

UNIVERSITÄT LEIPZIG

Institut für Medizinische Informatik, Statistik und Epidemiologie (IMISE)

# Automatische Erstellung von Quizfragen aus einer Ontologie von Krankenhausinformationssystemen

Besondere Lernleistung

Leipzig, Dezember 2020

vorgelegt von:

Max Niclas Wächtler

geb. am: 17.05.2003

Betreuer:

Konrad Höffner

## **Zusammenfassung**

Krankenhausinformationssysteme sind Informationssysteme, die sich in medizinischen Umgebungen mit der Verarbeitung von Daten, Informationen und Wissen beschäftigen. Informationssysteme bestehen aus Hardware, Software als auch Anwendern. Kompetenz in der Nutzung von Krankenhausinformationssystemen ist wichtig, um in Krankenhäusern beispielsweise die Zusammenarbeit mit Patienten effektiv und zeitsparend zu gestalten. Jedoch müssen Anwender diese Kompetenz erst einmal durch Quellen wie Lehrbücher erwerben, welche jedoch verschiedene Sichten auf das Themengebiet aufweisen und keinen schnellen Überblick über bestimmte Themen bieten können. Um dieses Problem anzugehen, habe ich mir als Ziel dieser besonderen Lernleistung gesetzt, ein Programm zu entwickeln, welches diese Informationen aus einer Ontologie, also eine Menge von Begrifflichkeiten mit Beziehungen untereinander, über eine Schnittstelle entnimmt, diese nutzerfreundlich in Form von zufällig generierten Fragen aufarbeitet und dem Nutzer ausgibt. Diese sollen eine grammatikalisch korrekte Form aufweisen und nebst der richtigen Antwort auch falsche Antworten enthalten, die trotzdem logisch wirken sollen, anstatt ein Zufallsverfahren zur Auswahl zu nutzen.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>5</b>
1.1	Gegenstand und Motivation . . . . .	5
1.1.1	Gegenstand . . . . .	5
1.1.2	Problematik . . . . .	7
1.1.3	Motivation . . . . .	8
1.2	Zielsetzung . . . . .	9
1.3	Vorgehensweise . . . . .	9
<b>2</b>	<b>Grundlagen</b>	<b>11</b>
2.1	Semantic Web . . . . .	11
2.1.1	WWW . . . . .	11
2.1.2	Linked Data . . . . .	11
2.1.3	RDF . . . . .	12
2.1.4	SPARQL . . . . .	12
2.1.5	Ontologien . . . . .	14
2.1.6	Künstliche Intelligenz . . . . .	14
2.2	Informationssysteme . . . . .	14
2.2.1	Krankenhausinformationssystem . . . . .	15
2.2.2	Grundbegriffe . . . . .	15
2.3	SNIK-Projekt . . . . .	15
2.3.1	JavaScript . . . . .	16
2.4	Multiple-Choice-Quiz . . . . .	17
2.4.1	Distraktoren . . . . .	17
<b>3</b>	<b>Lösungsansatz</b>	<b>19</b>
3.1	Lösungsansatz zu Aufgabe A1.1 . . . . .	19
3.2	Lösungsansatz zu Aufgabe A2.1 . . . . .	20
3.3	Lösungsansatz zu Aufgabe A2.2 . . . . .	20

3.4	Lösungsansatz zu Aufgabe A2.3 . . . . .	20
<b>4</b>	<b>Ausführung der Lösung</b>	<b>22</b>
4.1	Lösung für Aufgabe A1.1 . . . . .	22
4.1.1	Untersuchen der bestehenden Queries . . . . .	22
<b>5</b>	<b>Ergebnis</b>	<b>27</b>
<b>6</b>	<b>Diskussion und Ausblick TODO: 1–2 Seiten</b>	<b>28</b>

# Begriffs- und Abkürzungsverzeichnis

**SPARQL** SPARQL Protocol and RDF Query Language

**W3C** World Wide Web Consortium

**KI** Künstliche Intelligenz

**RDF** Resource Description Framework

**HTML** Hypertext Markup Language

**XML** Extensible Markup Language

**URIs** Universal Resource Identifier

**SNIK** Semantisches Netz des Informationsmanagements im Krankenhaus

**WWW** World Wide Web

**W3C** World Wide Web Consortium

**HTTP** Hypertext Transfer Protocol

**MIG** Management von Informationssystemen im Gesundheitswesen

# Kapitel 1

## Einleitung

### 1.1 Gegenstand und Motivation

#### 1.1.1 Gegenstand

Das Semantische Netz des Informationsmanagements im Krankenhaus (SNIK) ist ein abgeschlossenes Projekt, welches Begriffe des Informationsmanagements in dem Teilgebiet der medizinischen Informatik sowie deren Beziehungen untereinander beschreibt. Es ist in der Lage, dieses Wissen in einer Ontologie darzustellen und diese sowohl maschinen- als auch menschenlesbar auszugeben. Dabei nutzt es ein dem Semantic Web Stack ähnliches Modell (Abb. 2.2), welches Anwendungen wie dem SNIK Graph ermöglicht, Informationen abzufragen und diese Endnutzern in aufbereiteter Form zur Verfügung zu stellen. Das Projekt konzentriert sich hauptsächlich auf Wissen der medizinischen Lehre. Dieses wird für Krankenhausinformationssysteme zur Verfügung gestellt, um die Arbeit für medizinisches Personal zu erleichtern. Speziell hat der durch Prof. Dr. Alfred Winter geleitete Teilbereich Management von Informationssystemen im Gesundheitswesen (MIG) die Zielsetzung, durch Entwicklung von Methoden und Werkzeugen für das Informationsmanagement zu einer besseren Gesundheitsversorgung beizutragen.

Um nicht mit Ontologien versierten Nutzern eine Anwendung, die die Daten des Projektes nutzt, darzustellen, wurde unter anderem auch ein einfaches Multiple-Choice-Quiz auf Basis eines öffentlich verfügbaren Templates programmiert. Diese nutzt die Ontologie hinter SNIK, um Anwendern einfache, automatisch generierte Fragen zu stellen. Diese Fragen werden mithilfe der typisierten, gerichteten Verbindungen zwischen einzelnen Begriffen erzeugt, wobei das Quiz

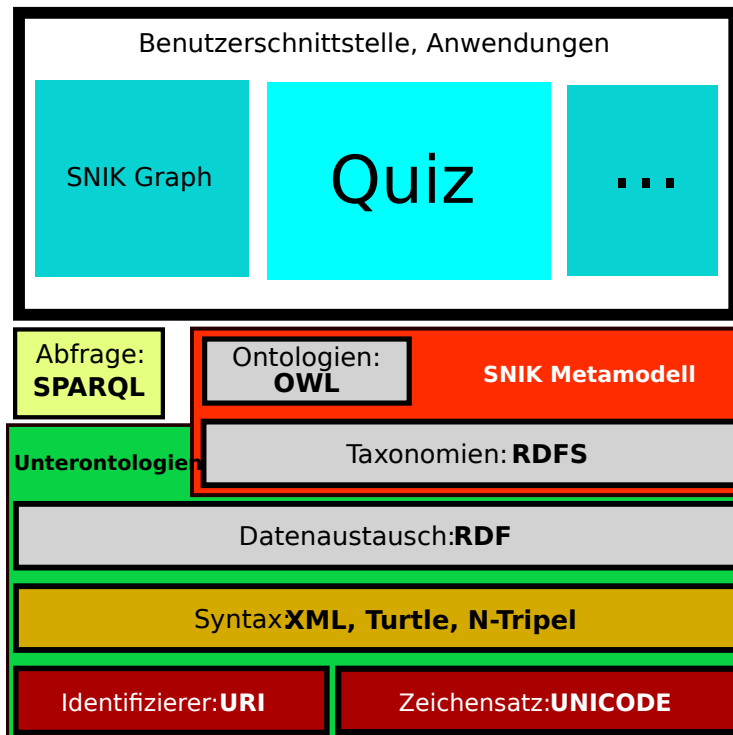


Abbildung 1.1: Das semantische Modell von SNIK.

aus der SNIK-Ontologie drei Haupttypen von Verbindungen nutzt:

- 1. Objekt A definiert Objekt B (A defines B),
- 2. Objekt A beinhaltet Objekt B (A contains B) und
- 3. Objekt A unterstützt Objekt B (A supports B).

Dadurch werden für jeden Verbindungstyp unterschiedliche Fragen generiert. Für Definitionen generiert sich der Fragetitel hierbei zum Beispiel nach dem folgenden Muster:

„What is defined by A?“

Unabhängig vom Fragetyp werden jeder Frage noch verschiedene Antworten zugeteilt. Dazu zählt einmal die richtige Antwort, neben der aber auch drei andere, falsche Antworten durch einfache SPARQL-Queries generiert werden. TODO:

- In welcher Welt/Domäne oder welchem Arbeitsbereich/-gebiet bewegen wir uns im Rahmen der Seminararbeit/der ausgewählten Papers?
- Worum geht es eigentlich?

Aus den Papers bzw. dem Antrag des Forschungsprojektes entnehmen.

### 1.1.2 Problematik

TODO: generell im gesamten text: jeden satz auf eine zeile Die Nutzer sollen möglichst mit logischen, aber nicht zu einfachen Fragen herausgefordert werden. Um mithilfe des Multiple-Choice-Quiz ihren Wissensstand zu verbessern, sollte das Quizprogramm bestimmte Vorgaben erfüllen und diese auch in der Qualität der Fragen widerspiegeln. Das stellt sich als relativ schwer heraus, da die Implementierung von natürlichsprachlicher Grammatik sich seit jeher als ein Problem in der Informatik herausstellt.

**Korrektheit** Für den Anwender soll die Frage gut lesbar und in seiner Grammatik verständlich sein. Da die Quizfragen in einem bestimmten Zeitfenster bearbeitet werden sollen, ist es wichtig, dass sie schnell vom Nutzer verstanden werden. Da der Hauptzweck die Selbstüberprüfung ist, werden dadurch fehlende Korrektheit gegenüber dem User die Ergebnisse verfälscht, was zu einer falschen Selbsteinschätzung führt.

**Übersichtlichkeit** Eine Frage muss für den Nutzer übersichtlich gestaltet sein. Sie soll schnell durchgelesen sein, damit er den metaphorischen „Faden“ nicht verliert und Fragen nicht erneut lesen muss. Um dies zu garantieren, muss die Anzahl der Zeichen im Fragetitel eine bestimmte Zahl nicht übersteigen. Dies kann zum Beispiel bei langen Definitionen (Fragetyp 1) ein Problem sein.

**Qualität** Neben den anderen beiden Bedingungen muss bei Fragen auch die Qualität und das Niveau der Frage sowie des Quiz allgemein hoch gehalten werden. Doch das ist im bereits existenten Quiz an vielen Stellen nur teilweise vorhanden, was sich in verschiedenen Aspekten widerspiegelt. Darunter fallen zum Beispiel die Erwähnung der richtigen Antwort im Fragetitel bei z.B. Definitionen oder in einfacheren Punkten wie z.B. dem Doppeln von Fragen. Dadurch wird der Nutzer nicht genug herausgefordert, was dazu führt, dass er die Fragen nicht mithilfe seines Fachwissens, sondern einfach durch logisches Denken beantworten kann.

**Heterogenität** Das Quiz sollte für den Nutzer möglichst einen großen Teil der Möglichkeiten der SNIK-Ontologie anschaulich machen. Da jedoch das



existierende Quiz nur zwei der 18 Typen von Verbindungen zwischen Objekten nutzt, die in der SNIK-Ontologie vorkommen, variieren die Fragetypen nur wenig, was auf Dauer für Nutzer langweilig oder anstrengend wird. Außerdem wird dadurch ein großer Teil des verfügbaren Wissens nicht abgefragt,

TODO:

- Worin bestehen die Probleme?
- Warum ist die geschilderte vorliegende Situation problematisch?
- Für wen ist sie problematisch?

Aus den Papers bzw. dem Antrag des Forschungsprojektes entnehmen.

TODO: Hier ist auf die generelle Problemlage einzugehen, die dem Aufsatz im Wissenschaftsfeld zu Grunde liegt.

Bsp.: Daten aus der Krankenversorgung stehen aus rechtlichen und technischen Gründen nicht für die Forschung zur Verfügung, sodass klinische Forscher eigene Datenerhebungen planen müssen.

### 1.1.3 Motivation

Das Ziel des SNIK-Projektes ist es, ein theoretisch und empirisch begründetes, sowie erprobtes Semantisches Netz des Informationsmanagements im Krankenhaus (SNIK), das die Konzepte des Informationsmanagements beschreibt und verbindet, zu kreieren. Um Medizinstudenten und anderen Mitarbeitern aus dem medizinischen Umfeld eine beispielhafte Nutzungsmöglichkeit für diese Ontologie zu bieten und ihnen einen einfachen Zugriff neben dem SNIK-Graph auf das Themengebiet zu ermöglichen, kann ein Multiple-Choice-Quiz einen einfachen Einblick auf die Einsatzmöglichkeiten von einer Ontologie aus Krankenhausssystemen bereitstellen und Interessenten aus nichtinformatischen Teilgebieten eine Vorstellung ermöglichen.

TODO:

- Warum lohnt es sich, die genannten Probleme zu lösen?
- Wer wird welchen Nutzen von dieser Abschlussarbeit haben?
- Warum sind die Papers wichtig?

Aus den Papers bzw. dem Antrag des Forschungsprojektes entnehmen. **TODO:** Worin liegt der zusätzliche Nutzen, die beiden Ansätze zu vergleichen? Worin liegt der Nutzen, den Ansatz des Papers auf die Problematik des Forschungsprojektes anzuwenden?

Bsp.: Gelänge es, Daten aus der Krankenversorgung vollständig zu anonymisieren, könnten sie für beliebige Forschungsvorhaben genutzt werden.

## 1.2 Zielsetzung

Das Ziel meiner besonderen Lernleistung ist es, automatisch Multiple-Choice-Quizfragen auf Basis der SNIK-Ontologie zu generieren. Dazu wird ein existierender Prototyp analysiert und hinsichtlich der oben genannten Qualitätsmaße optimiert.

- Ziel Z1: Analysieren des vorhandenen Prototyps
- Ziel Z2: Optimieren des vorhandenen Prototyps

## 1.3 Vorgehensweise

**TODO:** Stellen Sie Ihre Methode bzw. Herangehensweise dar. Was ist Ihre konkrete Aufgabenstellung im Rahmen der Seminararbeit? Das könnte z.B. sein:

- Vorstellen verschiedener Lösungsansätze, Vergleich und Bewertung der Lösungsansätze, Literaturrecherche nach weiteren vergleichbaren Papers.
- Bewertung eines Lösungsansatzes hinsichtlich einer konkreten Forschungssituation. Literaturrecherche nach weiteren vergleichbaren Papers.

**TODO:** Beschreiben Sie Ihre Vorgehensweise in einzelnen Schritten und stellen Sie die Reihenfolge der Abarbeitung dar. Auf diese Punkte sollten Sie in der Diskussion wieder eingehen. **TODO:**

1. Schritt

2. Schritt

TODO: Vermeiden Sie Formulierungen wie „Es ist nicht bekannt, ob...“ oder „Es existiert kein...“. Solche Formulierungen kehren in der Regel einfach das bereits angedachte Lösungsmodell um und postulieren das Fehlen der angedachten Lösung einfach als Problem. ...

Schritt n

Das ist ähnlich, wie wenn es in der Werbung hieße „Wenn Sie das Problem haben, dass Ihnen Aspirin fehlt, dann kaufen Sie doch Aspirin.“ Sinnvoller ist diese Aussage: „Wenn Sie das Problem haben, dass Ihnen der Kopf weh tut, dann kaufen Sie doch Aspirin.“ Es ist also bei der Problembeschreibung erforderlich, sich in die Lage dessen zu versetzen, den man mit der angedachten Lösung beglücken möchte. Sein Problem ist zu ermitteln und so zu formulieren, er/sie das Problem wiedererkennt und dadurch geneigt ist, sich für die Lösung des Problems zu interessieren.

- Aufgabe A1.1 zu Ziel Z1: Analyse der Strategien zur Generierung der Stämme und Distraktoren
- Aufgabe A2.1 zu Ziel Z2: Entwickeln neuer Strategien zur Generierung der Stämme
- Aufgabe A2.2 zum Ziel Z2: Entwickeln neuer Strategien zur Generierung von Distraktoren
- Aufgabe A2.3 zum Ziel Z2: Aufstellen von Regeln zur Nutzerfreundlichkeit der Fragen

TODO: (2–3 Seiten)

# Kapitel 2

## Grundlagen

### 2.1 Semantic Web

Das Semantic Web (oder auch: semantisches Netz) stellt eine Methode dar, um maschinenlesbares Wissen über das World Wide Web (WWW) in Form von HTML-Dokumenten zu verbreiten. Anders als bei der klassischen Website aber wird hier nicht nur primär Wert auf die Lesbarkeit durch einen Menschen, sondern auch auf die Maschinenlesbarkeit Wert gelegt, indem die Seite nebst dem normalen HTML-Code auch Informationen, die durch Computer gelesen werden können, hinterlegt. Diese sind durch Technologien des semantischen Webs ( RDF, SPARQL, ... ) einles- und verarbeitbar.

#### 2.1.1 WWW

Das WWW bildet ein verknüpftes Kommunikationsmodell zwischen allen Ressourcen und Nutzern des Internets, die das Hypertext Transfer Protocol (HTTP) nutzen. Es wurde vom Direktor der World Wide Web Consortium (W3C) Tim Berners-Lee 1991 entwickelt und ermöglicht den Zugriff auf sowie den Datenaustausch mit HTML-Dokumenten. Es definiert also das, was die Allgemeinheit als „das Web“ bezeichnet.

#### 2.1.2 Linked Data

Der Begriff „Linked Data“ bezieht sich auf eine Reihe bewährter Methoden zum Veröffentlichen und Verbinden strukturierter Daten im Web. Es verknüpft Dateien aus dem semantischen Web untereinander, sodass sie durch semantische

Abfragen benutzbar werden. Linked Data basiert auf standardisierten Modellen für den Datenaustausch im Web, beispielsweise HTML, um auch wie beim semantischen Prinzip maschinenlesbare Daten zu generieren. Zielsetzung von Linked Data ist es, das Internet zu einer globalen, computerverarbeitbaren Datenbank weiterzuentwickeln und so den Zugriff für technische Endgeräte weiter zu vereinfachen.

### 2.1.3 RDF

Resource Description Framework (RDF) ist ein weiteres Standardmodell für den Datenaustausch im Web, welches in den 90er Jahren des letzten Jahrtausends seine Anfänge findet. Im Gegensatz zu Hypertext Markup Language (HTML) und Extensible Markup Language (XML) ist es in der Lage, enthaltene Informationen zu kombinieren und weiter zu verarbeiten, anstatt diese nur korrekt anzuzeigen. Daher ist RDF auch oft als grundlegendes Repräsentationsformat für die Entwicklung im Semantic Web angesehen. Es erweitert die Verknüpfungsstruktur von diesem, um Universal Resource Identifier (URIs) einzubinden. Solche URIs werden im semantischen Netz dazu genutzt, Beziehungen zwischen Dingen beziehungsweise zwischen den beiden Enden der Verknüpfung herzustellen, um Dateien zu verknüpfen, verfügbar zu machen und für Anwendungen freizugeben. Dadurch können Daten zusammengeführt werden, die sich eigentlich im Schema grundlegend unterscheiden, ohne dass alle Datenkonsumenten geändert werden müssen. Diese Zusammenführung geschieht in einem Schichtenmodell: Aussagen werden als Tripel modelliert und bilden eine Aussage, die aus Subjekt, Prädikat und Objekt zusammengefügt werden kann. Man kann also einen Tripel mit einem einfachen Satz vergleichen: als Beispiel hier einmal der Teilsatz *Google produziert Prozessoren*. Hier stellt *Google* das Subjekt dar, *produziert* ist das Prädikat und *Prozessoren* sind das Objekt. Hat man eine Menge von Tripeln, bilden diese ein semantisches Netz, was tabellarisch übersichtlich dargestellt werden kann, siehe Tabelle 2.1.

### 2.1.4 SPARQL

SPARQL Protocol and RDF Query Language (SPARQL) ist ein Standard für die Abfrage von in RDF spezialisierten Informationen sowie für die Darstellung der zugehörigen Resultate, siehe Abb. 2.1. Es basiert auf einfachen RDF-Anfragen in Form von Graphmustern, also einer Menge von Tripeln,

Subjekt	Prädikat	Objekt
New York	liegt in	USA
USA	liegt in	Nordamerika
Nordamerika	liegt in	Amerika
Südamerika	liegt in	Amerika

Tabelle 2.1: Beispiel für RDF-Tripel.

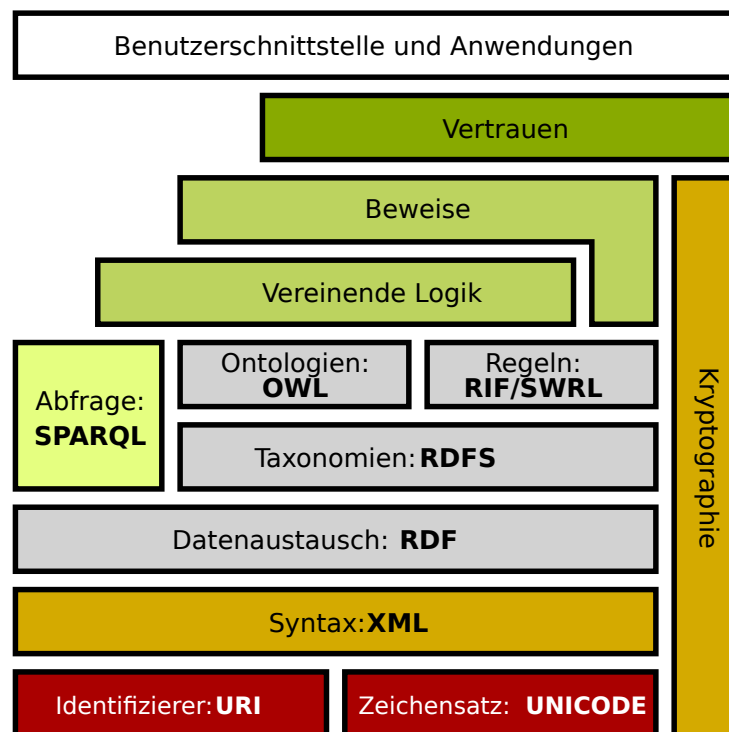


Abbildung 2.1: Der Semantic Web Stack.

enthält jedoch auch erweiterte Funktionen für die Konstruktion komplexerer Anfragemuster, die Verwendung zusätzlich hinzufügbare Filterbedingungen und für die Formatierung der schlussendlichen Ausgabe.

Man kann also SPARQL als eine faktische Abfragesprache des semantischen Webs bezeichnen.

### 2.1.5 Ontologien

*An ontology is an explicit specification of a conceptualization.*

—Thomas R. GruberTODO: [quelle zitieren](#)

Ontologien sind (im informatischen Kontext) meist sprachlich gefasste und geordnete Darstellungen einer Menge von Begrifflichkeiten mit festen Beziehungen untereinander. Diese Beziehungen befassen sich stets auf einen bestimmten Gegenstandsbereich und werden dazu genutzt, Wissen in digitalisierter Form zwischen Anwendungen und Diensten auszutauschen. Dabei müssen bestimmte Interferenz- und Integritätsregeln eingehalten werden, also Regeln zu Schlussfolgerungen sowie zu der Gewährleistung ihrer Gültigkeit. Ontologien erfreuen sich seit der Idee des semantischen Webs einem stets wachsenden Bekanntheitsgrad, was dazu führt, dass sie als Teil der Wissensrepräsentation im Teilgebiet KI einen großen Einfluss haben. Dabei beschreibt die Idee des Semantic Web eine Erweiterung des vorhandenen Netzes, um Daten zwischen Rechnern einfacher austausch- und verwertbar zu machen und sie somit besser zu explizieren, anstatt sie unkonstruiert stehen zu lassen. Zum Sinne dieser Realisierung dienen Standards zur Veröffentlichung und Nutzung maschinenlesbarer Daten – insbesondere RDF.

### 2.1.6 Künstliche Intelligenz

Künstliche Intelligenz (KI) ist ein Teilgebiet der Informatik, welches sich mit der Automatisierung intelligenten Verhaltens befasst. Es spiegelt eine Möglichkeit wieder, bestimmte Entscheidungsstrukturen des Menschen nachzubilden und es somit Computern zu ermöglichen, komplexe Probleme selbstständig zu bearbeiten und zu lösen. Um diese Informationen weiter zu nutzen, können Methoden des Semantic Web genutzt werden, um sie als Tripel einfach maschinenlesbar darzustellen und für andere Computer verfügbar zu machen.

## 2.2 Informationssysteme

Ein Informationssystem ist ein System, welches durch die Bildung logischer Zusammenhänge eine Deckung der Informationsnachfrage zur Aufgabe hat. Es produziert, beschafft, verteilt und verarbeitet Daten durch eine Zusammenarbeit zwischen Mensch und Technik durch Aufgaben.

### 2.2.1 Krankenhausinformationssystem

Krankenhausinformationssysteme beschreiben die Gesamtheit aller Informationssysteme zur Produktion, Beschaffung, Verteilung und Verarbeitung von medizinischen und administrativen Daten im Krankenhaus. Dazu gehören verschiedene Formen der Datenbereitstellung sowie auch konventionelle Methoden der papierbasierten Dokumentation und der sprachlichen Kommunikation. Das grundsätzliche Ziel eines Krankenhausinformationssystems ist, die Kommunikation zwischen Mitarbeitern zu verbessern und den Ablauf in Krankenhäusern zu steuern, indem Mitarbeitern gezielt Zugriff auf für ihn relevantes Wissen aus der ihm zugeteilten Benutzerrolle gegeben wird, zum Beispiel über den gerade zu behandelnden Patienten.

### 2.2.2 Grundbegriffe

**Information** Information ist die Kenntnis über bestimmte Sachverhalte oder Vorgänge, im engeren Sinne über einzelne, konkrete, benennbare Objekte der realen und der virtuellen Welt.

**Wissen** Wissen ist die Kenntnis über den in einem Fachgebiet zu gegebener Zeit gegebenen Konsens, vor allem bezogen auf eine gültige Terminologie, erlaubte Interpretationen, bestehender Zusammenhänge und Gesetzmäßigkeiten sowie empfehlenswerter Methoden und Handlungen. Wissen ist also auch Information, aber in dem Fall nicht auf einzelne Objekte, sondern eine Menge von Objekten und deren Beziehungen untereinander bezogen, vgl. ?.

**Daten** Daten sind Gebilde aus Zeichen oder kontinuierliche Funktionen, die durch bekannte oder unterstellte Beziehungen Informationen darstellen können. Sie stellen die Grundlage oder das Ergebnis eines Verarbeitungsschrittes dar.

## 2.3 SNIK-Projekt

Das SNIK ist ein vollendetes Projekt, welches Begriffe des Informationsmanagements sowie deren Beziehungen untereinander beschreibt. Es ist in der Lage, diese Menge an Informationen in einer Ontologie darzustellen und diese sowie maschinen- als auch menschenlesbar auszugeben. Dabei nutzt es ein dem Semantic Web Stack ähnliches Modell (Abb. 2.2), welches Anwendungen wie



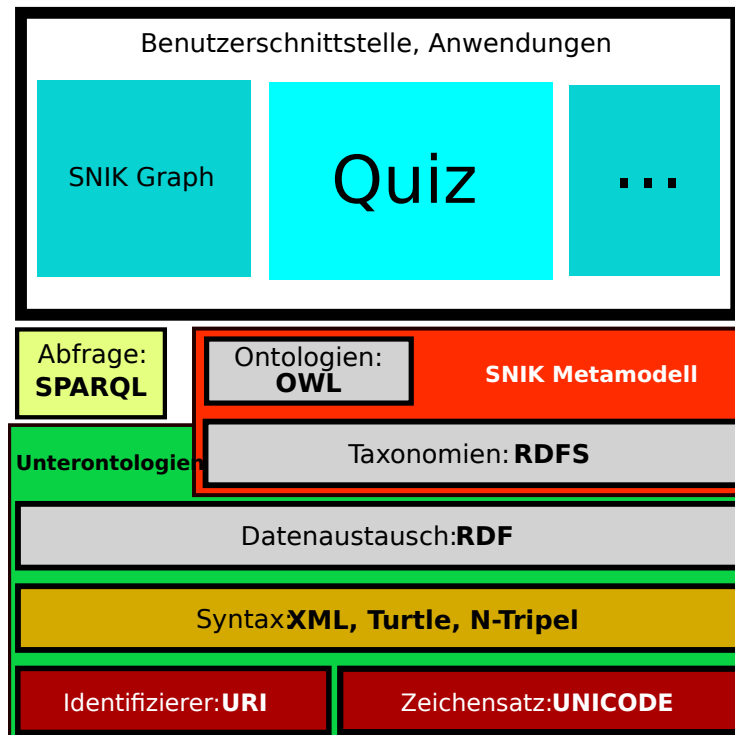


Abbildung 2.2: Das semantische Modell von SNIK.

dem SNIK Graph oder einem Multiple-Choice-Quiz ermöglicht, Informationen abzufragen und diese Endnutzern in aufbereiteter Form zur Verfügung zu stellen. Dabei konzentriert sich das SNIK-Projekt hauptsächlich auf die Daten der medizinischen Lehre, um diese für Krankenhausinformationssysteme zur Verfügung zu stellen und die Arbeit für medizinisches Personal zu erleichtern.

### 2.3.1 JavaScript

JavaScript ist eine leistungsstarke und flexible Skriptsprache, die von Entwicklern zunehmend zur Erstellung interaktiver Webanwendungen eingesetzt wird. Die Sprache ist dynamisch, schwach typisiert und verfügt über eine Vielzahl an für Webanwendungen essentiellen Funktionen. Sie lässt sich außerdem hervorragend mit anderen Websprachen wie CSS und HTML implementieren und besitzt die Fähigkeit, mit diesen zu interagieren.

JavaScript wird bei der Quiz-Webanwendung des SNIK-Projekts genutzt, um Fragen und Antworten anzuzeigen und diese dem Nutzer interaktiv zum Beantworten bereitzustellen.

## 2.4 Multiple-Choice-Quiz

Ein Multiple-Choice-Quiz ist eine Ansammlung an Fragen, die allesamt aus einem Stamm (der Frage), einem Schlüssel (der richtigen Antwort) und mehreren Distraktoren (eine Menge von inkorrekten, aber plausibel klingenden Antwortmöglichkeiten), vgl. TODO: [Quelle mit citet zitieren](#)

Ein Beispiel für eine solche Frage wäre hier aufgeführt:

*Q: In einer Hand halte ich ein rotes, ein grünes und ein blaues Blatt Papier. Ich lege nun das rote und das blaue Blatt aus meiner Hand. Welche Farbe hat das Blatt, welches sich immer noch in meiner Hand befindet?*

- Rot
- Grün
- Blau

Hierbei ist die Antwort „Grün“ der Schlüssel. „Rot“ und „Blau“ sind *Distraktoren*, die von dem Schlüssel ablenken sollen.

Der Sinn bei einem Multiple-Choice-Quiz besteht darin, den Befragten für richtige Antworten mit Punkten zu belohnen. Antwortet er falsch, bleibt seine Punktzahl entweder gleich oder Punkte werden abgezogen. Die Anzahl der vergebenen Punkte kann dabei von unterschiedlichen Faktoren abhängig gemacht werden, beispielsweise an der Zeit, die zum Lösen der Frage benötigt wurde.

### 2.4.1 Distraktoren

Distraktoren beschreiben im Kontext eines Quiz falsche Antwortmöglichkeiten, die den Befragten von der richtigen Antwort ablenken sollen. Durch sie soll vermieden werden, dass falschen Antworten durch Logik einfach von der richtigen Antwort unterschieden werden können. Der Sinn von Distraktoren ist es also, für den Befragten plausibel zu erscheinen. Daraus resultierend kann eine schlechte Auswahl von Distraktoren Fragen um einiges einfacher gestalten. Nehmen wir dazu einmal als Beispiel unsere Frage im Abschnitt 2.4: die Distraktoren klingen plausibel, da sie im Stamm der Frage erwähnt werden. Würde man hier allerdings andere Farben zur Auswahl stellen, die nicht im Kontext der Frage stehen und somit direkt ausgeschlossen werden können, zum Beispiel „Gelb“, könnte der Befragte diese logisch ausschließen.

TODO: Mehr zu URIs und Classes mit Attributen erklären (Klasse A hat Attribut Definition und Label bspw.)

# Kapitel 3

## Lösungsansatz

TODO: Der Lösungsansatz ist eine kurze Beschreibung der Arbeitshypothese sowie des Vorgehens zur Lösung der in der Einleitung beschriebenen Probleme. Dieses Kapitel soll dazu dienen, den Lösungsweg der genannten Probleme genauer zu beschreiben. Eine Erläuterung der Ausführung dieser erfolgt im nächsten Kapitel.

### 3.1 Lösungsansatz zu Aufgabe A1.1

Um den bestehenden Prototyp zu verstehen und diesen zu erweitern bzw. zu verbessern, müssen wir dazu die Methodik zur Generierung der Fragen analysieren. Diese Generierung hat die Zielsetzung, menschenlesbare Quizfragen zu erzeugen. Dazu wird als Erstes im existierenden Prototyp eine Abfrage an die Ontologie gesendet, um die nötigen Daten für die Generierung der Frage zu erhalten. Solche Abfragen werden in der SNIK-Ontologie durch SPARQL-Queries (Abschnitt 2.1.4) gehandhabt. Gesendete Queries werden von einem Endpunkt auf dem SNIK-Server angenommen, welcher diese dann als Abfrage an die Ontologie weiterleitet und diese umsetzt. Diese Rückgabewerte werden dann in dem Quiz genutzt und mithilfe von einfachem JavaScript-Code im Browser des Benutzers in einfacher Wortform angezeigt. Um Ressourcen zu schonen, wird jedoch nicht bei jedem Aufruf des Quizzes eine einzelne Frage generiert - das Quiz greift auf eine große Liste aus vorher generierten Fragen zu. Diese sind im JavaScript-Code, welcher durch den Browser angezeigt wird, fest implementiert.

Um den Inhalt dieser Fragen zu erhalten, muss eine Abfrage an den SPARQL-

Endpunkt gesendet werden, der auf die Ontologie zugreift und mithilfe der Abfrage bestimmte Informationen an das Programm ausgibt. Die erhaltenen Daten werden nun von dem Programm spezifisch für den Verbindungstyp verarbeitet, wie in Abschnitt 1.1.1 schon erklärt wurde. Bei der spezifischen Verarbeitung gibt es eine wichtige Unterscheidung: die Methodik der Generierung des Stammes und die Methodik der Generierung der Distraktoren. Während ersteres für jeden Fragetyp unterschiedlich ist, werden Distraktoren stets gleich generiert. Hierzu müssen also im nächsten Abschnitt die Generierung der verschiedenen Fragen und Antworten analysiert werden.

## **3.2 Lösungsansatz zu Aufgabe A2.1**

Um nun diesen vorhandenen Prototypen zu verbessern, müssen zuerst die Strategien der Queries überarbeitet werden, die die generellen Fragen und die richtige Antwort generieren. Hierzu müssen in Abschnitt 1.1.2 beschriebene Probleme gelöst werden. Weiterhin sollen weitere Typen von Abfragen entwickelt werden, die auf den drei bestehenden Verbindungstypen basieren und neue Typen nutzen. Die SNIK-Ontologie bietet eine Vielzahl von Verbindungstypen, welches sich das Quiz zunutze machen kann.

## **3.3 Lösungsansatz zu Aufgabe A2.2**

In dem aktuellen Prototyp wird immer dieselbe Query genutzt, um Distraktoren zu generieren. Um die Entwicklung bestimmter neuer Fragetypen zu ermöglichen, müssen jedoch neue Abfragen generiert werden, die auf die richtige Antwort von Fragen angepasst sind. Das ist beispielsweise bei Fragen notwendig, wo die Definition zu einem Wort gefragt ist oder bei einer Frage, die eine Antwort in Zahlen- oder Prozentform erfordert.

## **3.4 Lösungsansatz zu Aufgabe A2.3**

Um nun diesen vorhandenen Prototypen im Sinne der Nutzerfreundlichkeit zu verbessern, müssen die neuen Query-Strategien, welche den eben genannten neuen Konzepten zur Fragenbildung folgen, mit Regeln belegt werden. Diese sollten Aspekte wie beispielsweise die maximale Anzahl an Zeichen für eine

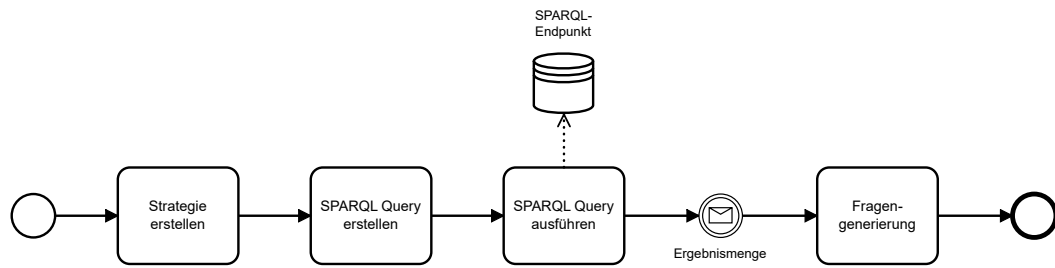


Abbildung 3.1: Die SPARQL-Abfrage im Ablauf.

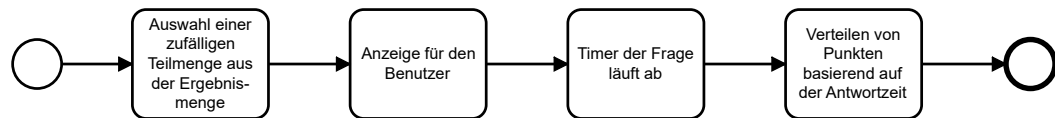


Abbildung 3.2: Ausgabe der Fragen an den Nutzer.

Frage regulieren, um sie so verständlich wie nur möglich zu gestalten. Außerdem kann ein Interpreter entworfen werden, der die Ergebnisse dieser SPARQL-Queries in den existierenden JavaScript-Code des Quiz einfügt. Damit kann der vorhandene Code zum Anzeigen weiterhin verwendet werden. Weiterhin sollte dieser Interpreter auch in der Lage sein, diese Query-Ergebnisse in QuizMaster-Dateien umzuwandeln, sodass die Fragen auch in einer Windows-Desktopapplikation angezeigt und beantwortet werden können.

# Kapitel 4

## Ausführung der Lösung

### 4.1 Lösung für Aufgabe A1.1

#### 4.1.1 Untersuchen der bestehenden Queries

TODO: [Tatsächlichen Referrer zu 1.1.1 hinzufügen](#) Wie in Abschnitt 1.1.1 bereits beschrieben wurde, gibt es bei dem existierenden Prototypen 3 Typen von Strategien, die zur Generierung von Fragen genutzt werden:

- die DEFINITION-Query (A defines B),
- die CONTAINS-Query (A contains B),
- und die SUPPORTS-Query (A supports B).

Die drei vorhandenen Strategien generieren auf Basis dieser Queries Fragen. Um die Verständlichkeit zu erhöhen, werde ich die erste Strategie umfangreicher erklären und umfangreicher auf bestimmte Operationen und Methoden eingehen.

#### DEFINITION-Query

Query:

```
SELECT SAMPLE(replace(str(? def),str(? cl),"X","i") as ? def)
SAMPLE(str(? cl) as ? cl)
SAMPLE(str(? a1l) as ? a1l)
SAMPLE(str(? a2l) as ? a2l)
SAMPLE(str(? a3l) as ? a3l)
```

```

# Gebe ?def, ?cl, ?a1l, ?a2l, ?a3l aus
(
  ?class a owl:Class.
# Sei ?class ein Attribut des Typus Klasse
  ?class rdfs:label ?cl.
# Sei ?cl der woertliche Bezeichner von ?class
  FILTER(LANGMATCHES(LANG(?cl), "en"))
# Sei ?cl in englischer Sprache

  ?class skos:definition ?def.
# Sei ?def die Definition von ?class
  FILTER(STRLEN(?def)>10&&STRLEN(?def)<600).
  FILTER(LANGMATCHES(LANG(?def), "en"))
# Sei ?def in englischer Sprache und zwischen 10 und 600 Zeichen lang

  ?class (!meta:subTopClass){1,2} ?a1,?a2,?a3.
# Seien ?a1, ?a2, ?a3 Attribute des Typus Klasse, die keine Unterklas

  owl:Class ^a ?a1,?a2,?a3.FILTER(
    ?class!=?a1&&?class!=?a2&&?class!=?a3&&?a1<?a2&&?a2<?a3)
# Seien ?a1, ?a2, ?a3 ungleich zueinander und zu ?class

  ?a1 rdfs:label ?a1l.FILTER(LANGMATCHES(LANG(?a1l), "en"))
  ?a2 rdfs:label ?a2l.FILTER(LANGMATCHES(LANG(?a2l), "en"))
  ?a3 rdfs:label ?a3l.FILTER(LANGMATCHES(LANG(?a3l), "en"))
# Seien ?a1l, ?a2l, ?a3l die woertlichen Bezeichnungen von ?a1, ?a2,
) GROUP BY ?class limit 1000

```

Die DEFINITION-Query dient als Ausführung der gleichnamigen Strategie der Generierung für Fragen, bei denen ein Parameter gegeben ist und ein anderer durch Antwortmöglichkeiten ausgewählt werden muss. Dabei ist im Speziellen hier eine Definition einer Klasse *x* vorgegeben, zu der diese Klasse *x* gesucht wird. Der Variable *?class* wird hier der Typus *owl:class* zugewiesen. Dadurch werden alle Attribute der Ontologie, die das Attribut *owl:class* besitzen, als potentielle Werte für *?class* angesehen. Einer zweiten Variable, *?cl*, wird nun der Klurname der Klasse *?class* in der gewünschten Sprache zugewiesen.



Hierbei ist `?class` ist eine Kategorisierung, die eine Menge an Unterelementen verbindet. Dabei sieht sich `?class` einem Primärschlüssel in SQL ähnlich: ein Identifier, der eindeutig und einzigartig ist. Während die Klasse selbst neben einem Identifikationsschlüssel keine Informationen beinhaltet, finden sich diese Informationen in den Unterelementen wieder. Dazu können beispielsweise Informationen wie Definitionen oder Ober- bzw. Unterklassen stecken.

*Um das Beispiel „Skateboard“ verständlich zu halten, bezieht es sich anstelle der SNIK-Ontologie auf Wikidata. Wikidata ist eine Ontologie, die einen Großteil der Informationen der Wissenssammlung Wikipedia zusammenfasst.*

„Skateboard“ ist hier als eigene Klasse mit dem Identifier „Q15783“ gelistet. Sie besitzt unter anderem die Eigenschaft „label“, die mehrsprachig vorliegt. Die Eigenschaft „label“ beinhaltet den Klarnamen der `?class`, also „Skateboard“, in verschiedenen Sprachen. Um nun die gewünschte Sprache auszuwählen, wird nun ein FILTER genutzt. Dieser schließt hier alle Attribute der Eigenschaft label für `?class` aus, die nicht in englischer Sprache sind. Dieses Label wird in der Variable `?cl` gespeichert.

Im nächsten Schritt wird auf ein weiteres Unterelement von `?class` zugegriffen: der Definition. Diese wird der Variable `?def` zugewiesen und liegt erneut mehrsprachig vor. Hier wird erneut auf FILTER zugegriffen, um nur englischsprachige Elemente anzuzeigen. Weiterhin wird die Anzahl der Zeichen so beschränkt, dass nur Ergebnisse zwischen 10 und 600 Zeichen angezeigt werden. Im nächsten Schritt werden nun die Distraktoren generiert. Diese heißen im Beispiel `?a1`, `?a2` und `?a3` und sind ebenfalls Klassen, die jedoch nicht mit `?class` („Skateboard“) zusammenhängen dürfen. Dazu wird die Limitierung `!subTop-Class` angewandt, die beschränkt, dass `?a1`, `?a2` und `?a3` keine Unterklassen von `?class` sein dürfen. Außerdem wird über FILTER verhindert, dass sich die drei Distraktoren untereinander gleichen oder mit `?class` übereinstimmen.

Im letzten Schritt werden die drei Variablen `?a1l`, `?a2l` und `?a3l` mit den Labels von `?a1`, `?a2`, `?a3` versehen. Nun werden die Variablen `?cl`, `?a1l`, `?a2l`, `?a3l` sowie `?def` ausgegeben, wobei bei `?def` noch jedes Vorkommen von `?cl` durch X ersetzt wird. Die Ausgabe der Query wäre nun:

Die eben generierte Fragestellung würde hier wie folgt aussehen:

```

?def: "Ein_X, gelegentlich verdeutscht auch Rollbrett genannt ,
ist ein Brett (Deck) mit zwei Achsen (Trucks) [...]"
?cl:  "Skateboard"
?a1l: "Boot"
?a2l: "Citybike"
?a3l: "Lastkraftwagen"

```

Tabelle 4.1: Beispielausgabe der DEFINITION-Query in Datenform.

Was wird hiervon definiert? „Ein X, gelegentlich verdeutscht auch Rollbrett genannt, ist ein Brett (Deck) mit zwei Achsen (Trucks) [...]“	
Boot	Lastkraftwagen
Skateboard	Citybike

Tabelle 4.2: Fragestellung der DEFINITION-Query.

Neben der richtigen Antwort „Skateboard“ werden hier noch die drei Distraktoren „Boot“, „Lastkraftwagen“ und „Citybike“ als Antwortmöglichkeit vorgegeben.

TODO: Beschreiben des Ergebnisses der Query sowie ihrer Vorgehensweise.

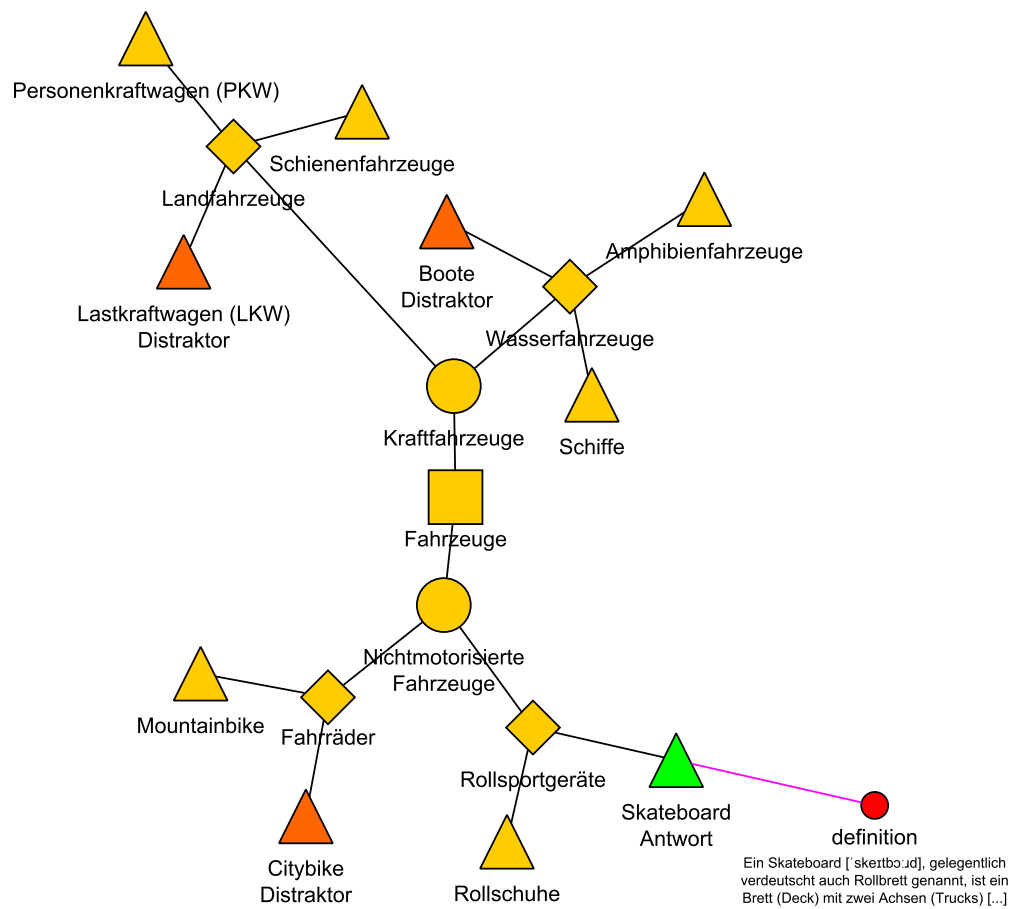


Abbildung 4.1: Fragengenerierung mithilfe der DEFINITION-Query.

# Kapitel 5

## Ergebnis

TODO: Im Ergebniskapitel soll beschrieben werden, inwiefern Sie Ihre in 1.2/1.3 aufgestellten Ziele bzw. Aufgaben erreicht haben oder auch weshalb sie (teilweise) nicht erreicht werden konnten. So soll es möglich sein, dass ein Leser von der Arbeit lediglich die Einleitung und die Zusammenfassung liest und doch die Ergebnisse der Arbeit erfassen kann. Dieses Kapitel kann auch in mehrere Unterkapitel aufgeteilt werden, wenn das sinnvoll ist!

## Kapitel 6

### Diskussion und Ausblick **TODO:** 1–2 Seiten

TODO: In der Diskussion wird das Ergebnis der Arbeit noch einmal kritisch bewertet. Dies umfasst auch neue Probleme, die erst während der Bearbeitung erkannt wurden. Die Eignung für ein konkretes Forschungsprojekt sollte hier diskutiert werden. In der Diskussion kann auch die Kritik des Autors an dem stehen, was er in der Literatur zu dem zu bearbeitenden Thema hier und da gelesen hat. Außerdem soll ein Ausblick gegeben werden, welche weiteren Fragestellungen noch bearbeitet werden sollten.