation lediglich eine Ontologie genutzt wurde, welche im Vorhinein manuell geprüft wurde. Außerdem wäre eine solche Syntaxprüfung in einem zukünftigen Prototypen nicht zwangsweise notwendig, da die Ontologien mithilfe des Programms und nicht in einem Texteditor erstellt werden würden.

Des Weiteren können Informationen im Pop-Up Fenster nur in Form von Text dargestellt werden. In verwandten Arbeiten wurden ebenfalls Bilder, Videos, Audios oder Dateien vorgestellt. Dies wurde im Prototypen aufgrund der begrenzten Zeit allerdings nicht umgesetzt.



# Kapitel 6

## Evaluation

In diesem Kapitel wird die durchgeführte Studie genauer beschrieben. Dabei wird genauer auf die Ziele, den Aufbau, die Bedrohungen der Validität und die gemessenen Ergebnisse eingegangen.

### 6.1 Verwendetes Beispiel

Für die Studie wurde ein zur Verfügung gestelltes Visionvideo verwendet. Das darin vorgestellte Produkt ist ein intelligenter Kühlschrank, welcher diverse innovative Funktionen bietet. Aus dem Visionvideo wurde eine Momentaufnahme genommen und im Prototypen integriert.

Des Weiteren wurden zusätzliche Informationen zu den im Bild dargestellten Objekten ausgedacht. Diese Informationen wurden im traditionellen Design auf drei Annotationen aufgeteilt, wohingegen die gleichen Informationen im ontologiebasierten Design auf insgesamt 17 Annotationen aufgeteilt wurden. Die unterschiedliche Menge an Annotationen ist darauf zurückzuführen, dass angenommen wird, dass der Ersteller der Annotationen im ontologiebasierten Design dazu angeregt wird, Strukturen und Verbindungen zwischen Annotationen darzustellen, so dass er dazu tendiert, die vorhanden Informationen auf insgesamt mehr Annotationen aufzuteilen. Die für die Evaluation verwendete Ontologie besteht aus mehr Annotationen, da die Informationen in subjektiv empfunden sinnvolle Kategorien aufgeteilt und miteinander verbunden zu werden. Es folgt, dass die Menge an Informationen im Pop-Up Fenster durch die Aufteilung auf mehr Annotationen durchschnittlich geringer ist. In Abbildung 6.1 und Abbildung 6.2 sind die Designs mit dem für die Evaluation verwendeten Beispiel zu sehen.

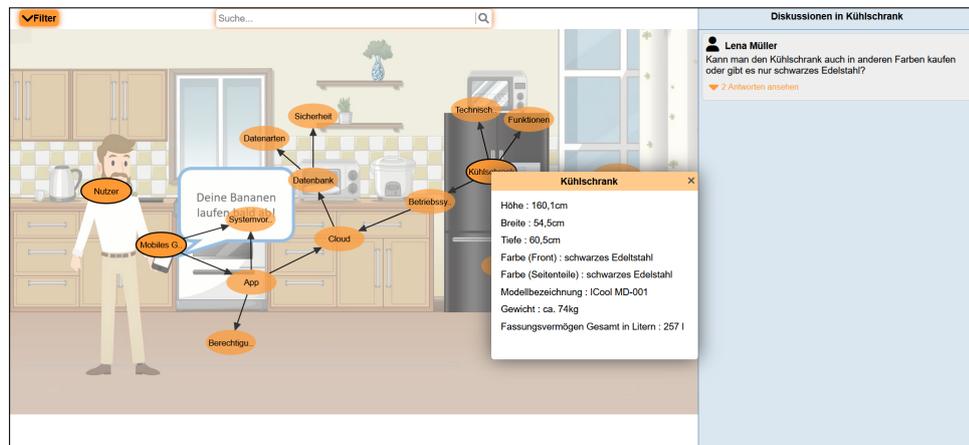


Abbildung 6.1: Ontologiebasiertes Design mit ausgeklappter Annotation „Kühlschrank“



Abbildung 6.2: Traditionelles Design mit ausgeklappter Annotation „Kühlschrank“

## 6.2 GQM

Um Metriken für die Evaluation zu bestimmen, wird das GQM-Verfahren verwendet. Ein Ziel der Evaluation ist die Untersuchung der Usability der grafischen Oberfläche. Das bedeutet, dass für dieses Ziel die allgemeinen Designentscheidungen untersucht werden sollen und nicht die Spezifitäten der ontologiebasierten oder traditionellen Konzepte aus Abschnitt 4.1. Nach dieser Untersuchung ist ein weiteres Ziel der Vergleich der Usability zwischen

den konzeptspezifischen Designs. Dafür wird die Effektivität, Effizienz und die Zufriedenheit der Nutzer für beide Designs untersucht und miteinander in Vergleich gestellt. Es wurden folgende Zielbäume entwickelt:

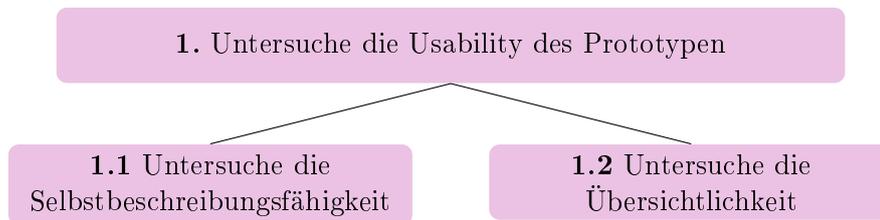


Abbildung 6.3: Zielbaum 1

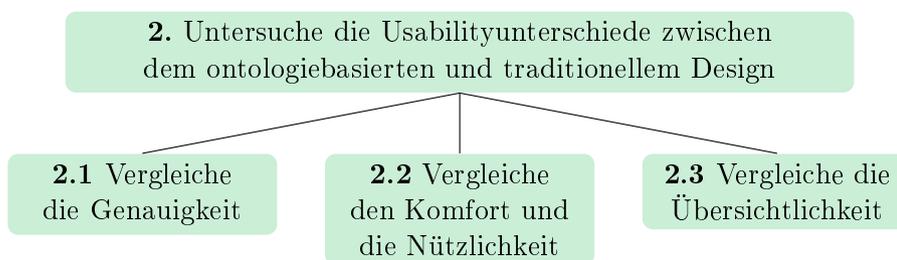


Abbildung 6.4: Zielbaum 2

Zu den Unterzielen der Zielbäume wurden Fragen und passende Metriken ausgearbeitet. Die Tabelle dazu befindet sich im Anhang.

## 6.3 Aufbau der Studie

In diesem Abschnitt wird der Aufbau des Experimentes beschrieben, wobei genauer auf die Probanden, Testgruppen, der Ablauf der Studie und die gemessenen Ergebnisse eingegangen wird.

### 6.3.1 Gewählte Probanden

Für die Studie wurden insgesamt 16 Personen ausgewählt, welche potentielle Stakeholder eines Softwareprojektes sein könnten. Insgesamt gab es neun männliche und sieben weibliche Probanden. Unter ihnen befanden sich fünf Softwareentwickler, fünf Projektleiter, fünf Studenten der Informatik und eine Person mit einer sonstigen Tätigkeit in einem Softwareunternehmen. Das Alter und der Abschluss der Probanden variierte. Zwei der Probanden besaßen die allgemeine Hochschulreife, sechs einen Bachelor, sieben einen Master und eine einen Doktor. Dabei waren acht zwischen 18 und 30 Jahre alt, vier zwischen 31 und 40, drei zwischen 41 und 50 und eine Person zwischen 51 und 60 Jahre.

### 6.3.2 Testgruppen der Studie

Die ausgewählten Probanden wurden gleichmäßig im Bezug auf ihre demografischen Daten auf zwei Testgruppen verteilt. Diese Gruppen wurden mittels *between-subjects* erstellt. Das bedeutet, dass beide Gruppen je nur ein Design bewerteten, so dass eine das ontologiebasierte und die andere das traditionelle Design evaluierte. Dadurch kann ein Lerneffekt der Probanden im Bezug auf die Designs ausgeschlossen werden. Dieser könnte die Ergebnisse der Studie verfälschen und somit ihre Vergleichbarkeit beeinflussen. In den folgenden Abschnitten wird von „Testgruppe-O“, welche das ontologiebasierte Design und der „Testgruppe-T“, welche das traditionelle Design evaluierte, gesprochen.

### 6.3.3 Ablauf der Studie

Während der Studie wurde das *Think Aloud* Protokoll verwendet, welches besagt, dass die Probanden ihre Gedanken während der Durchführung der Studie laut aussprechen sollen. Dieses Vorgehen nimmt wenig Einfluss auf die Probanden und identifiziert gleichzeitig mehr Usabilityprobleme als wenn es nicht verwendet werden würde [5].

Für die Einführung in das Ausgangsszenario wird den Probanden ein selbstgedrehtes Video gezeigt. Darin wird das Szenario, in das sich die Probanden einführen sollen, erklärt, sowie das verwendete Visionvideo gezeigt. Mithilfe des Videos wird die Validität der Studie erhöht, da alle Probanden exakt die gleichen Informationen erhalten.

Vor dem Start der Studie wurden einige Vortests unternommen um Mängel des geplanten Ablaufs zu finden und vorzeitig zu beseitigen. Dabei stellte sich heraus, dass viele Personen nicht alle Funktionen der Designs entdeckten, nutzten oder dessen Nützlichkeit in Betracht zogen. In der Studie hätte es dazu geführt, dass die Ergebnisse von diesem Faktor beeinflusst gewesen wären. Ein unverfälschter Vergleich zwischen den beiden Designs wäre nicht möglich gewesen, da die unterschiedlichen Ergebnisse zwischen den Gruppen nicht automatisch auf die unterschiedlichen Konzepte zurückgeführt werden könnten.

Deshalb wurde der Ablauf letztendlich in zwei Durchführungsblöcke aufgeteilt. Im ersten Block wird die intuitive Bedienung und der Aufbau der allgemeinen grafischen Oberfläche evaluiert. Daraufhin wird den Probanden eine kleine Einführung in Form eines Videos für alle Funktionen des Programms gegeben. Das Tutorial beinhaltet Erklärungen und eine Aufnahme des Bildschirms, während beispielhafte Interaktionen mit dem Programm vorgestellt werden. Dadurch erhalten alle Probanden exakt die gleichen Informationen.

Dank des Tutorial besitzen die Probanden ab diesem Zeitpunkt der Studie den gleichen Wissensstand, so dass alle Funktionen von jedem Probanden

voll ausgeschöpft werden können. Dadurch kann ab diesem Zeitpunkt der Studie eine bessere Vergleichbarkeit der Ergebnisse gewährleistet werden. Der genaue Ablauf der Studie wird in der Tabelle 6.3.3 dargestellt.

Schritt	Beschreibung	Genutzte Materialien
1	Sammele demografische Daten	Fragebogen
2	Video zur Einführung in das Szenario	Selbstgedrehtes Video
3	5 min. Ausprobierzeit	Prototyp
4	Durchführung (Teil 1)	Prototyp
5	Sammlung von subjektiven Erfahrungen	Fragebogen zu den Zielen 1.1 und 1.2
6	Tutorial zu den Funktionsweisen des Programms	Selbstgedrehtes Video
7	Durchführung (Teil 2)	Prototyp
8	Sammlung von subjektiven Erfahrungen	Fragebogen zu den Zielen 2.1, 2.2 und 2.3

Tabelle 6.1: Ablauf der Studie

### 6.3.4 Gestellte Aufgaben

In den Durchführungsteilen werden den Probanden Aufgaben gestellt, welche sie mithilfe des Prototypen lösen sollen. Die Aufgaben beziehen sich auf Inhalte der Annotationen. Im Durchführungsteil nach dem Tutorial werden fünf Aufgaben gestellt, wobei es zwei Aufgaben gibt, dessen Antworten in den Informationen im Pop-up Fenster gefunden werden können, zwei Aufgaben, dessen Antworten in der Diskussionsleiste gefunden werden können und eine Aufgabe zum Verständnis. Im traditionellen Design muss die Antwort auf die Verständnisaufgabe inhaltlich abgeleitet werden, im ontologiebasierten Design gibt es zusätzlich die Möglichkeit die Antwort an der Struktur und den Verbindungen der Annotationen abzulesen.

Die Aufgaben werden in unterschiedliche Kategorien aufgeteilt, da sich ihre Lösungen an verschiedenen Stellen im Prototypen befinden. Um die objektiv gemessenen Ergebnisse der Testgruppen besser interpretieren und vergleichen zu können, werden die Aufgaben in Gruppen aufgeteilt.

## 6.4 Ergebnisse

In diesem Kapitel werden die gemessenen Ergebnisse der Studie aufgelistet. Dabei werden in einem Unterkapitel die Ergebnisse zu der allgemeinen grafischen Oberfläche zum Oberziel aus Abbildung 6.2 und in dem anderen

Unterkapitel die Ergebnisse zu den Unterschieden der Designs zum Oberziel aus Abbildung 6.2 aufgelistet. Des Weiteren setzt sich ein Unterkapitel genauer mit spezifischen Ergebnissen der Testgruppe-O auseinander.

Die Ergebnisse der Fragebögen wurden metrisch ausgewertet, in dem die 5-Point-Likert-Skala in die Werte fünf (Stimme vollkommen zu) bis eins (Stimme überhaupt nicht zu) übersetzt wurde.

#### 6.4.1 Allgemeines Design

Die Ergebnisse der Fragen zum allgemeinen Design sind zwischen beiden Testgruppen statistisch nicht signifikant unterschiedlich. Deshalb werden sie für die Auswertung der allgemeinen grafischen Oberfläche zusammengefasst bzw. deskriptiv analysiert.

13 Probanden stimmten vollkommen zu, einer war unentschieden und zwei stimmten eher nicht zu, dass der allgemeine Aufbau der grafischen Oberfläche übersichtlich ist. Dabei kommentierte ein Proband, dass die Farbunterscheidungen im Design gut deutlich seien. Insgesamt wurde zu dieser Frage ein Durchschnitt von 4,5 und ein Median von 5 erreicht.

Mehrere Probanden hinterließen einen positiven Kommentar zur intuitiven Bedienung der grafischen Oberfläche. Dies spiegelte sich ebenfalls in den Antworten des Fragebogen wieder. 15 der Probanden stimmten vollkommen und einer eher zu, dass die Bedienung des Programms schnell verständlich ist. Der Median liegt somit bei 5 und der Durchschnitt bei 4,9. Die gleichen Werte wurden dafür erreicht, dass das Programm lediglich beabsichtigte Aktionen ausführt. Das bedeutet, dass ebenfalls 15 Probanden vollkommen und eine eher zustimmte und ein Durchschnitt von 4,9 bzw. ein Median von 5 erreicht wurde.

Vor dem Tutorial wurde abgefragt, ob die Probanden alle Funktionen des Programms wahrgenommen haben. Dabei stellte sich heraus, dass zwei der acht Personen der Testgruppe-T die Suchleiste und eine der acht Personen der Testgruppe-O den Filter nicht gesehen haben.

#### 6.4.2 Unterschiede zwischen den Designs

In diesem Unterkapitel werden die spezifischen Usabilityunterschiede zwischen den Designs untersucht. Dabei wird mithilfe eines T-Tests überprüft, ob die Ergebnisse statistisch signifikant unterschiedlich sind. Bei der Nutzung des T-Tests werden eine Normalverteilung der Daten, eine gleichmäßige Streuung der Daten und zwei voneinander unabhängige Testgruppen vorausgesetzt. Die Testgruppe-O und Testgruppe-T sind komplett voneinander unabhängig, da der Aufbau von *between-subjects* gewählt wurde. Mithilfe des Kolmogorov-Smirnov Tests wurden die Daten auf die Normalverteilung und Streuung geprüft. Die statistische Signifikanz wird durch den berechneten p-Wert bestimmt. Ist  $p < 0,05$ , so ist ein statistisch signifikanter Unterschied

gefunden.

Für die Auswertung der objektiven Metriken wurde, wie in Abschnitt 6.3.4 erwähnt, zwischen unterschiedlichen Aufgabentypen unterschieden. Dies ermöglicht eine bessere Interpretationen der Ergebnisse.

### **Richtig gelöste Aufgaben**

Jeder Proband löste fünf Aufgaben, so dass von allen 16 Probanden insgesamt 80 Aufgaben gelöst wurden. Die Testgruppen weisen keinen statistisch signifikanten Unterschied im Bezug auf die Anzahl richtig gelöster Aufgaben auf. Deshalb werden die Ergebnisse der beiden Testgruppen deskriptiv analysiert. Insgesamt war die Anzahl der richtig gelösten Aufgaben bei informationsbezogenen Aufgaben bei 31 von 32 (97%), bei diskussionsbezogenen Aufgaben bei 18 von 32 (56%) und bei der Verständnisaufgabe bei zehn von 16 (63%). Dabei waren die Ergebnisse der einzelnen Testgruppen wie folgt: die Testgruppe-O hat 15 von 16 (94%) und die Testgruppe-T 16 von 16 (100%) der informationsbezogenen Aufgaben richtig gelöst. Außerdem haben beide Testgruppen je neun von 16 (56%) der diskussionsbezogenen Aufgaben richtig gelöst. Die Verständnisaufgabe wurde von beiden Testgruppen fünf von acht Mal (63%) richtig gelöst.

### **Dauer und Effizienz**

Die Testgruppe-T benötigte zum Lösen der Aufgaben statistisch signifikant mehr Zeit. Allerdings sind lediglich die diskussionsbezogenen und informationsbezogenen Aufgaben betroffen, die Verständnisfrage weist keine statistisch signifikanten Unterschiede auf. In Abbildung 6.5 sind die Werte zur Dauer in einem Boxplotdiagramm dargestellt.

Die Effizienz jedes einzelnen Probanden wird berechnet, in dem die Anzahl der richtig gelösten Aufgaben dividiert wird durch die Dauer zum Lösen der Aufgaben. Die berechneten Werte für die Testgruppen werden deskriptiv analysiert, das sie keine statistisch signifikanten Unterschiede aufweisen. Die durchschnittliche Effizienz liegt bei den informationsbezogenen Aufgaben bei 0,079 pro Person, bei den Diskussionsbezogenen Aufgaben bei 0,042 pro Person und bei der Verständnisaufgabe bei 0,044 pro Person.

### **Übersichtlichkeit der Designs**

Im Fragebogen wurden Fragen zu der subjektiv wahrgenommenen Übersichtlichkeit gestellt. Dabei wurde zwischen der empfundenen Übersichtlichkeit der Menge an Annotationen, der Textmenge im Pop-up Fenster und der Menge an Diskussionen in der Seitenleiste unterschieden.

Die Testgruppe-O empfand die Übersichtlichkeit der Menge an Annotationen als etwas geringer (Durchschnitt = 4,5 und Median = 5) als die Testgruppe-T (Durchschnitt = 5 und Median = 5). Allerdings ist

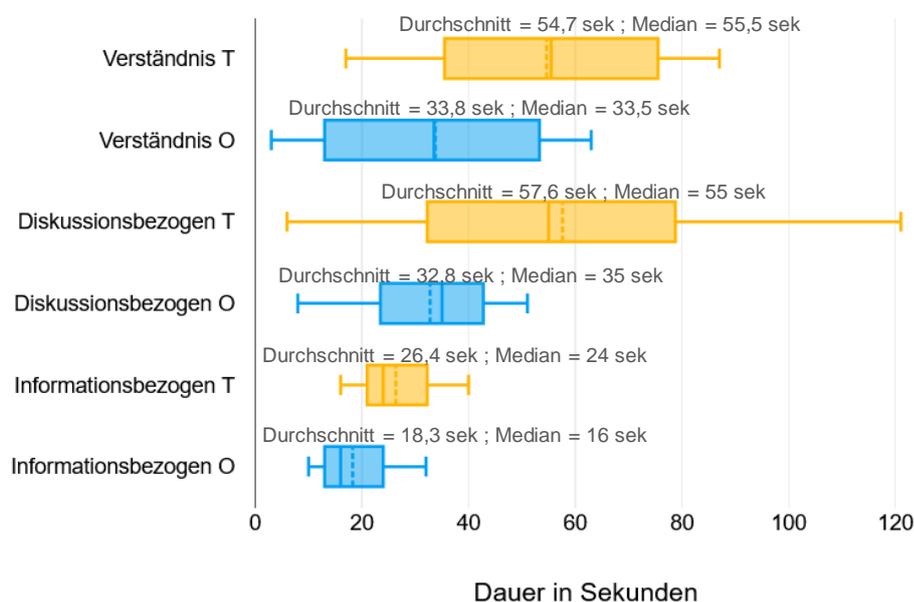


Abbildung 6.5: Werte zur Dauer zum Lösen der Aufgaben pro Proband. Die gestrichelte Linie stellt den Durchschnitt und die durchgezogene Linie stellt den Median dar.

der Unterschied zwischen den Testgruppen statistisch nicht signifikant ( $p = 0,103539$ ). Deskriptiv analysiert beträgt der Durchschnitt 4,8 und der Median 5. Die Testgruppe-O empfand die Diskussionen als übersichtlicher (Durchschnitt = 4,6 und Median = 5) als die Testgruppe-T (Durchschnitt = 3,9 und Median = 4,5). Allerdings ist der Unterschied zwischen den Gruppen statistisch nicht signifikant ( $p = 0,095924$ ). Ein Proband kommentierte, dass das Auf- und Zuklappen der Diskussionsthemen besonders praktisch sei.

Die Testgruppe-T empfand die Übersichtlichkeit der Textmenge im Pop-up Fenster jedoch als statistisch signifikant schlechter ( $p = 0,005167$ ). Abbildung 6.6 zeigt ein Boxplotdiagramm mit den Werten und dem Durchschnitt der Gruppen.

### Suchleiste

Die Testgruppe-T nutzte die Suchleiste bei den informationsbezogenen Aufgaben und der Verständnisaufgabe statistisch signifikant öfter. Bei den diskussionsbezogenen Aufgaben gibt es keinen statistisch signifikanten Unterschied. In Abbildung 6.7 sind die Werte zur Nutzung der Suchleiste in einem Boxplotdiagramm dargestellt.

Trotz des statistisch signifikanten Unterschied der Nutzung, weist die subjektiv wahrgenommene Nützlichkeit der Suchleiste keine statistisch si-

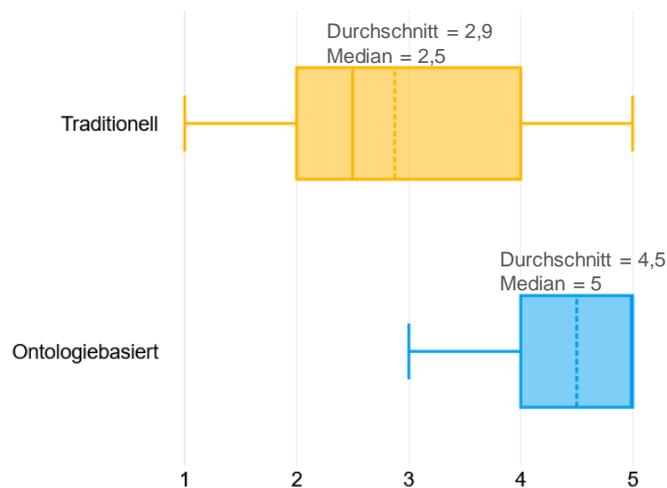


Abbildung 6.6: Fragebogendaten zur Übersichtlichkeit der Informationen im Pop-up Fenster pro Proband. Die gestrichelte Linie stellt den Durchschnitt und die durchgezogene Linie stellt den Median dar.

signifikanten Unterschiede zwischen den Gruppen auf ( $p = 0,30928$ ). Für Testgruppe-O liegt der Durchschnitt bei 4,3 und der Median bei 5 und für die Testgruppe-T liegt der Durchschnitt bei 4,5 und der Median bei 5. Deskriptiv analysiert empfanden die Probanden die Suchleiste insgesamt als hilfreich (Durchschnitt = 4,4 und Median = 5).

Allgemein empfanden beide Gruppen das Tutorial ähnlich hilfreich und weisen keine statistisch signifikanten Unterschiede auf ( $p = 0,201048$ ). Deskriptiv analysiert liegt der Durchschnitt im Fragebogen bei 4,1 und der Median bei 4,5.

### 6.4.3 Ontologiespezifische Ergebnisse

Die Testgruppe-O beantwortete eine Frage zu der Nützlichkeit der Verbindungen und speziell zu ihren Richtungen. Die Ergebnisse weisen eine statistisch signifikante Unterscheidung auf ( $p = 0,018606$ ). In Abbildung 6.8 sind die Werte und die Durchschnitte der Ergebnisse in einem Boxplotdiagramm dargestellt.

Allgemein stimmte die Testgruppe-O zu, dass die Bewegungen zwischen den losen Annotationen nicht verwirrend sind (Durchschnitt = 4,6 und Median = 5).

Auffallend war, dass alle acht Probanden der Testgruppe-O vollkommen zustimmten, dass die Aufteilung der Diskussionen auf die einzelnen Annotationen nützlich gewesen ist. Der Durchschnitt und der Median dieser Werte liegen bei 5. Außerdem war auffallend, dass der Filter im

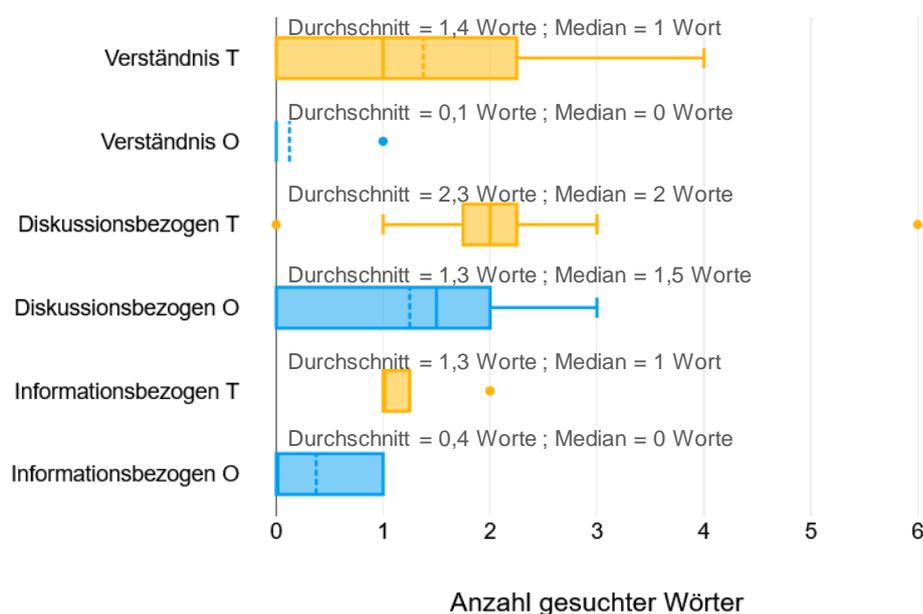


Abbildung 6.7: Werte zur Nutzung der Suchleiste zum Lösen der Aufgaben pro Proband. Die gestrichelte Linie stellt den Durchschnitt und die durchgezogene Linie stellt den Median dar.

ontologiespezifischen Design insgesamt nur einmal von einem Probanden beim Lösen einer Aufgabe verwendet wurde. Er wurde von den Probanden als eher neutral empfunden (Durchschnitt = 2,6 und Median = 3).

#### 6.4.4 Verbesserungsvorschläge

In den Fragebögen wurden Kommentare zu Verbesserungsvorschlägen gesammelt. Ein Vorschlag war, dass das Pop-Up Fenster verschiebbar und in der Größe veränderbar sein sollte. Außerdem könnten die Bezeichner der Informationen im Pop-Up Fenster besser formatiert sein wie z.B. in dicker Schrift, damit sie auf den ersten Blick erkennbar sind. Des Weiteren wurde eine automatische Rechtschreibkorrektur für die Suchleiste erwähnt. Speziell die Testgruppe-T schlug vor, dass der Inhalt des Pop-up Fensters mithilfe eines Inhaltsverzeichnisses kategorisiert werden könnte.

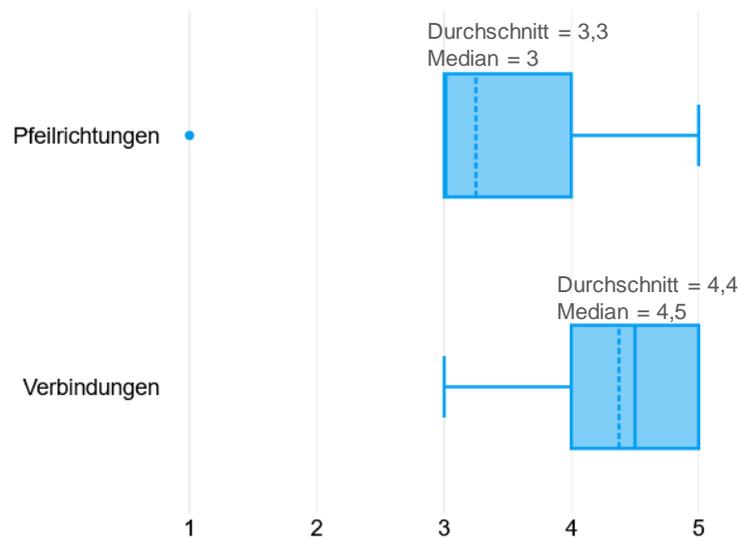


Abbildung 6.8: Fragebogendaten zur Nützlichkeit der Kanten zwischen losen Annotationen des ontologiebasierten Designs. Die gestrichelte Linie stellt den Durchschnitt und die durchgezogene Linie stellt den Median dar.



# Kapitel 7

## Diskussion

Dieses Kapitel interpretiert die Ergebnisse der Studie, beschreibt die Grenzen der Validität und gibt einen Ausblick für zukünftige Forschungen.

### 7.1 Interpretation der Ergebnisse

Die Interpretationen in diesem Kapitel basieren auf den gemessenen Ergebnissen der Studie.

#### 7.1.1 Allgemeines Design

Das allgemeine Design des Prototypen ist verständlich und übersichtlich aufgebaut. Des Weiteren ist die Bedienung intuitiv schnell verständlich und der Prototyp führt nur vom Nutzer erwartete Funktionen aus. Das Design der Diskussionen in der Seitenleiste ist ebenfalls intuitiv verständlich und durch die Verschachtlung in Diskussionsthemen gut strukturiert.

#### 7.1.2 Effektivität und Effizienz

Allgemein lässt sich sagen, dass beide Designs eine gleiche Effektivität aufweisen, da es keine statistisch signifikanten Unterschiede zwischen den Ergebnissen der Testgruppen gibt.

Allerdings ist die Effektivität zwischen den Aufgabengruppen selbst statistisch signifikant unterschiedlich. Bei den informationsbezogenen Aufgaben war sie deutlich höher als bei den diskussionsbezogenen. Dies ist darauf zurückzuführen, dass die Informationen in Key-Value Paaren dargestellt werden während die Diskussionen aus ganzen Sätzen und Gesprächssträngen bestehen. Dadurch mussten die Probanden bei den Diskussionen mehr Text überfliegen und gleichzeitig interpretieren, während sie bei den Informationen nach gezielten Stichworten im Key des Key-Value Paares suchen konnten. Auffällig ist, dass die Strukturierung der Informationen in Key-Value Paare eine sehr hohe Effektivität aufweist, da sie deskriptiv analysiert bei 97% und

somit beinahe 100% liegt. Die Effektivität der Diskussionen hingegen liegt lediglich bei 56%, so dass es sich im Rahmen der implementierten Konzept empfiehlt, möglichst viele Informationen strukturiert und gebündelt in Key-Value Paaren im Pop-up Fenster darzustellen.

Beide Designs sind ähnlich effizient, da es zu diesem Wert keine signifikanten Unterschiede zwischen den Testgruppen gibt. Unter den unterschiedlichen Aufgaben wiesen die informationsbezogenen Aufgaben die höchste Effizienz auf, da sie am häufigsten und durchschnittlich schnellsten richtig beantwortet wurden.

### 7.1.3 Subjektiv wahrgenommene Übersichtlichkeit

Allerdings gab es zwischen den Testgruppen signifikante Unterschiede im Bezug auf die Dauer zum Lösen der Aufgaben, die Nutzung der Suchleiste sowie die empfundene Übersichtlichkeit. Diese Ergebnisse stehen miteinander in Wechselwirkung und lassen Aussagen über die subjektiv wahrgenommene Übersichtlichkeit zu. Im Folgenden wird zwischen verschiedenen Aspekten der ontologiebasierten Konzepte unterschieden.

#### Layout der Annotationen

Allgemein empfanden beide Gruppen die dargestellte Annotationsmenge statistisch nicht signifikant unterschiedlich und insgesamt übersichtlich. Des Weiteren antwortete die Testgruppe des ontologiebasierten Designs, dass die Bewegungen der losen Annotationen nicht verwirrend seien.

Es lässt sich festhalten, dass die Ansicht der umgesetzten ontologiespezifischen Konzepte trotz der größeren Menge an Annotationen nicht zwangsweise weniger übersichtlich ist. Das bedeutet, dass die Darstellung mehrerer sich bewegender Annotationen und deren Zusammenhängen gleich übersichtlich sein kann wie wenige unabhängige Annotationen. Allerdings kann anhand der gemessenen Ergebnisse nicht ausgeschlossen werden, dass die Ansicht des ontologiebasierten Designs unübersichtlich werden würde, wenn deutlich mehr Annotationen dargestellt wären.

#### Informationen im Pop-up Fenster

Die Testgruppen empfanden die Informationen im Pop-up Fenster statistisch signifikant unterschiedlich. Dabei antworteten die Probanden des traditionellen Designs, dass die Textmenge weder übersichtlich noch unübersichtlich sei, während die Probanden des ontologiebasierten Designs sie als sehr übersichtlich empfanden. Gleichzeitig wurde ein statistisch signifikanter Unterschied unter den Testgruppen im Bezug auf die Nutzung der Suchleiste festgestellt. Zum Lösen der informationsbezogenen Aufgaben, nutzten die Probanden des traditionellen Designs sie signifikant öfter. Die Testgruppe mit

dem ontologiebasierten Design nutzte die Suchleiste meist erst nachdem die selbstständige Suche in den Annotationen aufwendig oder langwierig schien.

In Anbetracht dieser Ergebnisse kann die Nutzung der Suchleiste für informationsbezogene Aufgaben als Indikator für die subjektiv empfundene Übersichtlichkeit interpretiert werden. Dadurch, dass die Probanden des ontologiebasierten Designs die Informationen als sehr übersichtlich empfanden, waren sie eher dazu angeregt, selbstständig in den Annotationen zu suchen. Dafür ordneten sie die gesuchten Informationen in Themenkategorien ein und suchten gezielt in den Annotationen. Währenddessen empfand die Testgruppe des traditionellen Designs die Informationen als neutral übersichtlich, so dass sie die Suchleiste als Hilfsmittel zum Finden von Informationen präferierten.

### **Diskussionen in der Seitenleiste**

Die Diskussionen wurden von beiden Testgruppen statistisch nicht unterschiedlich übersichtlich empfunden. Durchschnittlich empfanden beide Gruppen sie als eher übersichtlich. Gleichzeitig nutzten beide Gruppen die Suchleiste ähnlich häufig zum Lösen von diskussionsbezogenen Aufgaben.

Die statistische Ähnlichkeit der Ergebnisse kann darauf zurückgeführt werden, dass die Diskussionen im Vergleich zu den Informationen im Pop-up Fenster komplette Sätze und Gesprächsstränge enthielten. Aufgrund dessen sind viele redundante Informationen in den Diskussionen enthalten, so dass die Probanden mehr Aufwand aufbringen müssen, um die für sie relevanten Informationen herauszufiltern. Deshalb wurde die Suchleiste von beiden Testgruppen als Hilfsmittel zum Überfliegen und Filtern der Texte genutzt.

Trotz der ähnlich empfundenen Übersichtlichkeit und Häufigkeit der Nutzung der Suchleiste zwischen den Gruppen, benötigte die Testgruppe des traditionellen Designs statistisch signifikant mehr Zeit zum Lösen der diskussionsbezogenen Aufgaben. Die kürzere Dauer der Probanden des ontologiebasierten Designs kann durch die Aufteilung der Diskussionen auf die Annotationen begründet werden. Während die Probanden des traditionellen Designs alle vorhandenen Diskussionen überfliegen mussten, wurden sie für die Testgruppe des ontologiebasierten Designs bereits thematisch strukturiert und vorgefiltert. Die Ergebnisse des Fragebogens unterstützen diese Annahme, da ausnahmslos alle Probanden der Testgruppe-O die Zuordnung der Diskussionen auf eine Annotation als hilfreich empfanden.

#### **7.1.4 Darstellung von Zusammenhängen**

Ursprünglich wurde angenommen, dass die Testgruppe des ontologiebasierten Designs weniger Zeit zum Lösen der Verständnisaufgabe benötigen, eine höhere Effektivität und höhere Effizienz erzielen würde. Die Annahme gründete darauf, dass die Testgruppe-O die Möglichkeit hatte, die Lösung der

Aufgabe an der Struktur der Ontologie abzulesen während die Testgruppe-T die Lösung aus den Texten interpretieren musste.

Allerdings weist die Effektivität und Effizienz zwischen beiden Testgruppen statistisch keinen signifikanten Unterschied auf. Trotz, dass die Aufgabe von den Probanden der ontologiebasierten Gruppe durchschnittlich schneller beantwortet wurde, ist der Unterschied zur Testgruppe des traditionellen Designs statistisch nicht signifikant.

Es war auffällig, dass viele Probanden der ontologiebasierten Testgruppe die Strukturen und Zusammenhänge zwischen den Annotationen nicht als Informationsquelle wahrgenommen haben und die Lösung der Verständnisaufgabe im Pop-up Fenster suchten.

Die Verbindungen zwischen den Annotationen wurden trotz dessen von den Probanden als hilfreich angesehen, obwohl sie beim Lösen der Verständnisfrage nicht signifikant zum Lösen der Verständnisaufgabe genutzt oder interpretiert wurden. Es wurde nicht genauer untersucht, für welchen Zweck die Verbindungen von den Probanden als hilfreich angesehen wurden. Allerdings wird angenommen, dass sie die Darstellung der Verbindungen mit dem Nicht-Darstellen der Verbindungen bei der gleichen Annotationsmenge verglichen haben. Dahingegen werden die Pfeilrichtungen der Verbindungen selbst von den Probanden als neutral hilfreich betrachtet. Das Ergebnis unterstützt die Annahme, dass die Probanden den Details der Darstellung der Ontologie eine eher neutrale Bedeutung beigemessen haben.

### 7.1.5 Zusammenfassung

Zusammenfassend lässt sich sagen, dass die umgesetzten ontologiespezifischen Konzepte im Vergleich zu den traditionellen Konzepten nicht zwangsweise effektiver oder effizienter sind. Allerdings können sie Stakeholder dazu anregen mehr Annotationen zu erstellen um diese miteinander in Verbindung zu setzen und Strukturen zu visualisieren. Dadurch können die ontologiespezifischen Konzepte zu einer übersichtlicheren Aufteilung der Informationen führen. Außerdem ermöglichen sie, dass Informationen und Diskussionen im Vergleich zu traditionellen Konzepten signifikant schneller gefunden werden.

Leider wurden nicht alle Vorteile der Darstellung einer Ontologie von den Probanden ausgeschöpft, da die Strukturen und Zusammenhänge zwischen den Annotationen nicht genauer analysiert oder interpretiert wurden. Das bedeutet jedoch nicht, dass sie nicht nützlich für die intuitive Erforschung oder übersichtliche Darstellung der Annotationen sind. Tatsächlich wurden die repräsentierten Verbindungen von den Probanden des ontologiebasierten Designs trotzdem als hilfreich empfunden.

Insgesamt zeigen die gemessenen Usabilityunterschiede positive Tendenzen für die ontologiebasierten Konzepte und veranschaulichen ihr Potential im Kontext von Visionvideos. Deshalb ist es für zukünftige Arbeiten loh-

nenswert weitere oder verbesserte Konzepte zu entwickeln und zu evaluieren um die Vorteile aller potentiellen Darstellungsformen von Annotationen bestmöglich auszuschöpfen.

## 7.2 Bedrohung der Validität

In erster Linie ist es wichtig zu erwähnen, dass die Studie lediglich ein Szenario simuliert hat. Das Visionvideo selbst stammt nicht aus einer realen Firma mit einem realen Produkt, sondern wurde von Studierenden zu einem fiktiven Produkt entwickelt. Die gewählten Probanden sind keine echten Stakeholder, da das vorgestellte Szenario erfunden ist. Außerdem haben einige der Probanden das Konzept von Visionvideos vorher nicht gekannt, so dass es ihnen eventuell schwerer gefallen ist, sich in das vorgegebene Szenario einzufühlen. Diese Bedrohung wird als minimal eingestuft, da die Darstellung von konkreten Mensch-System Interaktionen in einem Videoclip intuitiv verständlich ist. Zusätzlich wurde jeder Testperson eine Einführung in das Ausgangsszenario gegeben und ein Visionvideo gezeigt. Es wurden keine Rück- oder Verständnisfragen von den Probanden gestellt.

Ein weiterer Punkt ist, dass die Ergebnisse stark davon abhängen, in welchem Umfang das Programm genutzt wird. In dieser Studie wurde relativ viel Aufwand in die Erstellung der Annotationen und ihren Zusammenhänge gesteckt. Die verwendete Ontologie wurde über drei Wochen hinweg mehrere Male iterativ verbessert. Es ist nicht geklärt, wie viel Aufwand eine reale Firma für die Erstellung der Annotationen betreiben würde, und ob die für die Studie erstellte Informationsmenge und Strukturierung demnach realistisch ist.

Allgemein hängen die Ergebnisse der Studie stark von der dargestellten Ontologie ab. Dazu gehört die Menge an Individuen sowie deren Relationen. Es könnte demnach passieren, dass die Stakeholder mit denselben Informationen unterschiedliche Ontologien erstellen. In der Vorbereitung der Studie wurde die Qualität der verwendeten Ontologie nicht überprüft, da dies den Umfang der Arbeit überschritten hätte. Somit geben die im Rahmen dieser Arbeit verfügbaren Informationen keine Kenntnisse darüber, inwiefern die in der Studie verwendete Ontologie intuitiv und realistisch ist.

Außerdem wurde für den Aufbau der Studie intuitiv angenommen, dass die Ersteller der Annotationen bei den ontologiespezifischen Konzepten dazu angeregt werden, mehr Annotationen zu erstellen. Allerdings wurde nicht genauer untersucht, ob oder wie viele Stakeholder dazu tendieren. Für die Untersuchung hätte eine weitere Studie durchgeführt werden müssen, welche den Rahmen und Zeitraum dieser Bachelorarbeit überschritten hätte.

Ein letzter Punkt ist, dass die Evaluation online stattgefunden hat und nicht gewährleistet werden konnte, dass jeder Proband einen gleich großen Bildschirm hat. Die Größe des Bildschirms könnte die subjektiv empfundene

Übersichtlichkeit beeinflussen. Außerdem hätten die Probanden von ihrem Umfeld wie z.B. Lärm abgelenkt werden können. Allerdings wurde dies im Verlauf der Studie nicht beobachtet.

Alle gemessenen Ergebnisse der Studie können somit nicht zweifelsfrei auf die reale Welt der Softwareentwicklung übertragen werden.

## Kapitel 8

# Zusammenfassung und Ausblick

### 8.1 Zusammenfassung

Für den Erfolg eines Softwareprojekts ist es wichtig, dass es zwischen den Stakeholdern keine Missverständnisse über die Anforderungen an die Software gibt. Visionvideos können bei der Anforderungserhebung besonders hilfreich sein, da sie konkrete Szenarien aufzeigen, verständlich sind und einen guten Überblick geben. Stakeholder tendieren dazu, aufmerksamer zu sein, wenn sie mit dem Visionvideo interagieren können. Eine Möglichkeit dafür sind Annotationen im Videobild, welche jedoch unterschiedlich eingesetzt werden können. In dieser Bachelorarbeit werden sie als zusätzliche Informationsquelle betrachtet.

Außerdem beschäftigt sich diese Arbeit mit einer übersichtlichen Visualisierung von zusammenhängenden Annotationen und interpretiert sie als Ontologien. Es werden unterschiedliche ontologiespezifische Visualisierungstechniken diskutiert und implementiert.

Die Ontologien bzw. Annotationen werden als kräftebasierte Graphen visualisiert. Lose Annotationen spielen bei den Konzepten eine besonders wichtige Rolle. Sie werden vom kräftebasierten Algorithmus übersichtlich auf dem Videobild angeordnet und ihre Sichtbarkeit wird mithilfe eines rekursiven Algorithmus berechnet. Außerdem werden mehrere ontologiespezifische Darstellungsaspekte im Graphen eingebunden wie z.B. die Länge der Kanten und die berechneten Startpositionen der kräftegesteuerten Annotationen.

Eine Evaluation bestätigte, dass die umgesetzten ontologiebasierten Konzepte Vorteile für die Übersichtlichkeit der Informationen bringen können. Die Ergebnisse hängen jedoch stark von der erstellten Ontologie ab. Außerdem stellte sich heraus, dass die textbasierte Suche ein essentielles Hilfsmittel zum gezielten Suchen von Informationen ist.

Insgesamt kann gesagt werden, dass die vorgestellten ontologiespezifischen

schen Konzepte sich gut für die übersichtliche Darstellung von Verbindungen eignen. Der Ersteller der Annotationen erhält die Möglichkeit beliebige Strukturen zu visualisieren ohne sich größere Gedanken über das Layout zu machen. Gleichzeitig kann die Darstellung von Zusammenhängen einen positiven Einfluss auf die Usability haben. Letztendlich sind die umgesetzten ontologiespezifischen Konzepte eine gute Erweiterung der bisherigen. Das bedeutet, dass der Ersteller die Annotationen wie in verwandten Arbeiten als unabhängige und festgesetzte Punkte darstellen kann. Allerdings erhält er zusätzlich die Möglichkeit Annotationen als komplex strukturierte Graphen darzustellen.

Eine weitere Erforschung bezüglich der Kombination von Ontologien und Annotationen im Kontext von Visionvideos könnte weitere bisher nicht diskutierte Vorteile und Möglichkeiten aufzeigen.

## 8.2 Ausblick

Im Rahmen dieser Bachelorarbeit entstehen mehrere Ideen und weitere Forschungsthemen für zukünftige Arbeiten, welche das Potential von Ontologien in Visionvideos weiter entdecken und beurteilen könnten. Im Folgenden wird genauer auf die Möglichkeiten der vorgestellten ontologiebasierten Konzepte eingegangen.

### Erforschung der Ontologie

Ein wichtiger Aspekt ist die Untersuchung der Ontologien und der Grad der Aufteilung der Informationen. In dieser Arbeit wurde festgestellt, dass eine Aufteilung der Informationen auf mehrere zusammenhängende Annotationen übersichtlicher sein kann. Allerdings wurde nicht genauer untersucht, ab welcher Informationsmenge genau dies übersichtlicher ist und ob eine zu starke Aufteilung sich gegebenenfalls negativ auf die Usability auswirken könnte.

Der Filter hat im ontologiebasierten Design den Nutzen eine Überblick über das Layout der Annotationen zu geben. Bei einer vergleichsweise geringen Menge an Annotationen hat er deshalb wenig Nutzen. Je mehr Annotationen es im Visionvideo gibt, desto größer könnte jedoch sein Nutzen werden. In dieser Bachelorarbeit ist nicht genauer untersucht worden, wie viele Annotationen im Kontext von Visionvideos realistisch und nötig sind. Die Untersuchung des realistischen Umfangs der Ontologie und ihr Einfluss auf die Nützlichkeit des Filters könnte in weiteren Arbeiten genauer untersucht werden.

Ein weiterer Punkt ist, dass im Rahmen dieser Bachelorarbeit angenommen wurde, dass die vorgestellten ontologiespezifischen Konzepte den Ersteller der Annotationen dazu anregen, mehr Annotationen zu erstellen. Diese Annahme könnte in weiteren Forschungsarbeiten untersucht werden.

## **Erforschung der Konzepte**

In dieser Arbeit wurde die Intuitivität der Darstellung als Graph evaluiert. Dabei wurde festgestellt, dass die Probanden die Strukturen und Zusammenhänge der Ontologie größtenteils nicht genauer betrachtet haben. Es ist unklar, welchen Einfluss sie auf die Effektivität und Effizienz haben könnten, wenn die Nutzer explizit auf die Nützlichkeit der Annotationsstrukturen hingewiesen werden würden.

Weitere Untersuchungen der Usability von den in dieser Arbeit vorgestellten Konzepten sind ein interessantes Forschungsthema. Mit einem Beispiel, wo in beiden Designs die gleiche Menge an Annotationen sichtbar sind, könnte die Nützlichkeit der Darstellung von Verbindungen, Pfeilrichtungen oder gar Pfeilbeschriftungen zwischen Annotationen genauer untersucht werden.

Die textbasierte Suchfunktion könnte weiter verbessert werden, in dem ihre Fehleranfälligkeit verringert wird. Dafür könnte die Suchleiste bei ihrer Nutzung eine Liste aller in der Ontologie verfügbaren Stichworte anzeigen. Das bedeutet, dass sie das eingegebene Wort mit den verfügbaren Stichworten abgleicht und dynamisch auf jeden eingegebenen Buchstaben reagiert. Da der Stakeholder die Begriffe nicht ausschreiben müsste, könnten Rechtschreibfehler vermindert werden. Der Einfluss einer verbesserten Suchfunktion auf die Ergebnisse der Studie sind unklar und lohnenswert zu erforschen.

Des Weiteren könnte die Ansicht der Informationen verbessert werden, in dem das Pop-up Fenster verschiebbar und in der Größe veränderbar wäre. Außerdem ist eine bessere Formatierung des Textes im Pop-up Fenster ebenfalls ein guter Forschungsaspekt. Das bedeutet, dass z.B. der Key der Key-Value Paare in dickerer Schrift besser hervorgehoben werden könnte, so dass die Bezeichnung der Information auf den ersten Blick besser erkennbar ist.

Das Design und die Funktionen der Diskussionen in der Seitenleiste sind im Prototypen simpel gehalten. Viele mögliche Ideen wurden nicht implementiert, da sie aus dem Rahmen der Bachelorarbeit fallen. Dazu gehören z.B. ein „Gefällt-mir“ Button, eine Sortierung nach unterschiedlichen Kategorien (z.B. die neuesten oder die beliebtesten Kommentare), die Anzeige, wann ein Kommentar verfasst wurde, ein persönliches Profilbild usw..



# Anhang A

## Ein Anhang

Im Anhang befinden sich folgende Informationen:

1. Pseudocodes des rekursiven Algorithmus in Abschnitt 4.4.3
2. Syntax der Ontologie als JSON-Objekt aus Abschnitt 5.2
3. Fragen und Metriken des GQM aus Abschnitt 6.2
4. In der Studie gestellte Aufgaben aus Abschnitt 6.3.4

---

**Algorithm 1** Füge alle Annotationen zum Zeitpunkt „atSecond“ zum Graphen hinzu

---

```
function CALCULATEALLVISIBLENODES(allNodes, atSecond)

  // Iteration über alle Knoten
  for each currentNode  $\in$  allNodes do

    // Iteration über jeden Sichtbarkeitszeitraum des Knoten
    for each period  $\in$  currentNode.periods do

      // Ist der Knoten zum Zeitpunkt „atSecond“ festgesetzt?
      if period.from  $\leq$  atSecond and
        period.to  $\geq$  atSecond then

        // Falls ja, mache den Knoten sichtbar
        // und mache all seine Kinder mit Tiefensuche sichtbar
        ADDNODETOGRAPH(currentNode)
        ADDALLCHILDRENOF(currentNode)
      end if
    end for
  end for
end function
```

---

---

**Algorithm 2** Zeigt rekursiv alle Kindknoten von einem Elternknoten

---

```
function ADDALLCHILDRENOF(parent)

    // Hole alle ausgehenden Kanten des Elternknoten
    outcomingLinksOfParent ← GETOUTCOMINGLINKSOFF(parent)

    // Iteration ueber alle ausgehenden Kanten des Elternknoten
    for each parentLink ∈ outcomingLinksOfParent do

        // Überspringe bereits sichtbare Kanten und Knoten
        // um eine Endlosschleife (Deadlock) bei Zirkeln zu verhindern
        if parentLink.alreadyVisible = true then
            continue
        end if

        // Mache gefundene Kante und den Kindknoten sichtbar
        ADDLINKTOGRAPH(parentLink)
        parentRelation.alreadyVisible ← true
        child ← GETNODE(parentRelation.target)
        ADDNODETOGRAPH(child)

        // Mache rekursiv alle Kinder des gefundenen Kindes sichtbar
        SHOWALLCHILDRENOF(child)
    end for
end function
```

---

Objekt	Objekt : Datentyp	Erläuterung
Ontologie	<pre>{   "classes": String[ ],   "individuals": Individuum[ ],   "relations": Relation[ ] }</pre>	In der Ontologie werden drei wichtige Bestandteile aufgelistet. Jedes dieser Bestandteile kann mehrere Elemente enthalten.
Individuum	<pre>{   "id": String,   "label": String,   "validity": Validitaet[ ],   "classes": String[ ],   "attributes": Attribut[ ],   "discussions": Diskussion[ ] }</pre>	Die id des Individuums muss eindeutig sein innerhalb der ganzen Ontologie und darf keine Leerzeichen enthalten. Das label ist der Text, der im Prototypen angezeigt wird.
Validitaet	<pre>{   "from": int,   "to": int,   "x": float,   "y": float }</pre>	from und to geben in Sekunden die Zeitspanne an, in der ein Individuum als festgesetzte Annotation im Videobild sichtbar ist. x und y sind dabei im Wertebereich 0 bis 1 und geben an, an welcher Stelle die Annotation sich relativ zu der Breite und Höhe des Bildes befindet.
Attribut	<pre>{   "key": String,   "value": String }</pre>	In Key steht der Bezeichner der Information und in Value der Wert.
Diskussion	<pre>{   "description": String,   "author": String,   "comments": Kommentar[ ] }</pre>	Die Diskussionen eröffnen ein neues Thema, welches von einem Autor verfasst wird. Auf dieses Thema können andere mit Kommentaren antworten.

Kommentar	<pre>{   "content": String,   "author": String }</pre>	Kommentare bestehen aus dem Textinhalt und dem Autor, der den Kommentar verfasst hat.
Relation	<pre>{   "source": String,   "target": String }</pre>	In source und target stehen die ids der Anfangs- und der Endknoten um die Verbindung und die Richtung des Pfeils anzugeben. Es gibt keine Beschriftung.

Goal	Questions	Metrics
1.1	Ist das Programm intuitiv bedienbar?	Frage(n) mit 5-Point-Likert-Skala zur subjektiven Erfahrung
1.2	Ist der Aufbau der grafischen Oberfläche intuitiv?	Frage(n) mit 5-Point-Likert-Skala zur subjektiven Erfahrung
2.1	Werden mit dem ontologiebasierten Design mehr Aufgaben richtig gelöst?	Anzahl richtig gelöster Aufgaben
2.2	Benötigen die Nutzer im ontologiebasierten Design weniger Aufwand um die Aufgaben richtig zu lösen?	Effizienz = Dauer zum Lösen der Aufgaben / Anzahl richtig gelöster Aufgaben
2.3	Benötigen die Nutzer im ontologiebasierten Design weniger Zeit zum Lösen der Aufgaben? Ist das ontologiebasierte Design intuitiv verständlich? Fördert das ontologiebasierte Design eine bessere Übersicht über die Informationen? Welche Rolle spielt die Suchleiste in den Designs? Welche Rolle spielt der Filter im ontologiebasierten Design?	Dauer zum Lösen der Aufgaben, Anzahl der Nutzung der Features zum Lösen der Aufgaben und Frage(n) mit 5-Point-Likert-Skala zur subjektiven Erfahrung

Tabelle A.3: GQM - Fragen und Metriken

Durchführungsteil	Frage	Fragengruppe
1	Welchen Energieverbrauch hat der Kühlschrank?	Information
	In welchen Farben gibt es den Kühlschrank?	Diskussion
	Die Kamera im Kühlschrank hat eine Lichtstärke. Wie wird diese berechnet?	Diskussion
	In der Cloud werden Informationen zu Rezepten gespeichert. Welche Informationen sind es genau?	Information
2	Wie hoch ist der Kühlschrank?	Information
	Wer ist Chef der iCool Firma?	Diskussion
	Die App gibt eine Benachrichtigung, wenn das Mindesthaltbarkeitsdatum verfällt. Wie viele Tage im Voraus tut sie das?	Diskussion
	Die Datenbank in der Cloud ist verschlüsselt. Mit welchem Algorithmus?	Information
	Der iCool Kühlschrank ist mit zwei technischen Geräten ausgestattet, welche in anderen regulären Standardkühlgeschränken nicht verbaut sind. Welche Geräte sind das?	Verständnis

Tabelle A.4: Die gestellten Aufgaben in der Studie



# Literaturverzeichnis

- [1] S. M. Amiri. Konzeptionierung eines tools zur herstellung gemeinsamen verständnisses durch asynchrone betrachtung von vision videos. 2022.
- [2] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309, 2011.
- [3] O. Brill, K. Schneider, and E. Knauss. Videos vs. use cases: Can videos capture more requirements under time pressure? In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 30–44. Springer, 2010.
- [4] G. Broll, H. Hussmann, E. Rukzio, and R. Wimmer. Using video clips to support requirements elicitation in focus groups-an experience report. In *SE 2007 Workshop on Multimedia Requirements Engineering*, 2007.
- [5] A. Bruun and J. Stage. An empirical study of the effects of three think-aloud protocols on identification of usability problems. In *IFIP Conference on Human-Computer Interaction*, pages 159–176. Springer, 2015.
- [6] E. Börger, B. Hörger, D. Parnas, and D. Rombach. Requirements capture, documentation, and validation. In *Dagstuhl Seminar*, number 99241. Citeseer, 1999.
- [7] J. Cabrera. *Modular design frameworks: A projects-based guide for UI/UX designers*. Springer, 2017.
- [8] V. R. B. G. Caldiera and H. D. Rombach. The goal question metric approach. *Encyclopedia of software engineering*, pages 528–532, 1994.
- [9] C. Calero, F. Ruiz, and M. Piattini. *Ontologies for software engineering and software technology*. Springer Science & Business Media, 2006.
- [10] M. Dudáš, S. Lohmann, V. Svátek, and D. Pavlov. Ontology visualization methods and tools: a survey of the state of the art. *The Knowledge Engineering Review*, 33, 2018.

- [11] D. ENISO. 9241-11: Ergonomische anforderungen für bürotätigkeiten mit bildschirmgeräten—teil 11: Anforderungen an die gebrauchstauglichkeit; leitsätze (iso 9241-11: 1998). *Berlin: Beuth*, 1998.
- [12] L. Euler. Leonhard euler and the königsberg bridges. *Scientific American*, 189(1):66–72, 1953.
- [13] C. Fluit, M. Sabou, and F. v. Harmelen. Supporting user tasks through visualisation of light-weight ontologies. In *Handbook on ontologies*, pages 415–432. Springer, 2004.
- [14] T. M. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software: Practice and experience*, 21(11):1129–1164, 1991.
- [15] W. O. Galitz. *The essential guide to user interface design: an introduction to GUI design principles and techniques*. John Wiley & Sons, 2007.
- [16] F. Ghassemi Toosi, N. S. Nikolov, and M. Eaton. Simulated annealing as a pre-processing step for force-directed graph drawing. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, pages 997–1000, 2016.
- [17] M. Glinz. Improving the quality of requirements with scenarios. In *Proceedings of the second world congress on software quality*, volume 9, pages 55–60, 2000.
- [18] M. Glinz and S. A. Fricker. On shared understanding in software engineering: an essay. *Computer Science-Research and Development*, 30(3):363–376, 2015.
- [19] T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5-6):907–928, 1995.
- [20] M. Herrmann. Annotation von anforderungsbezogenen videos auf basis von ontologien. 2017.
- [21] J. Johnson. *Designing with the mind in mind: simple guide to understanding user interface design guidelines*. Morgan Kaufmann, 2020.
- [22] O. Karras. Communicating stakeholders’ needs-vision videos to disclose, discuss, and align mental models for shared understanding. *IEEE Softw. Blog*, 2019.

- [23] O. Karras, S. Kiesling, and K. Schneider. Supporting requirements elicitation by tool-supported video analysis. In *2016 IEEE 24th International Requirements Engineering Conference (RE)*, pages 146–155. IEEE, 2016.
- [24] S. Lohmann, S. Negru, F. Haag, and T. Ertl. Vowl 2: User-oriented visualization of ontologies. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 266–281. Springer, 2014.
- [25] B. Meixner and H. Kosch. Interactive non-linear video: definition and xml structure. In *Proceedings of the 2012 ACM symposium on Document engineering*, pages 49–58, 2012.
- [26] B. Meixner, B. Siegel, G. Hölbling, F. Lehner, and H. Kosch. Siva suite: authoring system and player for interactive non-linear videos. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1563–1566, 2010.
- [27] L. Nagel and O. Karras. Keep your stakeholders engaged: Interactive vision videos in requirements engineering. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 51–57. IEEE, 2021.
- [28] F. Paetsch, A. Eberlein, and F. Maurer. Requirements engineering and agile software development. In *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003.*, pages 308–313. IEEE, 2003.
- [29] F. Schneider and B. Berenbach. A literature survey on international standards for systems requirements engineering. *Procedia Computer Science*, 16:796–805, 2013.
- [30] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization*, pages 364–371. Elsevier, 2003.
- [31] D. Zhang, L. Zhou, R. O. Briggs, and J. F. Nunamaker Jr. Instructional video in e-learning: Assessing the impact of interactive video on learning effectiveness. *Information & management*, 43(1):15–27, 2006.

