

# Bewerbung auf ein Fellowship für Innovationen in der digitalen Hochschullehre

## Logik Digital

Thomas Zeume  
TU Dortmund

27. Juni 2017

Logik ist eines der Fundamente vieler moderner Teilgebiete der Informatik. Ob in der künstlichen Intelligenz, in der formalen Software- und Hardwareverifikation, oder als Grundlage für Datenbankanfragesprachen — das Modellieren von Szenarien mit Hilfe logischer Formeln und das Schließen neuen Wissens sind wichtige Kompetenzen für angehende Informatikerinnen und Informatiker.

Im Rahmen dieses Projektes soll ein web-basiertes, interaktives System zur Unterstützung des Erlernens logischer Methoden entwickelt werden, das insbesondere (a) das Modellieren von Wissen und das Schließen neuen Wissens mit Hilfe von Aussagenlogik, Modallogik und Prädikatenlogik unterstützt, und (b) unmittelbares, didaktisch wertvolles Feedback sowie Hilfestellung ermöglicht.

### Inhaltsverzeichnis

<b>1</b>	<b>Beschreibung des Kontextes</b>	<b>2</b>
<b>2</b>	<b>Herausforderungen in der Logik-Lehre und Ziel dieses Projektes</b>	<b>2</b>
<b>3</b>	<b>Vorhandene Software für die Logik-Lehre und eigene Vorarbeiten</b>	<b>3</b>
3.1	Relevante Arbeiten .....	3
3.2	Erfolgte Vorarbeiten.....	4
<b>4</b>	<b>Arbeitsprogramm und detaillierte Beschreibung des Vorhabens</b>	<b>4</b>
4.1	Konzeption und Entwicklung des Systems.....	4
4.2	Didaktischer Kontext und Evaluation .....	7
4.3	Ergründung theoretischer Grundlagen.....	8
<b>5</b>	<b>Vision und Einbindung des Projektes an der TU Dortmund</b>	<b>8</b>
5.1	Nachhaltigkeit und Beurteilung des Erfolges.....	8
5.2	Langfristige Vision und Übertragung auf andere Lehr-Lern-Situationen.....	9
5.3	Einbindung des Projektes an der TU Dortmund.....	9
<b>6</b>	<b>Persönliche Motivation</b>	<b>9</b>
<b>7</b>	<b>Literaturverzeichnis</b>	<b>10</b>
	<b>Zeit- und Arbeitsplan</b>	<b>11</b>
	<b>Finanzierungsplan</b>	<b>11</b>

## 1 Beschreibung des Kontextes

In vielen informatischen Anwendungen spielen logische Formalismen eine große Rolle. In der künstlichen Intelligenz wird Wissen häufig durch logische Formeln repräsentiert, und mit Hilfe logischer Schlussverfahren neues Wissen gewonnen. Die Grundlage vieler moderner Datenbankanfragesprachen wie SQL, Cypher und XQuery sind logische Formalismen. In der Programm- und Hardwareverifikation werden wünschenswerte Eigenschaften oft durch auf Logik beruhenden Spezifikationssprachen wie beispielsweise den Temporallogiken LTL und CTL\* spezifiziert, und Systeme dann mithilfe von logischen Verfahren auf Korrektheit bezüglich der Spezifikationen überprüft.

Grundlegende Kenntnisse im Bereich der Logik sind somit für angehende Informatikerinnen und Informatiker unabdingbar, um in diesen Anwendungsfeldern erfolgreich zu sein. Solide Kenntnisse in formaler Modellierung helfen Studierenden nicht zu letzt auch, Probleme strukturiert anzugehen und zu lösen.

Wie in vielen Universitäten spiegelt sich auch an der TU Dortmund der hohe Stellenwert von Logik für die Informatik in einer festen Verankerung von vielen logiknahen Veranstaltungen im Lehrplan für den Bachelor und Master Studiengang Informatik wieder. Die Grundlage für viele dieser Veranstaltungen bildet in Dortmund eine Pflichtvorlesung “Logik für Informatiker” im 3. Semester des Bachelorstudiengangs. In dieser werden Aussagenlogik, Modallogik und Prädikatenlogik vorgestellt und die Modellierung von informatischen Szenarien sowie das Schlussfolgern von Wissen aus solchen Modellierungen vermittelt.

Auf die Einführungsveranstaltung bauen dann u.a. Veranstaltungen zur Wissensrepräsentation (“Darstellung, Verarbeitung und Erwerb von Wissen”, Bachelor Wahlpflicht; “Wissensmodellierung”, Bachelor Fachprojekt), zur Verifikation (“Grundlagen des Model Checking”, “Logische Methoden des Softwareengineering” beide Master Vertiefung), sowie Veranstaltungen im Datenbankbereich (“Informationssysteme”, Bachelor Pflichtvorlesung; “Datenbanktheorie”, Master Vertiefung) auf.

## 2 Herausforderungen in der Logik-Lehre und Ziel dieses Projektes

Während die Vermittlung grundlegender Logikkenntnisse für angehende Informatikerinnen und Informatiker von immenser Bedeutung ist, stellen die notwendigerweise eher theoretischen Grundlagen einen Großteil der Informatik-Studierenden vor eine große Herausforderung. Auch weil viele Studierende vom Programmieren zum Informatik-Studium kommen, und damit eher an praktisch-orientierten Veranstaltungen interessiert sind. Gerade für diese Studierenden ist es sehr wichtig motivierende Anreize zu schaffen.

### **Herausforderung A** *Steigern der Motivation Studierender für die Grundlagen der Logik*

In der Grundlagenvorlesung zur Logik steht die Modellierung von informatischen Szenarien und das Folgen von neuem Wissen im Vordergrund. So wird beispielsweise im prädikatenlogischen Teil der Vorlesung vorgestellt, wie

- (a) eine Problemstellung prädikatenlogisch modelliert werden kann,
- (b) die dabei aufgestellten Formeln in eine passende Normalform überführt werden können, und
- (c) neues Wissen (in Form von Formeln) mit Hilfe prädikatenlogischer Resolution gefolgert werden kann.

In den Übungen wird dieser Prozess dann anhand von einzelnen Übungsaufgaben geübt. Aufgrund beschränkter Ressourcen können Studierende jedoch nur zu wenigen Übungsaufgaben ausführliches Feedback erhalten.

### **Herausforderung B** *Studierenden bedarfsgerechte Hilfestellung sowie Feedback ermöglichen*

Beim Erlernen von neuen Formalismen ist schnelles, präzises Feedback notwendig. Momentan lernen die Studierenden in Woche  $n$  ein Thema kennen, bearbeiten bis Woche  $n + 1$  Übungen zu diesem Thema, und erhalten dann in Woche  $n + 2$  Feedback zu ihren Lösungen. Die Studierenden beschäftigen sich mit neu erlernten Methoden somit verteilt über zwei Wochen, wodurch die Haltbarkeit des Wissens erhöht wird. Allerdings schreitet in diesen zwei Wochen die Vorlesung voran, sodass wenn die in Woche  $n$  kennengelernten Methoden nicht verstanden wurden, dies zu spät erkannt und behoben werden kann. In der Vorlesung “Logik für Informatiker” tritt dieses Problem besonders für die Teile zur Modallogik und Prädikatenlogik

### 3 Vorhandene Software für die Logik-Lehre und eigene Vorarbeiten

auf, da die Studierenden hier oft nicht auf Erfahrungswerte aus anderen Vorlesungen zurückgreifen können. Um Unklarheiten und Missverständnisse frühzeitig erkennen und auszuräumen zu können, ist es für Studierende hilfreich unmittelbar neues Wissen auszuprobieren und dazu Feedback zu erhalten.

Diese Herausforderungen sollen im Rahmen des Projektes unter Ausnutzung digitaler Technologien angegangen werden.

**Ziel des Projektes** *Es soll ein web-basiertes, interaktives System zur Unterstützung des Erlernens logischer Methoden entwickelt werden, das (a) den gesamten Modellierungsprozess für die Aussagenlogik, Modallogik und Prädikatenlogik unterstützt, und (b) unmittelbares, didaktisch wertvolles Feedback sowie individuelle Hilfestellung ermöglicht.*

Ein modernes, einfach zu bedienendes Websystem allein kann bereits die Motivation steigern, siehe beispielsweise [Oblinger et al., 2005]. Langfristig sollen bewusst praxis-orientierte Anbindungen motivierende Anreize schaffen.

Das System soll so gestaltet sein, dass einzelne Teilaufgaben im Modellierungsprozess leicht austauschbar sind. Einerseits können auf diese Weise unterschiedliche Verfahren unterstützt werden, beispielsweise sowohl Resolution als auch Tableau-Kalkül für das Folgern in der Aussagenlogik. Andererseits ist es denkbar, dass unterschiedliche didaktische Anforderungen zu unterschiedlichen Umsetzungen einer Teilaufgabe führen. So kann beispielsweise beim Umformen von Formeln das Erlernen und Verwenden der korrekten Notation ein Lernziel sein; das strategische Herangehen an die Umformung ein anderes.

## 3 Vorhandene Software für die Logik-Lehre und eigene Vorarbeiten

Seit Anfang der 90er Jahre wurden diverse Tutoring-Systeme für den Logikbereich entworfen. Während viele Systeme einzelne Aufgaben des oben skizzierten Modellierungsprozesses beinhalten, scheint keines darauf ausgelegt den gesamten Prozess abzudecken. Nach einem kurzen Überblick über vorhandene Systeme, beschreiben wir kurz von uns durchgeführte Vorarbeiten.

### 3.1 Relevante Arbeiten

Der folgende Überblick konzentriert sich auf web-basierte Systeme. Viele ältere Tutoring-Systeme sind spezifisch für Windows oder Mac, und daher nur für eine eingeschränkte Gruppe der Studierenden zugänglich.

Einige vorhandene Systeme decken von uns anvisierte Übungsaufgaben ab. Im *LogEX* System kann die Umformung von aussagenlogischen Formeln geübt werden [Lodder et al., 2015, Lodder and Heeren, 2011]. Das Folgern neuen Wissens über Kalküle, die dem natürlichen Schließen ähnlich sind, wird von vielen Systemen unterstützt, siehe u.a. [Sieg, 2007, Huertas et al., 2011, Ehle et al., 2015, Gasquet et al., 2011]. Resolution, die in der Logik Einführungsvorlesung in Dortmund als Schlußverfahren gelehrt wird, wird jedoch nach unserem Kenntnisstand nur vom *AELL* System [Huertas et al., 2011] unterstützt (welches jedoch nicht öffentlich zugänglich ist, und nur auf spanisch und katalanisch zur Verfügung steht). Mit der Umgebung *Tarski's World* kann in einer 3D-Welt die Auswertung von prädikatenlogischen Formeln gelernt werden. Ein spielerischer Ansatz für das Erlernen der Themen einer typischen Logikeinführung wird prototypisch in [Schäfer et al., 2013] verfolgt.

Digitale Werkzeuge kommen auch in interaktiven Logik-Lehrbüchern zum Einsatz. So unterstützt beispielsweise die Lehr-Umgebung aus [Fricke, 2012] das Überführen von in Textform gegebenen Aussagen in logische Formeln durch einen Markieren-und-Ersetzen-Verfahren. Viele kleine interaktive Aufgaben finden sich im interaktiven Lehrbuch [Velleman, 2010] sowie in den unterstützenden Webseiten zum Buch *Power of Logic* [Wasserman et al., 2012]. Eine Inspiration für die Darstellung von Modellen für modallogische Formeln findet sich in [Kirsling, 2015].

Eine Übersicht über weitere, zum Teil auch ältere Tutoring-Systeme, findet sich in [Huertas, 2011].

Insgesamt werden einige Aspekte schon durch verschiedene Systeme abgedeckt. Die verwendeten Technologien sind jedoch sehr unterschiedlich und unterscheiden sich zudem zum Teil stark in ihrer Modernität,

## 4 Arbeitsprogramm und detaillierte Beschreibung des Vorhabens

so dass eine Zusammenführung in ein Gesamtsystem zur kontinuierlichen Unterstützung in einer Logik-Veranstaltung unpraktikabel bzw. unmöglich erscheint.

### 3.2 Erfolgte Vorarbeiten

Ein prototypisches System für die Unterstützung des Erlernens des Modellierungsprozesses in der Aussagenlogik wurde bereits entwickelt<sup>1</sup>. Das System besteht momentan aus einer auf didaktische Anwendungen zugeschnittenen Logikbibliothek (bislang mit Unterstützung für elementare syntaktische und Semantische Funktionalität der Aussagenlogik und sowie elementare Syntax der Modallogik), einer webbasierten Oberfläche (auf GWT-basierend) sowie einer Aufgabenspezifikationsprache die die Komposition von Teilaufgaben erlaubt. Musterhaft sind Aufgaben implementiert, die das Üben der aussagenlogischen Modellierung durch Formeln erlauben, aus denen anschließend (nach einer Umformung der Formeln) neues Wissen gefolgert werden kann. Für die Erstellung aussagenlogischer Formeln ist bereits ein Feedbackmechanismus implementiert, der Hinweise zur Korrektur einer fehlerhaften Formel liefert. (Dieser wird momentan getestet und ist im Live-System noch nicht freigeschaltet, Stand: 30. Juni 2017.)

Der Prototyp wurde im Wintersemester 2016/2017 in geringem Umfang und auf freiwilliger Basis in der Logik-Einführung eingesetzt. Er wurde von den Studierenden in einer begrenzten Umfrage mit 9 Teilnehmern positiv evaluiert. Ihren mit dem System erzielten Lernfortschritt beurteilten 7 Studierende als gut bis sehr gut, und 7 Teilnehmer würden das Werkzeug beispielsweise bei der Klausurvorbereitung wiederverwenden. In Freitextkommentaren wurde insbesondere das direkte Feedback sowie die Möglichkeit Teilaufgaben und vollständige Modellierungsaufgaben zu üben als sehr hilfreich bewertet.

Die Arbeiten an der prototypischen Umsetzung begannen im April 2016 und wurden bislang von einem Doktoranden, Gaetano Geck, sowie zwei wissenschaftlichen Hilfskräften, Sebastian Peter und Jonas Schmidt, unterstützt.

## 4 Arbeitsprogramm und detaillierte Beschreibung des Vorhabens

Wie bereits beschrieben, ist das Ziel des Projektes die Implementierung eines web-basierten, modernen Systems zur Unterstützung des Lernprozesses logischer Formalismen. Entsprechend den skizzierten Herausforderungen in der Lehre, ist es wichtig, dass (1) die Oberfläche intuitiv und leicht zu bedienen ist, und (2) vom System unmittelbar didaktisch sinnvolle Hilfestellungen und Feedback gegeben werden.

Der Kern des Arbeitsprogramms bildet naturgemäß die Konzeption und Entwicklung des Systems (Abschnitt 4.1). Die Entwicklung soll zudem didaktisch begleitet und das Ergebnis evaluiert werden (Abschnitt 4.2). Sich ergebende Fragestellungen theoretischer Natur, insbesondere der Entwurf und die Entwicklung effizienter Algorithmen, sollen gesondert in einem theoretischen Arbeitspaket untersucht werden (Abschnitt 4.3). Abbildung 1 gibt einen Überblick über die Bereiche und Arbeitspakete.

### 4.1 Konzeption und Entwicklung des Systems

Die Entwicklung des Systems unterteilt sich in verschiedene Bereiche. Für die Entwicklung sind neben Gaetano Geck und dem Antragsteller fünf wissenschaftliche Hilfskräfte aus der Informatik vorgesehen. Drei der wissenschaftlichen Hilfskräfte sollen aus den beantragten Projektmitteln finanziert werden (siehe Finanzierungsplan).

#### *Arbeitspaket S.1 Erweiterung der bestehenden didaktik-orientierten Logikkomponente*

Im Gegensatz zu vielen verfügbaren Logikbibliotheken muss die hier zu entwickelnde Komponente die geplante didaktische Anwendung unterstützen. So sollte beispielsweise das schrittweise Umformen von Formeln unterstützt werden, für das bei Bedarf auch Erklärungen und angewendete Regeln abrufbar sind; und das Erzeugen von Feedback erfordert das Vorhalten von Formeln in geeigneten Datenstrukturen.

Bislang unterstützt die prototypische Umsetzung Syntax und Semantik der Aussagenlogik und z.T. modallogische Syntax. Im Projekt soll diese Funktionalität finalisiert und auf die Semantik von Modallogik sowie auf Syntax und Semantik von Prädikatenlogik erweitert werden. Für die Modallogik sind hier keine Probleme zu erwarten, gegebenenfalls sollen dabei bereits die Grundlagen für die Integration von

<sup>1</sup>Ausprobiert werden kann der Prototyp hier: <http://gaga.cs.tu-dortmund.de:8080/LogicWeb/>

#### 4 Arbeitsprogramm und detaillierte Beschreibung des Vorhabens

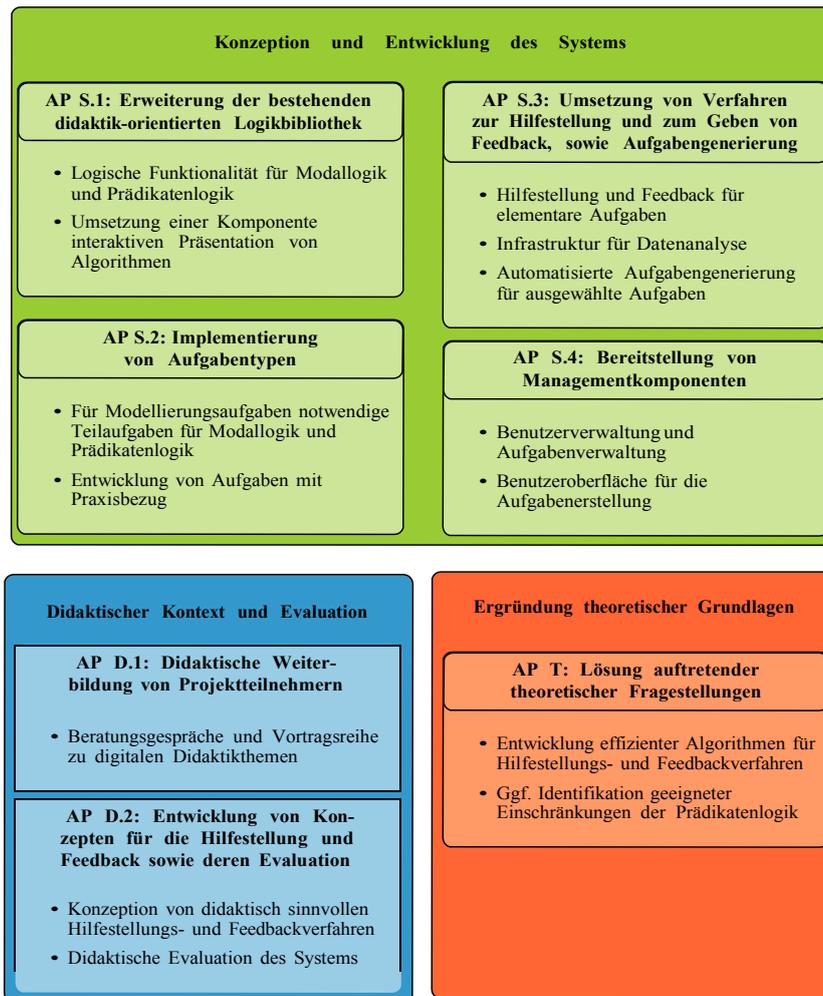


Abbildung 1: Übersicht über die Arbeitsbereiche und Arbeitspakete.

Temporallogiken (wie sie in der Verifikation Verwendung finden) geschaffen werden. Für die Unterstützung insbesondere semantischer Funktionalität für die Prädikatenlogik sollen im Arbeitspaket T zunächst mögliche Vorgehensweisen sondiert werden.

Außerdem sollen für alle drei Logiken unterstützende Methoden zur Generierung von Feedback, Hilfestellung und Aufgaben zur Verfügung gestellt werden (siehe Arbeitspaket 3). Dabei soll unter anderem der implementierte deklarative Mechanismus zur Selektion und Veränderung von Teilformeln, der bereits in der Transformation von aussagenlogischen Formeln sowie der Feedbackgenerierung für die Erstellung aussagenlogischer Formeln Anwendung findet, auf Modallogik und Prädikatenlogik erweitert werden.

Außerdem soll ein allgemeiner Mechanismus für die interaktive Präsentation von iterativen Verfahren implementiert werden. Dieser soll es ermöglichen typische Verfahren schrittweise durchführen und erklären zu lassen.

#### **Arbeitspaket S.2 Implementierung von Aufgabentypen**

In diesem Arbeitspaket sollen konkrete Übungsaufgaben umgesetzt werden. Für jeden Aufgabentyp beinhaltet dies das Programmieren der Anwendungslogik sowie das Entwerfen einer intuitiven Benutzeroberfläche. Wie oben beschrieben sind Teilaufgaben für den aussagenlogischen Modellierungsprozess bereits umgesetzt. Fehlende Teilaufgaben für die Aussagenlogik werden voraussichtlich noch vor 2018 und somit vor Projektbeginn fertiggestellt. Im Projekt sollen diese Aufgaben stetig verbessert werden.

Im Rahmen des Projektes soll der Fokus auf der Implementierung typischer Übungsaufgaben für die

#### 4 Arbeitsprogramm und detaillierte Beschreibung des Vorhabens

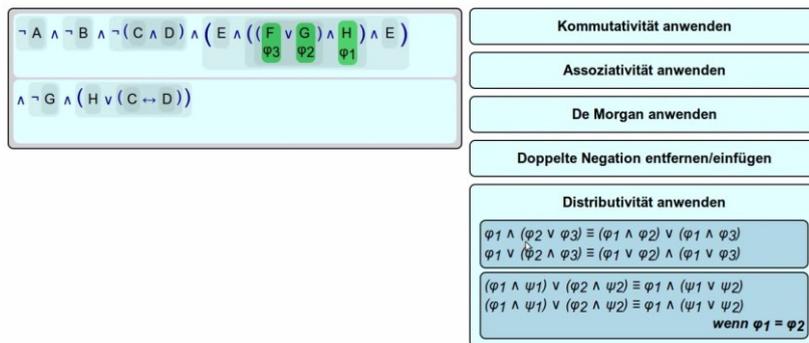


Abbildung 2: Entwurf der graphischen Oberfläche für die Umformung von Formeln. Nach dem Auswählen von Teilformeln müssen passende Umformungsregeln ausgewählt werden.

Modallogik sowie die Prädikatenlogik liegen. Unter anderem sollen für beide Logiken Aufgaben zur Auswertung von Formeln, zur Konstruktion von Modellen, dem Aufstellen von Formeln, sowie der Umformung von Formeln implementiert werden. Für die Modallogik sollen zudem Aufgaben zum Üben des Tableaukalküls und für die Prädikatenlogik Aufgaben zum Üben der Resolution umgesetzt werden.

Während der Aufwand für die Umsetzung modallogischer Aufgaben voraussichtlich mit dem Aufwand für die Aussagenlogik vergleichbar ist, wird die Umsetzung der Aufgaben zur Prädikatenlogik aller Voraussicht nach aufwendiger. So ist beispielsweise die Überprüfung, ob eine von einem Studierenden angegebene prädikatenlogische Formel korrekt ist, im allgemeinen algorithmisch nicht möglich (da unentscheidbar). Zunächst sollen für die Prädikatenlogik deshalb Übungsaufgaben zur Auswertung von Formeln, zu syntaktischen Umformungen, Unifikation sowie Resolution umgesetzt werden. Für Aufgaben stärker semantischer Natur sollen im begleitenden theoretischen Arbeitspaket T vorab gute Herangehensweisen untersucht werden. Gegebenenfalls können die gefundenen Lösungen erst nach dem Projektzeitraum umgesetzt werden.

In diesem Arbeitspaket sollen zudem die Elemente für die graphische Benutzeroberfläche entwickelt werden. Dabei sollen auch die im Prototyp verwendeten Elemente überarbeitet werden. So sind beispielsweise für die Umformung von Formeln zwei unterschiedliche Benutzeroberflächen angedacht; eine zum Erlernen der elementaren Umformungsregeln (siehe Abbildung 2), und eine zum freien Umformen für fortgeschrittene Studierende.

Im Rahmen dieses Arbeitspaketes sollen zudem erste konkrete Aufgabenideen für einen stärkeren Praxisbezug entwickelt werden. Denkbar ist die Einbindung von Aufgaben zum Lernen des Umgangs mit einem SAT-Solver oder der Übersetzung von prädikatenlogischen Formeln in SQL-Anfragen.

Für alle implementierten Aufgaben sollen im Arbeitspaket D.2 Hilfestellungs- und Feedbackmechanismen gefunden und dann in Arbeitspaket S.3 implementiert werden.

#### **Arbeitspaket S.3** Umsetzung von Verfahren zur Hilfestellung und zum Geben von Feedback, sowie Aufgabengenerierung

Im Zusammenspiel mit den Arbeitspaketen D.2 und T sollen didaktisch sinnvolle und gleichzeitig algorithmisch effiziente Verfahren für Hilfestellung und Feedback umgesetzt werden. Für das Aufstellen aussagenlogischer Formeln wird Feedback bereits prototypisch unterstützt. Dazu wird versucht aus einer von einem Studierenden angegebenen, möglicherweise fehlerhaften Formel durch Fehlerregeln eine zur korrekten Formel äquivalente Formel herzuleiten. Voraussichtlich lässt sich dieses Verfahren einfach auf modallogische Formeln erweitern.

In diesem Arbeitspaket soll zudem die Infrastruktur zur Erhebung von anonymisierten Daten über die implementierten Hilfestellungs- und Feedbackverfahren geschaffen werden, sodass diese im Rahmen von Arbeitspaket D.2 aus didaktischer Sicht evaluiert werden können.

Außerdem soll für ausgewählte Teilaufgaben eine automatische Generierung von Übungsaufgaben unter gegebenen didaktischen Parametern implementiert werden. So könnten beispielsweise für Formelumformungsaufgaben maßgeschneidert Formeln generiert werden, für die zweimal distributiv und einmal die De Morgan'sche Regel angewendet werden muss um eine Zielformel in einer Normalform zu erhalten. Auf diese Weise lassen sich dann Aufgaben verschiedenen Schwierigkeitsgrades spezifizieren.

#### 4 Arbeitsprogramm und detaillierte Beschreibung des Vorhabens

##### **Arbeitspaket S.4** *Bereitstellung von Managementkomponenten*

Für eine leichte Nutzung des Systems im laufenden Übungsbetrieb ist eine Benutzerverwaltung, ein System für die Aufgabenverwaltung sowie Unterstützung bei der Aufgabenerstellung notwendig. In diesem Arbeitspaket sollen zunächst vorhandene Lösungen für Benutzer- und Aufgabenverwaltung auf eine Wiederverwendung hin evaluiert werden. Anschließend sollen gefundene Lösungen je nach Ergebnis integriert oder neue Lösungen implementiert werden.

Für die Aufgabenerstellung soll zudem eine für Übungsleiter/innen einfach zu bedienende Oberfläche implementiert werden.

**Bemerkungen zur Wiederverwendbarkeit** Um eine Erweiterung auf andere Gebiete (wie beispielsweise die theoretische Informatik oder mathematisches Modellieren zu ermöglichen, siehe Abschnitt zur langfristigen Vision) soll beim Entwurf des Systems stark darauf geachtet werden, Komponenten wiederverwendbar zu entwerfen. So ist die Spezifikation von Aufgaben über XML Dokumente bereits im Prototypen allgemein implementiert. Dies soll auch bei der Oberfläche für die Aufgabenerstellung so fortgesetzt werden. Auch der Zugriff auf Formelbestandteile ist so gestaltet, dass dieselbe Bibliothek für logische Formeln, mathematische Formeln und andere baumartige Strukturen verwendet werden kann.

Es ist absehbar, dass Elemente der grafischen Benutzeroberfläche in leicht abgewandelter Form auch in anderem Kontext auftreten werden. So ähneln Darstellungen von Kripke-Strukturen aus der Modallogik der Darstellung endlicher Automaten in der theoretischen Informatik. Hier soll durchgängig vorab analysiert werden, für welche ähnlichen Szenarien Elemente wiederverwendet werden können, und die Implementierung entsprechend erfolgen.

## 4.2 Didaktischer Kontext und Evaluation

Im Rahmen dieses Teilprojektes sollen einerseits die Projektteilnehmer didaktisch geschult und andererseits ggf. didaktische Erkenntnisse in der Umsetzung des Websystems berücksichtigt werden.

Hierfür ist eine Zusammenarbeit mit dem Institut für Entwicklung und Erforschung des Mathematikunterrichts (IEEM) der TU Dortmund sowie der Einsatz einer aus Projektmitteln finanzierten wissenschaftlichen Hilfskraft aus dem Didaktikbereich vorgesehen.

##### **Arbeitspaket D.1** *Didaktische Weiterbildung von Projektteilnehmern*

Den am Projekt beteiligten Mitarbeitern und wissenschaftlichen Hilfskräften aus der Informatik sollen grundlegende didaktische Kenntnisse vermittelt werden. Dies soll einerseits durch regelmäßige Gespräche mit den direkt im Projekt eingebundenen Mitarbeitern des IEEM erfolgen. Auch kurze Vorträge zu theoretischen Hintergründen wie dem Geben von Feedback und Hilfestellungen sowie dem Aufsetzen von Evaluationen sollen stattfinden. Über aktuelle Entwicklungen in der didaktischen Anwendung digitaler Technologien sollen im Rahmen des Projektes zwei bis drei fakultätsweite Vorträge informieren. Angedacht ist hier, Experten aus dem Bereich Tutoring-Systeme (beispielsweise aus den Projekten AELL [Huertas et al., 2011] oder LogEX [Lodder et al., 2015]) und zur Evaluation von Tutoring-Systemen einzuladen.

##### **Arbeitspaket D.2** *Entwicklung von Konzepten für die Hilfestellung und Feedback sowie deren Evaluation*

Die Beachtung des Kenntnisstandes eines Studierenden ist bei der Wahl von Aufgaben, Hilfestellungen und Feedback sehr wichtig. Eine typische Herangehensweise von Tutoring-Systemen ist es beispielsweise, für Teilaufgaben Feedbackvarianten zu unterschiedlichen Aspekten und mit verschiedenem Detailgrad zur Verfügung zu stellen, und diese dann entsprechend dem Kenntnisstand auszuwählen (siehe beispielsweise [Vanlehn, 2006] für einen Überblick). So kann für einen Studierenden mit großem Vorwissen im Erstellen von Formeln das Feedback “Die von dir angegebene Formel ist leider falsch” gegeben werden, während bei einem Studierenden ohne großes Vorwissen detaillierteres Feedback wie “Erinnerst du dich daran, dass du bei Nur-Dann-Aussagen etwas zu beachten hast?” sinnvoll ist.

Der Fokus in diesem Projekt soll zunächst darauf liegen, für Teilaufgaben Hilfestellungs- und Feedbackverfahren zu verschiedenen Aspekten und unterschiedlichem Detailgrad zu entwickeln. Es ist angedacht das System dann später (voraussichtlich nach Abschluss dieses Projektes) um eine Komponente zu erweitern, die den individuellen Lernstand von Studierenden modelliert und diesen bei der Wahl von Aufgaben

sowie bei Hilfestellung und dem Geben von Feedback berücksichtigt. Für die Modellierung des Lernstandes kann beispielsweise für jeden Studierenden vorgehalten werden, welche Aufgaben mit welchem Schwierigkeitsgrad bereits bewältigt wurden.

Für das Geben von Feedback ist ein genaues Verständnis möglicher Fehlertypen notwendig. Für das Aufstellen aussagenlogischer und modallogischer Formeln haben wir in einer Vorabuntersuchung beispielhaft Fehler zusammengestellt, die in einer Klausur gemacht wurden. Für die Aussagenlogik haben wir daraus deklarative Fehlerregeln abstrahiert. Mithilfe dieser lässt sich Feedback zu unterschiedlichen Aspekten und mit unterschiedlichem Detailgrad generieren. Es ist geplant die Implementierung bis zum Beginn des Wintersemesters 2017/2018 fertigzustellen und dann zu evaluieren.

Eine ähnliche Analyse soll im Rahmen des Projektes für andere Aufgabentypen erfolgen. Die Fehler beim Aufstellen modallogischer Formeln können voraussichtlich ähnlich gehandhabt werden. Für Aufgabentypen eher algorithmischer Natur, wie der Umformung von Formeln und der Resolution können, nach einer gewissen Zeit oder eine Reihe nicht zielführender Schritte Hilfestellungen wie “Versuche es doch mal...” sinnvoll sein. Es soll unter Berücksichtigung der in [Aleven et al., 2003] beschriebenen Problematiken didaktisch-sinnvolle Hilfestellungsverfahren entwickelt werden.

Die implementierten Hilfestellungs- und Feedbackmechanismen sollen evaluiert werden. Orientiert an [Shute and Regian, 1993] soll hier zusammen mit dem IEEM ein geeignetes Verfahren gefunden werden. Angedacht sind unter anderem begleitende Interviews mit Studierenden. Zusammen mit detaillierten anonymisierten Fehlerdaten (die mit Hilfe der in Arbeitspaket S.3 implementierten Mechanismen gewonnen werden können) erhoffen wir uns Erkenntnisse darüber ableiten zu können, wie Studierende Logik lernen und wie sich der Lernprozess verbessern lässt.

### 4.3 Ergründung theoretischer Grundlagen

Wie oben beschrieben, ist es für einige der zu implementierenden Aufgaben nötig, auch theoretische Fragestellungen zu betrachten.

#### *Arbeitspaket T Lösung auftretender theoretischer Fragestellungen*

Auf der einen Seite müssen für die Feedbackgenerierung effiziente Algorithmen gefunden werden. Allerdings ist bereits für Aussagenlogik und Modallogik das Finden guten Feedbacks für eine von einem Studierenden angegebene Formel nach momentanen Forschungsstand im Allgemeinen nicht effizient möglich (da NP-schwierig). Hier gilt es auszunutzen, dass die Eingaben in unserem Fall oft klein oder anderweitig eingeschränkt sind. Die zu entwickelnden Algorithmen müssen dies berücksichtigen.

Für die Prädikatenlogik ist, wie bereits beschrieben, schon das Überprüfen von von Studierenden angegebenen Formeln auf Korrektheit algorithmisch nicht lösbar. Hier sollen zunächst evaluiert werden, ob vorhandene logische Bibliotheken zum automatischen Beweisen wiederverwendet werden können. Zudem sollen weitere Alternativen ausgelotet werden. Denkbar ist beispielsweise nur eingeschränkte Formelklassen zuzulassen oder nur randomisiert auf Korrektheit zu testen.

Im Rahmen dieses Arbeitspaketes sollen die theoretischen Grundlagen nur soweit erkundet werden, wie es für eine praktikable Umsetzung nötig ist. Es ist jedoch geplant, die auftretenden Fragen in unsere zukünftige fachliche Forschung einfließen zu lassen.

Dieser Bereich soll von Gaetano Geck und dem Antragsteller bearbeitet werden.

## 5 Vision und Einbindung des Projektes an der TU Dortmund

### 5.1 Nachhaltigkeit und Beurteilung des Erfolges

Unser Ziel ist es, dass die Webumgebung fester Bestandteil des Übungsbetriebes im Bereich Logik an der TU Dortmund wird. Dazu sollen wie oben beschrieben zunächst die grundlegenden Verfahren aus der Aussagenlogik, Modallogik und Prädikatenlogik zur Verfügung gestellt werden. Der Prototyp wird bereits jetzt in der Grundlagenvorlesung verwendet, und fertig werdende Komponenten sollen stets sobald wie möglich im Lehrbetrieb eingesetzt werden. Langfristig soll die Webumgebung auch für die Logik-Lehrveranstaltungen anderer Universitäten zur Verfügung gestellt werden. Interesse am Einsatz des Systems besteht bereits von Prof. Dr. Wim Martens (Universität Bayreuth).

Es ist zudem geplant das System auf internationalen Konferenzen wie der *Conference on Innovation and Technology in Computer Science Education (ITiCSE)* einem breiteren Publikum vorzustellen. Der Quellcode des Projektes soll mittelfristig unter einer Open-Source-Lizenz zur Verfügung gestellt werden, um die Wiederverwendbarkeit zu vereinfachen.

Die Evaluation des didaktischen Erfolges ist, wie in Arbeitspaket D.2 beschrieben, fester Bestandteil des Projektes. Eine wissenschaftliche Hilfskraft aus der Didaktik soll diese Evaluation vornehmen. Dazu sollen einerseits die Benutzer befragt werden, andererseits anonymisierte Daten zum Lernfortschritt erhoben werden. Eine erste, kleine Evaluation des Prototypen war wie oben beschrieben positiv.

Das Projekt soll als Grundlage für ein auf einen breiteren Rahmen angelegtes Folgeprojekt zur Digitalisierung der Lehre in der theoretischen Informatik dienen.

### 5.2 Langfristige Vision und Übertragung auf andere Lehr-Lern-Situationen

Nach der Abdeckung der im Projekt anvisierten Logiken, sollen mittelfristig sowohl Komponenten für praktische Anbindungen geschaffen als auch forschungsnähere Aufgaben eingebunden werden. So ist es von einer modallogischen Komponente nur ein kleiner Schritt zur Unterstützung von in der Verifikation eingesetzten Temporallogiken wie LTL und CTL\*. Auch Aufgaben zu Datenbankabfragesprachen wie SQL sowie deren Verbindung zur Prädikatenlogik sollen zukünftig unterstützt werden.

Die in diesem Projekt geschaffene Infrastruktur und die gewonnenen Erfahrungen sollen langfristig auf breitere Gebiete angewendet werden. Wir planen beispielsweise auch die anderen theoretischen Grundlagenveranstaltungen der Informatik abzudecken. Grundsätzlich können die in diesem Projekt gewonnenen Erfahrungen sowie einige der entwickelten Komponenten auch für ähnliche Projekte für andere Fachbereiche, in denen mathematische Formalismen gelehrt werden, Anwendung finden.

Wie bereits im Arbeitsprogramm beschrieben, soll deshalb beim Entwurf des Systems stark darauf geachtet werden, Komponenten wiederverwendbar zu entwerfen.

### 5.3 Einbindung des Projektes an der TU Dortmund

Das Projekt wird aktiv durch Prof. Dr. Gabriele Kern-Isberner und Prof. Dr. Schwentick unterstützt, die regelmäßig die Vorlesung „Logik für Informatiker“ lesen. Die Anwendung soll im Übungsbetrieb der Vorlesung eingesetzt und evaluiert werden. Der Prototyp wurde in kleinem Umfang bereits im Wintersemester 2016/2017 eingesetzt. Zudem stehen Frau Kern-Isberner und Herr Schwentick mit Rat zur Seite. Die Entwicklung des prototypischen Systems wird sowohl durch die Fakultät (durch Mittel zur Sicherung der Qualität in der Lehre) als auch vom Lehrstuhl Logik in der Informatik unterstützt.

Vom Institut für Entwicklung und Erforschung des Mathematikunterrichts soll das Projekt durch die beiden wissenschaftlichen Mitarbeiter Christian Büscher und Carina Zindel (beide Master of Education, Mathematik und Informatik) aus fachdidaktischer Sicht begleitet werden. Prof. Dr. Johannes Fischer, der in Dortmund in der Informatik die Lehramtsausbildung für Informatik betreut, steht dem Projekt beratend zur Seite.

## 6 Persönliche Motivation

**Lehren macht mir Spaß!** Neue Wege zu finden, Studierende beim Lernen zu unterstützen, finde ich faszinierend. Der einfachste Weg Inhalte zu vermitteln ist Begeisterung zu entfachen. In der Logik ist das manchmal nicht ganz einfach; gerade auch weil in den Grundlagenvorlesungen oft wenig Zeit bleibt, Spaß durch Exkursionen zu praxis- und forschungsnahen Themen zu wecken. Webbasierte Systeme erlauben es hier neue Wege zu gehen. Ich bin mir sicher, dass sich mit einem modernen System und unter Verwendung geschickter Praxis- und Forschungsverbindungen, Studierende in die Welt der Theorie, insbesondere der Logik, locken lassen.

Auf der Suche nach guten Lehrformen war ich in den letzten beiden Jahren auch an der Erstellung eines Interview-Bandes zur Hochschullehre im internationalen Rahmen beteiligt [Albrecht et al., 2017]. Einige von unseren Interviewpartnern verwendete digitale Lehrformate haben mich beeindruckt. Ich würde mich freuen im Rahmen des Fellowship auch die Lehre in der Logik moderner gestalten zu können.

**Theorie macht Spaß (und kann niemandem schaden)!** Rational und strukturiert zu denken war, ist, und wird auch in Zukunft wichtig sein. Nicht nur für Informatiker, sondern für uns alle. Es ist eine Herausforderung für die Gesellschaft, aber auch für die Wissenschaft, alternativen Fakten und Informationsblasen im Internet etwas entgegenzusetzen. In der Hoffnung einen kleinen Beitrag zu leisten, möchte ich langfristig das im vorgeschlagenen Projekt entwickelte System nutzen um ein interaktives Logik-Lehrbuch für einen größeren Personenkreis, insbesondere auch für Kinder, zu erstellen.

**Austausch mit anderen Fellows** Gerade für Nachwuchswissenschaftler:innen ist es oft schwierig ein gutes Netzwerk für die Diskussion von Lehrideen aufzubauen. Vom Austausch mit anderen Fellows verspreche ich mir neue Ideen und Formate für die eigene Lehre. Ich glaube aber auch, dass ein Treffen Interessierter zu ganz neuen Ideen in der digitalen Hochschullehre führen kann. Meine Expertise aus der Informatik hoffe ich hier einbringen zu können.

## 7 Literaturverzeichnis

- [Albrecht et al., 2017] Albrecht, J., Brenner, C., Gökce, B., Malkemper, P., Niemeyer, J., and Zeume, T. (2017). *I Love to Share Knowledge: A personal perspective on academic teaching*. Universität Duisburg-Essen, online: <https://ls1-www.cs.tu-dortmund.de/images/user/logidac/zeume/research/books/AlbrechtBGMNZ-ILoveToShareKnowledge.pdf>. ISBN: 978-3-940402-09-7. *Interview volume on academic teaching*.
- [Aleven et al., 2003] Aleven, V., Stahl, E., Schworm, S., Fischer, F., and Wallace, R. (2003). Help seeking and help design in interactive learning environments. *Review of educational research*, 73(3):277–320.
- [Blackburn et al., 2011] Blackburn, P., van Ditmarsch, H., Manzano, M., and Soler-Toscano, F., editors (2011). *Tools for Teaching Logic - Third International Congress, TICTTL 2011, Salamanca, Spain, June 1-4, 2011. Proceedings*, volume 6680 of *Lecture Notes in Computer Science*. Springer.
- [Ehle et al., 2015] Ehle, A., Hundeshagen, N., and Lange, M. (2015). The sequent calculus trainer - helping students to correctly construct proofs. *CoRR*, abs/1507.03666.
- [Fricke, 2012] Fricke, M. (2012). Software and tutorials for instruction in symbolic logic. <http://softoption.us>. Accessed: 2017-05-12.
- [Gasquet et al., 2011] Gasquet, O., Schwarzentruher, F., and Strecker, M. (2011). Panda: A proof assistant in natural deduction for all. A gentzen style proof assistant for undergraduate students. In [Blackburn et al., 2011], pages 85–92.
- [Huertas, 2011] Huertas, A. (2011). Ten years of computer-based tutors for teaching logic 2000-2010: Lessons learned. In [Blackburn et al., 2011], pages 131–140.
- [Huertas et al., 2011] Huertas, A., Humet, J. M., López, L., and Mor, E. (2011). The SELL project: A learning tool for e-learning logic. In [Blackburn et al., 2011], pages 123–130.
- [Kirsling, 2015] Kirsling, R. (2015). Software and tutorials for instruction in symbolic logic. <http://rkirsling.github.io/modallogic/>. Accessed: 2017-05-12.
- [Lodder and Heeren, 2011] Lodder, J. and Heeren, B. (2011). A teaching tool for proving equivalences between logical formulae. In [Blackburn et al., 2011], pages 154–161.
- [Lodder et al., 2015] Lodder, J., Heeren, B., and Jeurig, J. (2015). A pilot study of the use of logex, lessons learned. *CoRR*, abs/1507.03671.
- [Oblinger et al., 2005] Oblinger, D., Oblinger, J. L., and Lippincott, J. K. (2005). *Educating the net generation*. Boulder, Colo.: EDUCAUSE, c2005. 1 v.(various pagings): illustrations.
- [Schäfer et al., 2013] Schäfer, A., Holz, J., Leonhardt, T., Schroeder, U., Brauner, P., and Ziefle, M. (2013). From boring to scoring - a collaborative serious game for learning and practicing mathematical logic for computer science education. *Computer Science Education*, 23(2):87–111.
- [Shute and Regian, 1993] Shute, V. J. and Regian, J. W. (1993). Principles for evaluating intelligent tutoring systems. *Journal of Interactive Learning Research*, 4(2):245.
- [Sieg, 2007] Sieg, W. (2007). The apros project: Strategic thinking & computational logic. *Logic Journal of the IGPL*, 15(4):359–368.
- [Vanlehn, 2006] Vanlehn, K. (2006). The behavior of tutoring systems. *International journal of artificial intelligence in education*, 16(3):227–265.
- [Velleman, 2010] Velleman, J. D. (2010). blogic: A web logic textbook. [www.nyu.edu/classes/velleman/blogic](http://www.nyu.edu/classes/velleman/blogic). Accessed: 2017-05-12.
- [Wasserman et al., 2012] Wasserman, R., Howard-Snyder, D., and Howard-Snyder, F. (2012). *The Power of Logic*. McGraw-Hill Education.