



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

Fakultät Informatik

Institut für Software- und Multimediatechnik

Arbeitsgruppe Didaktik der Informatik/Lehrerbildung – DIL

**Master-Arbeit**

im Fach Informatik,

konsekutiver Masterstudiengang für das Höhere Lehramt an Gymnasien

**Entwicklung einer Offline-Hilfe für die IDE „Lazarus for  
Education“ auf Grundlage einer didaktischen Analyse von  
Hilfesystemen**

Development of an offline help for the IDE „Lazarus for Education“ based on a  
didactic analysis of help systems

Eingereicht von: Florian Koch

Geburtsdatum: 29.09.1992

Fachsemester: 5.

Fachrichtung: Mathematik/Informatik

Matrikelnummer: 3778758

Betreuer/Erstgutachter: Dr. Holger Rohland

Zweitgutachter: Peter Arnold

Tag der Einreichung: 27.10.2016

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis .....</b>	<b>IV</b>
<b>Tabellenverzeichnis.....</b>	<b>V</b>
<b>1. Einleitung .....</b>	<b>1</b>
1.1 Einführung und Motivation .....	1
1.2 Zielstellung .....	1
1.3 Aufbau der Arbeit.....	2
<b>2. Hilfesysteme.....</b>	<b>3</b>
2.1 Definition und Notwendigkeit von Hilfesystemen.....	3
2.1.1 Definition.....	3
2.1.2 Notwendigkeit .....	5
2.2 Klassifikation von Hilfesystemen .....	9
2.3 Hilfearten .....	13
2.4 Kriterien und Anforderungen für Hilfesysteme .....	17
2.4.1 Kriterienkatalog nach Urbanek.....	17
2.4.2 Anforderungen an Hilfesysteme.....	22
<b>3. Einsatzmöglichkeiten von Hilfesystemen im Informatikunterricht .....</b>	<b>26</b>
3.1 Hilfesysteme als Bestandteil des Lehrplans .....	26
3.2 Einsatzszenarien von Hilfesystemen .....	30

<b>4. Entwurf der Offline-Hilfe für die IDE „Lazarus for Education“ .....</b>	<b>33</b>
4.1 Lazarus und Lazarus for Education .....	33
4.2 Wahl der Klasse, Art und Inhalte der Hilfe .....	37
4.2.1 Tätigkeitsanalyse .....	38
4.2.2 Nutzeranalyse .....	43
4.2.3 Schlussfolgerung .....	48
4.3 Aufbau der Hilfe .....	51
<b>5. Fazit.....</b>	<b>54</b>
<b>Literaturverzeichnis.....</b>	<b>56</b>
<b>Selbstständigkeitserklärung.....</b>	<b>58</b>

## **Abbildungsverzeichnis**

1. Klassifikation von Hilfesystemen nach Bauer .....	11
2. Turnkey Help von Microsoft Word 2007.....	14
3. Dialog-Hilfe von Microsoft Word 2007 .....	14
4. Direkt-Hilfe von Microsoft Word 2007 .....	15
5. eingebettete Hilfeinformationen .....	15
6. Benutzeroberfläche von Lazarus .....	33
7. Hinzufügen von „Lazarus for Education“.....	34
8. Einstellung in „Lazarus for Education“ (am Beispiel von Eigenschaften und Ereignissen.....	35
9. Entwicklungsphasen der Softwaredokumentation .....	37
10. Entworfenes Hilfesystem für Lazarus .....	52

## **Tabellenverzeichnis**

1. Wichtige Bestandteile der minimalen und erweiterten Konfiguration .....	36
2. Folgerungen aus den Betrachtungen und Analysen.....	48

# **1. Einleitung**

## **1.1 Einführung und Motivation**

Die ersten Schritte in einem neuen Programm sind für den Nutzer nicht immer einfach. Funktionalitäten müssen erst erkundet werden, bevor man souverän in der Anwendersoftware agieren kann. Dabei werden die Anwender meist durch sogenannte Hilfesysteme unterstützt, damit Probleme schnell behandelt werden können.

Programmierungsumgebungen zeichnen sich durch eine hohe Komplexität aus, die schon durch deren Oberfläche den Nutzer überfordern können und deswegen eine gute Hilfe benötigen. Im Rahmen der Diplomarbeit „Strukturelle Untersuchung einer IDE mit dem Ziel einer möglichst frei skalierbaren Anpassung der IDE Lazarus“ von Michael Kuhardt im Jahr 2010 ist die Version „Lazarus for Education“ entstanden. Wie der Name suggeriert, handelt es sich dabei um eine Entwicklungsumgebung für den schulischen Kontext. Von Nachteil ist jedoch, dass bisherige Hilfen nur online verfügbar oder in englischer Sprache verfasst sind. Eine angepasste Unterstützung für die entworfene IDE existiert nicht, so dass die Lernenden beim Kontakt mit der Software nicht optimal in der Handhabung des Programms gefördert werden.

## **1.2 Zielstellung**

Das Ziel dieser Master-Arbeit besteht darin, eine Offline-Hilfe für die „IDE Lazarus for Education“ zu entwickeln, die den Nutzer in der Handhabung der Anwendersoftware unterstützt sowie zum Verständnis der zugrundeliegenden Programmiersprache Object Pascal beitragen soll. Dazu wird zunächst eine Analyse von Hilfesystem vorgenommen und die Bedeutung dieser Komponente für den Informatikunterricht betrachtet, um anschließend ein solches System für die spezielle Version der IDE anhand der gewonnenen Erkenntnisse zu implementieren.

### 1.3 Aufbau der Arbeit

Die vorliegende Arbeit ist in drei Schwerpunkte gegliedert. Im Kapitel „Hilfesysteme“ werden diese definiert und deren Notwendigkeit begründet. Es folgt eine Klassifikation der Hilfesysteme, um das Spektrum der Möglichkeiten zu erfassen. Die anschließende Vorstellung der Hilfearten erfolgt unter Einbezug der Klassen und dient der Darstellung von Umsetzungsmöglichkeiten. Das Kapitel endet mit der Betrachtung von verschiedenen Kriterien für und Anforderungen an Hilfesysteme, so dass allgemeine Rahmenbedingungen für den Entwurf einer Hilfe gefunden werden können, die dann bei der zu entwerfenden Offline-Hilfe beachtet werden.

Im nächsten Kapitel wird betrachtet, inwieweit ein Hilfesystem Bestandteil des Informatikunterrichts sein kann. Dazu wird ein ausführlicher Blick in den sächsischen Lehrplan geworfen. Im Anschluss daran werden mögliche Einsatzszenarien der Unterstützung im Unterricht im Zusammenhang mit verschiedenen Methoden formuliert.

Um die Offline-Hilfe für „Lazarus for Education“ zu entwickeln, werden im letzten Kapitel die Besonderheiten der Version analysiert, welche unbedingt in der Hilfe thematisiert werden müssen, sowie Klasse, Art und Inhalt auf Grundlage von Nutzer- und Tätigkeitsanalysen gewählt. Hierbei fließen die Anforderungen und Kriterien des zweiten Kapitels mit ein sowie die besondere Situation des schulischen Kontexts. Am Ende dieses Gliederungspunktes erfolgt dann die Realsierung der Hilfe, indem der Aufbau dargestellt wird. Die Arbeit endet mit einem Fazit zu den gewonnen Erkenntnissen.

## 5. Fazit

Wie aus den Betrachtungen hervorgeht, sind Hilfesysteme ein wichtiger Bestandteil für die Nutzung von Programmen. Ein Nichtvorhandensein dieser Unterstützung kann zu schwerwiegenden Problemen bei der Bearbeitung von Aufgaben führen und die Bewältigung verhindern bzw. heraus zögern. Durch Hilfesysteme werden dem Nutzer der Software wertvolle Information zur Verfügung gestellt, die je nach Art entweder vom Anwender selbst oder vom Programm aufgerufen werden. Die verschiedenen Erscheinungsformen lassen sogar manchmal vermuten, dass diese kein Hilfesystem sind, sondern diese direkt zur Software gehören, zum Beispiel Hinweise auf Syntaxfehler und automatische Korrekturen. Damit wird der Nutzer schon auf Probleme hingewiesen, bevor er diese überhaupt wahrnimmt und dementsprechend handeln kann. Natürlich müssen für einen solchen gewinnbringenden Einsatz die diversen Anforderungen erfüllt sein, um eine Person nicht in der Arbeit zu behindern.

Auch für den Informatikunterricht spielen Hilfesysteme eine wichtige Rolle. Statt eines langwierigen Suchens von möglichen Themen im Lehrplan, in denen man eventuell die Nutzung von Hilfesystemen für das Erreichen von Lernzielen instrumentalisieren könnte, ist die Unterstützungssoftware ein fester Bestandteil des Lehrplans. Die Bedeutung wird außerdem im Verlauf der Klassenstufen schrittweise immer größer. Dies ermöglicht dem Lehrer eine Hilfe als ein wichtiges Medium für Lernprozesse zu nutzen, um wiederum den Unterricht anders zu gestalten und zur selbstständigen Erschließung neuer Programme bei Schülern beizutragen.

Das ursprüngliche Ziel der Arbeit, eine angepasste Hilfestellung für die spezielle Entwicklungsumgebung „Lazarus for Education“ zu entwickeln, wurde zwar im Verlauf der Arbeit aufgrund von Inkompatibilitäten mit der aktuellen Version von Lazarus verworfen, doch es konnte ein Hilfesystem für die Schule entwickelt werden, dass sich an den Lerninhalten des sächsischen Informatikunterricht und an den Erkenntnissen aus der Diplomarbeit von Michael Kuhardt orientiert, um eine gute Handhabung der normalen Version von Lazarus zu ermöglichen. Durch diese Unterstützung und die Führung des Lehrers wird im Unterricht die Überforderung des Lernenden mit der Komplexität der Benutzeroberfläche minimiert sowie die weitere Arbeit mit der IDE gewährleistet, so dass auch keine erneute Anpassung des Schülers bei Verwendung der normalen Version erfolgen muss.

## 5. Fazit

---

Sollte es in Zukunft jedoch dazukommen, dass „Lazarus for Education“ wieder vollständig funktioniert, dann kann im Rahmen der minimalen Konfiguration auf diese Hilfe zurückgegriffen werden und alle sichtbaren Komponenten werden dem Nutzer erklärt. Damit existiert dennoch eine Hilfe, die die Arbeit mit der speziellen Version unterstützen kann.

## Literaturverzeichnis

- [Bau88a] Bauer, J. (1988): Konzepte und Prototypen interaktiver Hilfesysteme, Diss., Universität Stuttgart 1988
- [Bau88b] Bauer, J., Schwab, T.(1988): Anforderungen an Hilfesysteme. In Balzert, H. (Hrsg.): Einführung in die Software-Ergonomie, de Gruyter, Berlin, New York, S. 197-214
- [Dic00] Dick, E. (2000): Multimediale Lernprogramme und telematische Lernarrangements. Einführung in die didaktische Gestaltung, BW Bildung und Wissen Verlag und Software GmbH, Nürnberg
- [Grü13] Grünwied, G. (2013): Software-Dokumentation: Grundlagen – Praxis – Lösungen, 3. Auflage, expert verlag, Renningen
- [Har06] Hartmann, W., Näf, M., Reichert, R. (2006): Informatikunterricht planen und durchführen, Springer-Verlag, Berlin, Heidelberg
- [Her88] Hermann, T. (1988): Probleme bei der Konstruktion und beim Einsatz von Hilfesystemen. In Balzert, H. (Hrsg.): Einführung in die Software-Ergonomie, de Gruyter, Berlin, New York, S. 215-228
- [Her09] Herczeg, M. (2009): Software-Ergonomie. Theorien, Modelle und Kriterien für gebrauchstaugliche interaktive Computersysteme, Oldenbourg Wissenschaftsverlag GmbH, München
- [Kuh10] Kuhardt, M. (2010): Strukturelle Untersuchung einer IDE mit dem Ziel einer möglichst frei skalierbaren Anpassung der IDE Lazarus, Diplomarbeit, Technische Universität Dresden
- [LpGes09] Sächsisches Staatsministerium für Kultus und Sport (Hrsg.) (2009): Lehrplan Gymnasium. Gesellschaftswissenschaftliches Profil, Dresden: Sächsisches Staatsministerium für Kultus und Sport
- [LpInf11] Sächsisches Staatsministerium für Kultus und Sport (Hrsg.) (2011): Lehrplan Gymnasium. Informatik, Dresden: Sächsisches Staatsministerium für Kultus und Sport

- [LpNawi09] Sächsisches Staatsministerium für Kultus und Sport (Hrsg.) (2009): Lehrplan Gymnasium. Naturwissenschaftliches Profil, Dresden: Sächsisches Staatsministerium für Kultus und Sport
- [LpSpo09] Sächsisches Staatsministerium für Kultus und Sport (Hrsg.) (2009): Lehrplan Gymnasium. Sportliches Profil, Dresden: Sächsisches Staatsministerium für Kultus und Sport
- [LpTC09] Sächsisches Staatsministerium für Kultus und Sport (Hrsg.) (2009): Lehrplan Gymnasium. Technik und Computer, Dresden: Sächsisches Staatsministerium für Kultus und Sport
- [Pre15] Preim, B., Dachzelt, R. (2015): Interaktive Systeme Band 2. User Interface Engineering, 3D-Interaktion, Natural User Interfaces, 2. Auflage, Springer Verlag, Berlin, Heidelberg
- [Shn98] Shneiderman, B. (1998): Designing the user interface: strategies for effective human-computer-interaction, 3. Auflage, Addison Wesley Longman, Inc, Amsterdam
- [Thi08] Thiemann, P.(2008): Benutzerfreundliche Online-Hilfen. Grundlagen und Umsetzung mit MadCap Flare, Vieweg + Teubner Verlag|GWV Fachverlage GmbH, Wiesbaden
- [Urb91] Urbanek, W.(1991): Software-Ergonomie und benutzerangemessene Auswahl von Werkzeugen bei der Dialoggestaltung, de Gruyter, Berlin, New York
- [Zie88] Ziegler, J. (1988): Aufgabenanalyse und Funktionsentwurf. In Balzert, H. (Hrsg.): Einführung in die Software-Ergonomie, de Gruyter, Berlin, New York, S. 231-253