

September 2003

# Ein Helpdesk-System zur Bearbeitung von Anliegen Studierender am Beispiel der VGU

Conny Kühne

*Otto-von-Guericke-Universität Magdeburg*

Daniel Pauer

*Otto-von-Guericke-Universität Magdeburg, pauer@iti.cs.uni-magdeburg.de*

Claus Rautenstrauch

*Otto-von-Guericke-Universität Magdeburg, rauten@iti.cs.uni-magdeburg.de*

Follow this and additional works at: <http://aisel.aisnet.org/wi2003>

---

## Recommended Citation

Kühne, Conny; Pauer, Daniel; and Rautenstrauch, Claus, "Ein Helpdesk-System zur Bearbeitung von Anliegen Studierender am Beispiel der VGU" (2003). *Wirtschaftsinformatik Proceedings 2003*. 46.

<http://aisel.aisnet.org/wi2003/46>

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISEL). It has been accepted for inclusion in Wirtschaftsinformatik Proceedings 2003 by an authorized administrator of AIS Electronic Library (AISEL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

In: Uhr, Wolfgang, Esswein, Werner & Schoop, Eric (Hg.) 2003. *Wirtschaftsinformatik 2003: Medien - Märkte - Mobilität*, 2 Bde. Heidelberg: Physica-Verlag

ISBN: 3-7908-0111-9 (Band 1)

ISBN: 3-7908-0116-X (Band 2)

© Physica-Verlag Heidelberg 2003

# Ein Helpdesk-System zur Bearbeitung von Anliegen Studierender am Beispiel der VGU

**Conny Kühne, Daniel Pauer, Claus Rautenstrauch**

Otto-von-Guericke-Universität Magdeburg

*Zusammenfassung: In diesem Beitrag werden Grundlagen, Fachkonzept und Implementierung eines Helpdesk-Systems für die Verwaltung eines Internet-basierten Studiengangs Master of Business Informatics (MBI) der Virtual Global University (VGU) vorgestellt. Aus einer kurzen Darstellung des Szenarios MBI/VGU wird der Einsatz von Helpdesk-Systemen für die Bearbeitung von Anliegen weltweit verteilter Studierender durch Lehrende, die über Zentraleuropa vertreten sind, motiviert. Die Grundlagen von Helpdesks und Szenario-bezogenen Besonderheiten bilden die Basis für das Fachkonzept des Helpdesk-Systems SMILE (Support system for the Management of Internet-based Learning and Education), dessen Darstellung Kern des Beitrags ist. Am Schluss wird die Systemarchitektur vorgestellt.*

*Schlüsselworte: Helpdesk-System, Trouble-Ticket-System, Distance Learning*

## 1 Szenario

Der *International Master of Business Informatics* (MBI) ist der weltweit erste online verfügbare Master-Studiengang Wirtschaftsinformatik. Der englischsprachige Fernstudiengang nutzt Internet- und Multimedia-Technologien. Er ist ein Kooperationsvorhaben der Virtual Global University<sup>1</sup> und der Europa-Universität Viadrina Frankfurt (Oder)<sup>2</sup>, das im Wintersemester 2001/2002 startete und vom Bundesministerium für Bildung und Forschung (BMBF) im Rahmen der Initiative „Neue Medien in der Bildung“ gefördert<sup>3</sup> wird. Das MBI-Programm wird von der *School of Business Informatics* (SBI) getragen, welche die Infrastruktur für die Kurse und die Administration zur Verfügung stellt.

---

<sup>1</sup> <http://www.vg-u.de/>

<sup>2</sup> <http://www.euv-frankfurt-o.de/>

<sup>3</sup> Das diesem Bericht zugrundeliegende Vorhaben wird mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 08NM135C gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

Die Dozenten der einzelnen Kurse sowie deren Assistenten sind geografisch über Zentral-Europa verteilt. Im Wintersemester 2002/2003 lehren 13 Professoren insgesamt 14 Kurse, die über das Internet angeboten werden. Als Web-basierte Lernplattform kommt das System WebCT<sup>4</sup> zum Einsatz. Es dient in erster Linie zur Verwaltung der einzelnen Kurse. Dazu zählt die Zuordnung von Studenten und Dozenten zu den Kursen, die Authentifizierung der Benutzer, die Bereitstellung von Lehrmaterialien und deren Strukturierung in Module. Es beinhaltet ferner Internet-basierte Kommunikationsmittel wie E-Mail und Diskussionsforen.

Die Studierenden der VGU sind über die ganze Welt verteilt und mehrheitlich Teilzeitstudenten. Sie belegen die Kurse in der Regel neben ihrem Beruf und lernen hauptsächlich abends und am Wochenende.

Beim Studieren an der VGU treten wie an Präsenzuniversitäten auch Fragen und Probleme auf. Diese können auf Grund der räumlichen Verteilung und unterschiedlichen Arbeitszeiten von Studierenden und Dozenten ausschließlich elektronisch kommuniziert und beantwortet werden. Bisläng stehen E-Mail, Telefon, WebCT-Mail und WebCT-Diskussionsforen als Kommunikationsmittel zur Auswahl. Die Bearbeitung der Anliegen mit diesen Mitteln weist folgende Schwachstellen auf:

- Die Suche nach dem richtigen Ansprechpartner gestaltet sich für Studierende oftmals zeitaufwändig.
- Die Koordination mehrerer Bearbeiter zur Beantwortung eines Anliegens ist nur bedingt möglich.
- Es besteht die Gefahr der verspäteten oder vergessenen Beantwortung von Anliegen.
- Die Suche über bereits gelöste Fälle ist nicht möglich, da diese nicht an zentraler Stelle archiviert sind. Häufig wiederkehrende Fragen müssen immer wieder beantwortet werden.

Daher entstand die Idee, ein System zu entwickeln, mit dessen Hilfe sich Fragen und Probleme der Studierenden auf effizientere und effektivere Art und Weise als bisher lösen lassen. Als Vorbild für die Entwicklung sollen Helpdesk-Systeme dienen. Diese leisten Mitarbeitern von Unternehmen Unterstützung bei Schwierigkeiten und Problemen insbesondere im PC-Endbenutzer-Bereich. Die verwendeten Konzepte werden auf den neuen Anwendungsbereich übertragen.

---

<sup>4</sup> <http://www.webct.com/>

## 2 Helpdesk- und Trouble-Ticket-Systeme

### 2.1 Helpdesks und Helpdesk-Systeme

Ein *Helpdesk* wird durch folgende Merkmale charakterisiert (vgl. [Woot01, S. 5], [GöRo99, S. 134], [Knap99, S. 1], [ThSt96, S. 37] und [LeLü00, S. 2]):

- Ein Helpdesk erfüllt Unterstützungsfunktionen (Supportfunktionen) für seine Kunden<sup>5</sup>.
- Gegenstand des Supports sind Produkte und Dienstleistungen.
- Der Helpdesk dient als eine zentrale Anlaufstelle für Fragen und Probleme (single point of contact).
- Er stellt eine organisatorische Einheit innerhalb eines Unternehmens dar.
- Ein Helpdesk ist lösungsorientiert. Das Ziel ist die effiziente und effektive Beantwortung von Fragen bzw. Lösung von Problemen.
- Dezentralisierte Helpdesks können über mehrere Standorte verteilt sein, ohne dass dies für Benutzer erkennbar sein muss.

Helpdesk-Organisationen sind mehrstufig aufgebaut. Die erste Stufe bildet der First-Level-Support (auch frontline support oder Hotline). Dieser zeichnet für den direkten Kontakt mit dem Anwender verantwortlich und ist in der Lage, einfache oder wiederkehrende Probleme aufgrund des zentralisierten Wissens sofort zu lösen. Kann der First-Level-Support das Problem nicht beheben, leitet er es an Spezialisten auf der nächsten Ebene (dem Second-Level-Support) weiter. Diese sind höher qualifiziert als die Mitarbeiter des First-Levels und besitzen die nötige methodische und fachliche Kompetenz, um komplexere Probleme lösen zu können. Sie erfüllen meist noch andere Aufgaben innerhalb des Unternehmens und sind im Gegensatz zu den Mitarbeitern des First-Level-Supports nicht Vollzeit für den Helpdesk tätig. Die dritte Ebene des Supports besteht aus technischen Spezialisten. Hier laufen nur noch wenige komplexe Probleme auf. Häufig ist dieser Teil der Supportstruktur in andere Abteilungen (z. B. Entwicklungsabteilung) oder externe Unternehmen ausgelagert.

Als *Helpdesk-System* wird ein softwaretechnisches Subsystem verstanden, das die Funktionen des Helpdesks ganz oder teilweise unterstützt (vgl. [LeLü00, S. 2]).

---

<sup>5</sup> Als *Kunden* eines Helpdesks werden alle Personen bezeichnet, welche die Unterstützung des Helpdesks in Anspruch nehmen.

## 2.2 Trouble-Ticket-Systeme

Eine Unterklasse von Helpdesk-Systemen bilden die *Trouble-Ticket-Systeme* (TTS). Es gibt für dieses Gebiet keine Arbeiten zur Standardisierung des Begriffes (vgl. [San<sup>+</sup>98, S. 1]). Allerdings ähneln sich die Begriffsauffassungen in den verschiedenen Beiträgen sehr stark. Ein Trouble-Ticket-System bildet die Meldungsbearbeitung mit Laufzettel per Software ab. Ein *Trouble-Ticket* fungiert dabei als elektronische Entsprechung des Laufzettels<sup>6</sup>. Aufgetretene Problemfälle werden auf einem Ticket<sup>7</sup> notiert und im System gespeichert. Jeder Bearbeiter vermerkt die zur Lösung des Problems durchgeführten Aktivitäten auf dem Trouble-Ticket und leitet es weiter oder schließt es ab. Dadurch wird der Prozess der Problembearbeitung dokumentiert und die Zusammenarbeit mehrerer Bearbeiter an einem Problem koordiniert. Trouble-Ticket-Systeme dienen häufig zur Unterstützung der Mitarbeiter eines Helpdesks. Deshalb wird der Begriff auch synonym zum Begriff Helpdesk-System gebraucht (vgl. [JaUn98, S. 100ff] und [Scha99, S. 70ff]).

### Anforderungen an ein Trouble-Ticket-System

TTS besitzen eine lange Tradition auf technischen Gebieten wie dem System- und Netzwerkmanagement. Dreo identifiziert und erläutert funktionale Anforderungen an ein TTS und leitet daraus eine Architektur für ein generisches TTS ab (vgl. [Dreo95, S. 14ff]). Ihre Betrachtungen bauen auf dem Fehlermanagement (fault management) auf. Fehlermanagement verfolgt das Ziel, die Verfügbarkeit der verwalteten Ressourcen (z. B. Netzwerkressourcen, verteilte Systeme oder verteilte Applikationen) zu gewährleisten. Die während der Fehlerbehebung durchgeführten Aktivitäten werden auf dem Trouble-Ticket vermerkt. Dadurch wird die Hauptaufgabe des TTS ermöglicht. Diese besteht darin, mehrere Experten bei der Zusammenarbeit zur Lösung eines Problems zu unterstützen. Von dieser Hauptaufgabe leitet Dreo als *Basisanforderungen* an ein TTS die Dokumentation jedes Schrittes während des Prozesses der Fehlerbehebung und die Koordination der dazu erforderlichen Aktivitäten ab (vgl. [Dreo95, S. 14ff]).

Für die Behebung von Fehlern mit komplexen Ursachen ist häufig die Kooperation vieler Experten nötig, die in der Regel nacheinander an der Problemlösung arbeiten. Die Dokumentation der Fehlerbehebung gewährleistet, dass ein Experte die Schritte sehen kann, die andere Bearbeiter vor ihm bereits durchgeführt haben.

Die Koordination der Aktivitäten zur Fehlerbehebung beinhaltet die Entwicklung effizienter Mechanismen, um die Kommunikation zwischen den Personen zu unterstützen, die gemeinsam an der Lösung eines Problems arbeiten. Die Grundlage

---

<sup>6</sup> Eine weitere sehr anschauliche Erklärung liefert Johnson, indem er das Trouble-Ticket mit einem Hospital Chart vergleicht (vgl. [John92, S. 1]).

<sup>7</sup> Ticket wird im Kontext der Trouble-Ticket-Systeme als Synonym für Trouble-Ticket verwendet.

dieser Koordination ist das so genannte *Weiterleiten* (auch Forwarding) von Trouble-Tickets. Wenn ein Problem von einem Bearbeiter nicht gelöst werden kann, leitet dieser es an einen anderen weiter. Eine solche Situation kann z. B. auftreten, wenn der Bearbeiter feststellt, dass der aufgetretene Fehler nicht in sein Anwendungsgebiet fällt. In diesem Fall wird er es an jemanden weiterleiten, der das Problem auf Grund seines spezifischen Fachwissens lösen kann.

### **3 Ein Helpdesk für die VGU – Besonderheiten, Vorteile und Anforderungen**

#### **3.1 Besonderheiten gegenüber herkömmlichen Helpdesks**

Der VGU-Helpdesk SMILE (Support system for Management of Internet-based Learning and Education) ist nach dem Vorbild eines herkömmlichen Helpdesks aufgebaut. Dafür wurden die für Helpdesks bzw. Helpdesk-Systeme verwendeten Konzepte auf den neuen Anwendungsbereich übertragen. Die sich daraus ergebenden Vorteile sind im nächsten Abschnitt beschrieben.

Die Besonderheiten des neuen Anwendungsbereichs werden im Folgenden herausgearbeitet. Als Vergleichsobjekt dient auf Grund der fachlich-konzeptionellen Nähe der typische Helpdesk-Anwendungsbereich Benutzerservice.

#### **Art der Anliegen**

Anliegen, d. h. Fragen und Probleme, sowie deren Beantwortung bzw. Lösung sind Bearbeitungsgegenstand eines jeden Helpdesks. Die Anliegen im Endbenutzerbereich sind Hard- und Softwareprobleme, die sich im Umgang mit PC und (Anwendungs-)Software ergeben. Hierzu zählen ggf. auch Probleme mit der Systemperipherie (Drucker, Scanner, Netzwerke).

Gegenstand des VGU-Helpdesks sind die Anliegen Studierender. Nach den Erfahrungen, die in den ersten Semestern des Lehrbetriebs an der VGU gemacht wurden, gehören zu diesen Anliegen u. a. folgende:

- Fachliche Fragen, die einzelnen Lehrveranstaltungen betreffend
- Fragen zu Terminen
- Fragen zu Prüfungsvoraussetzungen
- Technische Probleme mit den Online-Lehrmaterialien
- Fragen zum Studienablauf

- Fragen zur Studien- bzw. Prüfungsordnung

Aus dieser Aufzählung lässt sich ersehen, dass das Anliegen-Spektrum des VGU-Helpdesks breiter gefasst ist als das eines herkömmlichen, der nur technische Probleme für einen abgegrenzten Bereich (z. B. PCs mit installiertem Windows und dazugehörigem Office-Paket) behandelt. Häufig treten beim PC-Helpdesk wiederkehrende bzw. ähnliche Anliegen auf. Diese können dann auf Grund des gesammelten Wissens schneller gelöst werden.

### **Mitarbeiter des Helpdesks**

Die Mitarbeiter der ersten Stufe eines PC-Helpdesks sind meist Angestellte der IT-Abteilung des Unternehmens, die extra für ihre Aufgabe geschult wurden und sie hauptamtlich wahrnehmen. Auf den nachgelagerten Support-Levels kommen Experten zum Einsatz.

Die Mitarbeiter des VGU-Helpdesks sind Professoren, wissenschaftliche Mitarbeiter, Sekretärinnen und studentische Hilfskräfte. Sie sind nicht für diese Aufgabe geschult und erledigen sie nebenbei.

### **Kunden des Helpdesks**

Die Kunden eines herkömmlichen Helpdesks sind Endbenutzer, die des VGU-Helpdesks dagegen Studenten. Aber auch die Mitarbeiter der VGU, die sich räumlich weit voneinander entfernt befinden, können den Helpdesk zur Klärung ihrer Probleme benutzen. Somit können auch die Mitarbeiter des Helpdesks zu Kunden des selben werden.

### **Geografische Verteilung**

Die Mitarbeiter der ersten Stufe(n) eines Endbenutzer-Helpdesks befinden sich meist in geografischer Nähe zu ihren Kunden, d. h. am Standort der Firma. Selbst wenn diese mehrere Standorte hat, können die Support-Mitarbeiter nach dem Prinzip eines dezentralen Helpdesks über diese Standorte verteilt sein. Im Unterschied dazu sind beim VGU-Helpdesk die Mitarbeiter über Zentraleuropa und die Kunden weltweit verteilt.

## **3.2 Vorteile eines Helpdesks für die VGU**

Im Folgenden werden die Vorteile eines herkömmlichen Helpdesks (vgl. [LeLü00, S. 4ff], [Woot01, S. 11ff] und [ThSt96, S. 55]) dahingehend überprüft, ob sie auch für das neue Anwendungsgebiet zutreffen.



### **Zentrale Wissensquelle**

SMILE ist zentrale Wissensquelle der VGU. Das gesamte gesammelte Wissen zur Verwaltung des MBI-Studiengangs wird hier dokumentiert, archiviert und zentral sowohl Studenten als auch Professoren und Mitarbeitern zugänglich gemacht. Als Ausgangsdaten dienen die Anliegen der Studierenden (und deren Beantwortung), aber auch der Professoren und Mitarbeiter, sowie weitere Informationsquellen wie die Studien- und Prüfungsordnung. Tritt ein Anliegen mehrmals auf, so muss es nur einmal bearbeitet werden. Bei wiederholtem Auftreten kann auf bereits gespeicherte Informationen zurückgegriffen werden. Außerdem steht das Wissen je dem Benutzer rund um die Uhr zur Verfügung.

### **Zentrale Anlaufstelle**

Ein Helpdesk fungiert als *single point of contact*. Er hilft Studierenden und Mitarbeitern bei der effizienten Suche nach den richtigen Ansprechpartnern. Das ist insbesondere dann relevant, wenn fächerübergreifende Fragen oder Fragen zum Studienablauf auftreten. Das Helpdesk-System reduziert die Anzahl der momentan genutzten Kommunikationskanäle (E-Mail, WebCT, Telefon) auf einen. Dadurch werden Redundanzen bei der Beantwortung vermieden und Regeln konsistent auf alle Anliegen angewendet.

### **Verbesserung der Erreichbarkeit der Mitarbeiter**

Das Helpdesk-System ist rund um die Uhr für die Recherche verfügbar. Es kann dazu beitragen, dass Anliegen schneller den richtigen Adressaten zugeordnet werden und dadurch deren Gesamtbearbeitungsdauer verkürzen. Außerdem soll es sicherstellen, dass Anliegen nicht liegen bleiben. All das erhöht die Zufriedenheit der Helpdesk-Kunden (in diesem Falle der Studenten) und trägt somit zur Steigerung des Images der VGU bei.

Allerdings ergibt sich auch ein Nachteil: durch Zwischenschaltung des Helpdesk-Systems als Mittler zwischen Professoren und Studenten ist mit Einbußen beim persönlichen Betreuungsverhältnis zu rechnen.

### **Controlling-Instrument**

Das Helpdesk-System ermöglicht Analysen z. B. über Herkunft und Anzahl der Anliegen, durchschnittliche Bearbeitungs- und Liegezeiten, Häufigkeit der einzelnen Kategorien usw. Daraus können Missstände erkannt und evtl. Verbesserungsmaßnahmen abgeleitet werden. Darüber hinaus stellt die ständige Transparenz dieser Daten einen Anreiz zur zügigen Bearbeitung der Anliegen dar.

### 3.3 Anforderungen an ein Helpdesk-System für die VGU

Im Folgenden werden die Anforderungen an ein Helpdesk-System für die VGU aus den Anforderungen für ein Trouble-Ticket-System im System- und Netzwerkmanagement hergeleitet (vgl. [Dreo95, S. 15ff] und Abschnitt 2.2).

#### Dokumentation der Fehlerbehebung

Für die Unterstützung der Zusammenarbeit ist es wichtig, dass die Bearbeiter sehen, welche Anmerkungen oder Antworten Bearbeiter vor ihnen gegeben haben. Auch Studierende sollen durch die Dokumentation nachvollziehen können, wie lange ein Ticket bei welchem Bearbeiter lag und was dieser auf ihr Anliegen geantwortet hat. Zusätzlicher Aufwand entsteht durch die Dokumentation nicht, da auch ohne Helpdesk-System Fragen schriftlich beantwortet werden.

#### Koordination der Aktivitäten zur Fehlerbehebung

Zur Behebung von Fehlern im Netzwerk- und Systemmanagement ist häufig die Zusammenarbeit mehrerer Experten nötig. Die Unterstützung dieser Zusammenarbeit ist ein primäres Ziel von Trouble-Ticket-Systemen, die in diesem Bereich eingesetzt werden. Auch ein TTS für die VGU soll bei der Koordination der Zusammenarbeit mehrerer Bearbeiter helfen.

Primäres Ziel ist jedoch nicht die Förderung der Zusammenarbeit, sondern die *effiziente Bearbeitung durch möglichst wenige Bearbeiter*. Im Idealfall findet der Student die Antwort auf sein Anliegen durch eine Recherche über bereits gelöste Fälle und eine Bearbeitung kann ganz vermieden werden.

#### Integration

Ein weiteres Ziel der Entwicklung ist die Integration mit der Lernplattform *WebCT*. Anliegen werden in SMILE Kategorien zugeordnet. Die Zuordnung von Kategorien und angebotenen Kursen muss dabei so gestaltet sein, dass die Dozenten der Kurse gleichzeitig auch die Verantwortlichen der dazugehörigen Kategorie sind. Nun ist es möglich, in WebCT an geeigneten Stellen Links zu platzieren, die automatisch auf die richtige Kategorie in SMILE verweisen. Weiterhin ist die Integration mit dem im gleichen Projekt zu entwickelnden *Elektronischen Studentensekretariat und Prüfungsamt (ESPA)* vorgesehen. ESPA soll alle relevanten Informationen über den Studienverlauf, von der Einschreibung über die Prüfungsleistungen bis zur Verleihung des Master-Zeugnisses, verwalten.

### **Sicherheit**

Das TTS sollte sicherstellen, dass autorisierte Personen das Recht haben, Tickets zu erstellen, zu bearbeiten und weiterzuleiten. Zugriffsberechtigungen sollten für verschiedene Benutzergruppen (Studenten und Bearbeiter) vergeben werden können. Weiterführende Sicherheit wie zum Beispiel verschlüsselte Verbindungen sind vorerst nicht nötig, da keine vertraulichen Personendaten übertragen werden. Zudem ist die Möglichkeit einer Recherche über in der Fallbasis vorhandene Anliegen durch alle Personen erwünscht. Eine Ausnahme bilden vertrauliche Tickets. Hier muss sichergestellt werden, dass diese nur von berechtigten Personen (dem Ersteller und den direkten beteiligten Bearbeitern) eingesehen werden können.

### **Statistiken und Berichte**

Oben wurden die Vorteile beschrieben, die der Einsatz des Helpdesks als Controlling-Instrument bringen kann. Dazu ist es notwendig, Statistiken und Berichte über bestimmte Kennzahlen zu erstellen. Die Komponente zur Erstellung von Statistiken kann Teil des Helpdesk-Systems oder ein externes Werkzeug sein.

### **Intelligente Suchmechanismen**

Das zu erstellende System soll die Möglichkeit bieten, Recherchen über bereits bearbeitete Anliegen durchzuführen. Dies stellt einen wesentlichen Vorteil gegenüber der jetzigen Situation dar, in der eine Suche nach gleichen oder ähnlichen Anliegen nicht durchführbar ist. Dadurch wird eine Mehrfachbeantwortung vermieden, sowohl die Studenten als auch die Mitarbeiter können Aufwand und Zeit sparen. Mit der geplanten Suchfunktion soll es möglich sein, flexibel nach einzelnen Wörtern, ganzen Wortgruppen oder mit booleschen Operatoren verknüpften Begriffen zu suchen. Weiterhin ist geplant, mit Hilfe von aus dem *Case-Based Reasoning* bekannten Methoden auch eine Ähnlichkeitssuche zu implementieren.

## **4 SMILE – der VGU-Helpdesk**

Ausgehend von den vorher beschriebenen Anforderungen und Besonderheiten wurden verschiedene Modelle für SMILE erstellt. Die wichtigsten davon werden im Folgenden vorgestellt. Im Anschluss daran werden einige wichtige Konzepte wie die Recherche über bereits vorhandene Anliegen sowie die Verteilung und Eskalation der Trouble-Tickets erläutert.

## 4.1 Prozessmodell

Das Makro-Prozessmodell in Abbildung 1 beschreibt den groben Ablauf der Bearbeitung eines Anliegens. Auf Grund der Übersichtlichkeit wurde es für den vorliegenden Artikel nicht weiter detailliert. So sind z. B. in der Funktion „Beantworte Trouble-Ticket“ Funktionen für das Weiterleiten und Neueinordnen von Anliegen zusammengefasst.

Die Kunden des Helpdesk-Systems können neben den Studierenden auch die Dozenten und Mitarbeiter sein. Bereits zu Beginn des Prozesses hat der Kunde die Möglichkeit, nach ähnlichen, bereits dokumentierten Anliegen und deren Lösung zu suchen. Ist die Recherche erfolgreich, kann der Prozess hier abgebrochen werden, ohne dass jemand das Anliegen bearbeiten musste. Diese Situation stellt den Idealfall dar.

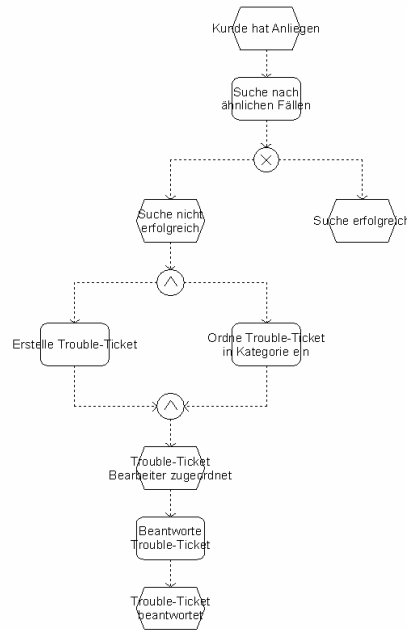


Abbildung 1: VGU Helpdesk-System – Makro-Prozessmodell

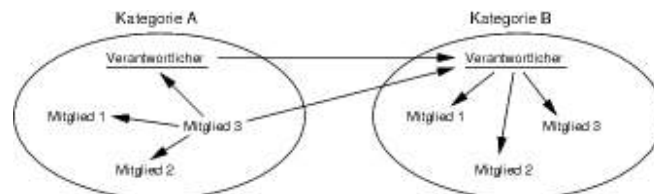


Abbildung 2: Möglichkeiten der Weiterleitung von Trouble-Tickets

Um den richtigen Bearbeiter für ein Anliegen bestimmen zu können, wird es in eine Kategorie eingeordnet, der wiederum ein Verantwortlicher zugeteilt ist. Nachdem das Ticket erstellt und über die Kategorie einem Bearbeiter zugeordnet wurde, beantwortet er es. Der Bearbeiter hat außerdem die Möglichkeit, das Trouble-Ticket weiterzuleiten. Dies ist beispielsweise erforderlich, wenn die Einordnung des Tickets in die Kategorie vom Kunden falsch vorgenommen wurde. Weiterhin ist eine Weiterleitung notwendig, wenn die Beantwortung eines Anliegens die Mitarbeit mehrerer Personen erfordert. Sie kann wahlweise an eine Kategorie oder an einen bestimmten Mitarbeiter der eigenen Kategorie erfolgen. Dieser Zusammenhang ist in Abbildung 2 dargestellt.

Während des gesamten Bearbeitungsprozesses werden Informationen wie Absender, Bearbeiter, Problemstellung, Lösungsweg und Lösung des Anliegens gespeichert. Ein Anliegen gilt als dokumentiert und recherchierbar, wenn es sich im Zustand *abgeschlossen* befindet, d. h. wenn es beantwortet wurde und der Kunde mit dieser Antwort zufrieden ist. Ausnahme: wenn das Anliegen vom Kunden oder Bearbeiter als vertraulich gekennzeichnet wurde, erscheint es nicht in den Suchergebnissen.

## 4.2 Fachklassenmodell

Abbildung 3 zeigt die identifizierten Fachklassen. Kunden, Bearbeiter und Administratoren des Systems werden durch die Klasse User abgebildet. Sie enthält allgemeine Attribute des Benutzers (Name, Vorname, E-Mail), Informationen zu Authentifizierung sowie zur Abwesenheit des Benutzers. Jeder Benutzer kann Trouble-Tickets erstellen.

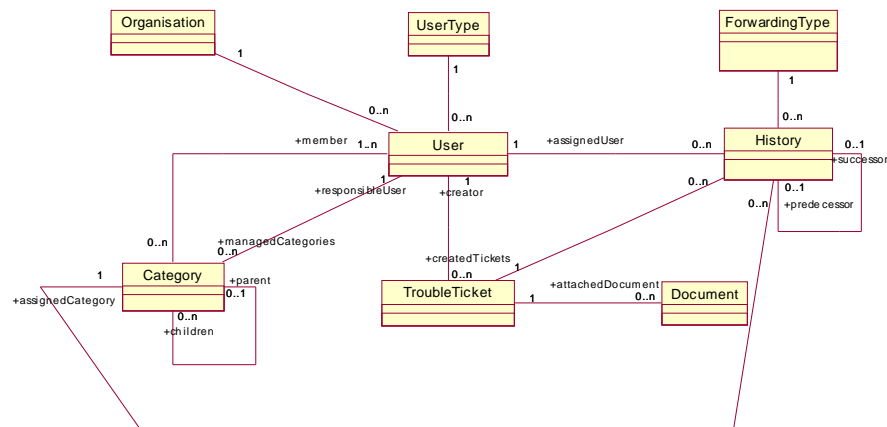


Abbildung 3: VGU Helpdesk-System – Fachklassenmodell

Die Klasse `History` bildet den Verlauf der Bearbeitung eines Trouble-Tickets ab. Sie enthält Informationen darüber, welcher Benutzer wann welches Ticket zu bearbeiten hatte und in welche Kategorie es zum jeweiligen Zeitpunkt eingeordnet war. `History`-Objekte sind durch eine Vorgänger-Nachfolger-Beziehung miteinander verbunden. Dadurch existiert für jedes Trouble-Ticket eine verkettete Liste von Bearbeitungsschritten.

Außerdem enthält das letzte `History`-Objekt der Liste den aktuellen Stand der Ticket-Bearbeitung, d. h. bei welchem Benutzer sich das Trouble-Ticket gerade befindet. Weiterhin wird durch die `History`-Klasse sichergestellt, dass ein Trouble-Ticket zu einem gegebenen Zeitpunkt jeweils nur genau einem Benutzer und genau einer Kategorie zugeordnet ist. Bei jeder Erstellung, Weiterleitung, Beantwortung und Eskalation eines Trouble-Tickets wird ein neues `History`-Objekt angelegt. Welche von den eben genannten Weiterleitungsarten zum Einsatz kommt, wird durch die Klasse `ForwardingType` festgelegt.

Die Kategoriehierarchie wird durch die Klasse `Category` und deren rekursive Beziehung dargestellt. Die Kardinalitäten zeigen an, dass jede Kategorie die Oberkategorie von gar keiner bis beliebig vielen anderen Kategorien sein, selbst aber maximal eine Oberkategorie haben kann. Für jede Kategorie gibt es einen verantwortlichen Benutzer, zusätzlich können noch anderen Benutzer Mitglieder einer Kategorie sein. An ein Trouble-Ticket können ein oder mehrere Dokumente angehängt werden.

### 4.3 Recherche über vorhandene Anliegen

Die Suche nach bereits bestehenden Anliegen soll auf mehreren Wegen möglich sein. Die erste ist die exakte Stichwortsuche. Dabei können Suchbegriffe mit Booleschen Operatoren wie *UND*, *ODER* und *NICHT* verbunden werden oder nach einer kompletten Phrase gesucht werden. Die zweite Methode ist die Ähnlichkeitssuche. Hier erhält man außer den exakten auch ähnliche Ergebnisse. Die dritte Möglichkeit besteht darin, sich durch die Baum-Struktur der Kategorien zu navigieren und direkt nach einem Anliegen zu suchen. Voraussetzung dafür ist, dass man zumindest ungefähr weiß, wo ein Problem eingeordnet werden könnte. Allerdings kann diese Methode bei sehr vielen vorhandenen Anliegen unübersichtlich werden.

### 4.4 Verteilung der Anliegen

Das System bestimmt den Bearbeiter eines Anliegens anhand der Kategorie, in die es eingeordnet wurde. Zu jeder Kategorie existiert genau ein Verantwortlicher. Diese anonyme Verteilung hat gegenüber der personengebundenen den Vorteil, dass der Kunde nicht herausfinden muss, wer für sein Anliegen zuständig ist. Wenn er sich einmal nicht schlüssig ist, z. B. bei fächerübergreifenden Fragen,

ordnet er sein Problem weiter oben in eine allgemeinere Kategorie ein. Ein weiterer Vorteil besteht darin, dass der Kunde nicht auf die Anwesenheit eines bestimmten Mitarbeiters angewiesen ist. Im Falle der Abwesenheit eines Bearbeiters übernehmen andere Mitglieder der Kategorie die Bearbeitung des Anliegens.

#### **4.5 Eskalation und Pooling**

Wie bereits oben erläutert, wird ein neues Anliegen immer zuerst dem Verantwortlichen einer Kategorie zugeordnet. Ist dieser nicht anwesend oder wurde die maximale Bearbeitungsdauer überschritten, so wird das Anliegen in einen Pool eingefügt, d. h. alle Mitarbeiter der gleichen Kategorie können darauf zugreifen. Fühlt sich jemand dafür zuständig, so übernimmt er das Anliegen, entfernt es damit aus dem Pool und bearbeitet es. Bleibt das Anliegen länger als eine vorher festgelegte Zeit im Pool, so wird es automatisch eskaliert, d. h. es wird in die Oberkategorie der aktuellen Kategorie eingeordnet. Dort bekommt es wiederum der Kategorieverantwortliche. Bei jeder Eskalation des Anliegens wird eine E-Mail-Benachrichtigung an die betreffenden Bearbeiter verschickt. Befindet sich das Anliegen bereits in der obersten Kategorie (Wurzelkategorie), so verbleibt es im Pool und die Kategoriemitglieder bekommen in einem bestimmten Zeitintervall Erinnerungsmails.

Der oben beschriebene Eskalationsmechanismus wird deaktiviert, sobald der Bearbeiter das Trouble-Ticket angesehen hat. Danach verbleibt es bei ihm. Bei Zeitüberschreitung bekommt er in einem bestimmten Zeitintervall Erinnerungsmails, bis er sich der Sache annimmt.

Wenn ein Kunde ein Anliegen zurücksendet, weil er mit der Antwort nicht zufrieden ist, so wird es wieder dem gleichen Bearbeiter zugeordnet. Ist dieser jedoch abwesend oder bearbeitet das Anliegen nicht, so wird die Eskalation genau wie oben beschrieben angewendet.

Aber auch mit diesen Maßnahmen kann man eine rechtzeitige Bearbeitung nicht sicherstellen. Die Gefahr der Verzögerung und Nichtbearbeitung besteht generell und ist organisatorischer Natur. Das System kann dieses Problem nicht beheben, sondern nur bei dessen Lösung unterstützen. Da es die Liegezeiten dokumentiert, kann es zur Analyse der Missstände und als Argumentationshilfe herangezogen werden. Beispielsweise ist es denkbar, eine Top-Ten-Liste der Professoren und Mitarbeiter mit den längsten Antwortzeiten oder der höchsten Nichtbeantwortungsquote zu erstellen.

#### **4.6 E-Mail-Benachrichtigung**

Wie bereits erwähnt, verschickt das System E-Mail-Benachrichtigungen an die Benutzer. Dies geschieht immer dann, wenn ein Anliegen erstellt, weitergeleitet,

beantwortet oder eskaliert wurde. Im Falle der Erstellung und Weiterleitung wird der entsprechende Bearbeiter informiert, bei der Beantwortung eines Anliegens bekommt der Kunde eine E-Mail. Wird ein Anliegen in den Pool eskaliert, so erhalten alle Mitglieder der betreffenden Kategorie eine Benachrichtigung.

## 5 Realisierung

In Abbildung 4 ist die Architektur des entwickelten Systems abgebildet. Zur Realisierung des Systems wurde die Java-2-Plattform Enterprise Edition (J2EE) ausgewählt. Diese basiert auf der Programmiersprache Java und besitzt demzufolge deren Eigenschaften wie Objektorientiertheit, Sicherheit, Robustheit, Portabilität oder Multithreading-Fähigkeit. Die J2EE repräsentiert einen Standard zur Entwicklung und zum Einsatz mehrschichtiger, unternehmensweiter Anwendungen. Die Plattform vereinfacht deren Entwicklung durch die Bereitstellung von Basisdiensten wie Transaktionsverwaltung, Sicherheitsmechanismen und Namensdiensten und dem Einsatz standardisierter modularer Komponenten wie Enterprise JavaBeans und Java Servlets.

### 5.1 Geschäftsschicht

Für die Implementierung der Geschäftsschicht wurde die *Enterprise JavaBeans*-Architektur (EJB) in der Version 2.0 verwendet.

Enterprise JavaBeans ist eine Architektur zur Entwicklung komponentenbasierter, verteilter Anwendungen (vgl. [EJB02, S. 41]). Zentraler Bestandteil der Architektur sind Enterprise-Beans. Dabei handelt es sich um serverseitige Software-Komponenten, welche die von den Client-Programmen genutzte Anwendungslogik enthalten (vgl. [DePe00, S. 21]).

Enterprise-Beans nutzen EJB-Container als Laufzeitumgebung. Über den EJB-Container haben Enterprise-Beans Zugriff auf diverse APIs der J2EE. Außerdem werden vom Container Dienste wie Transaktions-Verwaltung, Instanzen-Pooling, Lebenszyklus- oder Persistenzverwaltung erbracht. Der Komponenten-Entwickler wird von der Implementierung dieser Systemdienste entlastet und kann sich voll auf die Umsetzung der Geschäftslogik konzentrieren. Die Container-Dienste können deklarativ über den so genannten Deployment-Deskriptor parametrisiert werden (vgl. [DePe00, S. 24ff] und [Sch<sup>+</sup>02, S. 123]).

Die Fachklassen (vgl. Abbildung 3) sind durch Entity-Beans umgesetzt. Entity-Beans stellen eine objektorientierte Sicht auf die Anwendungsdaten dar, die in einem Speicher persistent gehalten werden. Entity-Beans werden durch einen Primärschlüssel eindeutig identifiziert und erlauben den gleichzeitigen Zugriff durch mehrere Clients. Um die Konsistenz der Daten zu sichern, können Entity-Beans an



Transaktionen teilnehmen. Als Speicher dient im Falle des VGU-Helpdesk-Systems ein Relationales Datenbankmanagementsystem (RDBMS). Die Verwaltung der persistenten Attribute einer Entity-Bean übernimmt der Container. Dieser generiert die Zugriffsmethoden anhand des sog. Persistenzschemas, das im Deployment-Deskriptor definiert ist. Dies wird als Container Managed Persistence (CMP) bezeichnet. Ebenso wird mit Beziehungen zwischen Entity-Beans, den Container Managed Relations (CMR), verfahren.

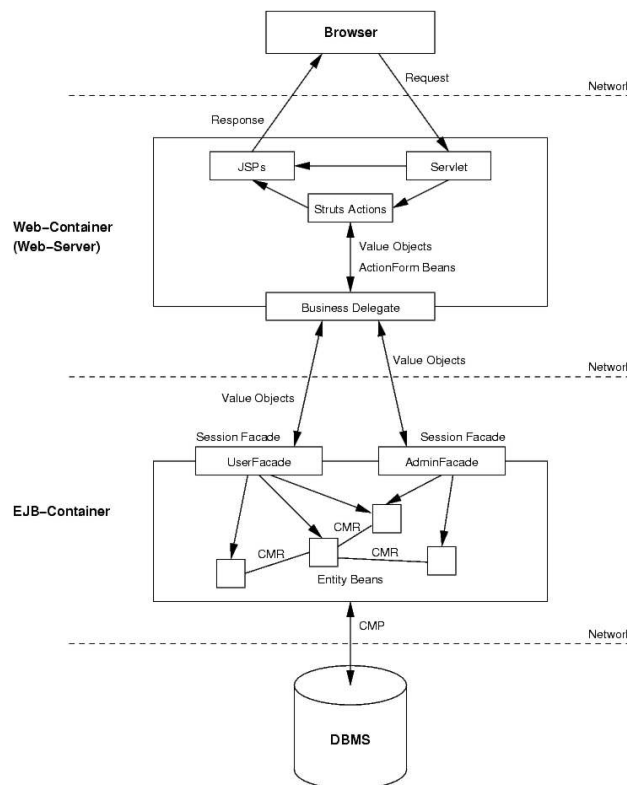


Abbildung 4: Systemarchitektur

Für die Umsetzung der Geschäftslogik werden Session-Beans verwendet. Session-Beans sind private Ressourcen, d. h. sie stehen exklusiv einem Client zur Verfügung. Im Unterschied zu Entity-Beans sind sie anonym und besitzen keinen eindeutigen Schlüssel. Für das SMILE werden Session-Beans nach dem Fassade-Muster benutzt. Demnach greifen Clients nicht direkt auf die verschiedenen Entity-Beans zu, sondern benutzen ausschließlich die Session-Fassade als Schnittstelle. Diese enthält für jeden Anwendungsfall eine Methode, die von den Clients der Präsentationsschicht aufgerufen werden kann. Sämtliche Zugriffe auf die am Anwendungsfall beteiligten Entity-Beans werden durch die Session-Fassade getätigt.

Außerdem werden Transaktionen von den Methoden der Session-Fassade initiiert. Das bewirkt die Klammerung jedes Anwendungsfalls durch eine Transaktion.

Für den Austausch von Daten mit der Präsentationsschicht werden Value Objects verwendet. Dabei handelt es sich um einfache Java Beans, die Daten einer Entity-Bean kapseln und dem Client einen einfachen Zugriff auf ihre Attribute eröffnen.

Ein Business-Delegate wird eingesetzt, um die Klassen der Präsentationsschicht von der Session-Fassade zu entkoppeln, und dadurch die komplexen Vorgänge beim Umgang mit der EJB-Schicht vor den Präsentationsklassen zu verbergen.

## 5.2 Präsentationsschicht

Zur Implementierung der Präsentationsschicht wurde das Open-Source-Framework *Struts* der Apache Software Foundation<sup>8</sup> verwendet. Struts ist eine Umsetzung des Model-View-Controller-Paradigmas (MVC). Es wird durch drei Objekte umgesetzt. Das Model-Objekt enthält und verwaltet die Anwendungsdaten. Das View-Objekt ist für die grafische Darstellung des Model-Objektes zuständig. Zu einem Model können mehrere Views existieren. Das Controller-Objekt reagiert auf Benutzereingaben und steuert den Programmfluss. Durch die Trennung von Anwendungslogik, -daten und Präsentation wird die Flexibilität und Wiederverwendbarkeit erhöht (vgl. [Gam<sup>+</sup>96, S. 5 f.]).

Innerhalb von Struts übernimmt ein Servlet die Aufgabe des Controllers. Als Views kommen JavaServer Pages (JSPs) zum Einsatz. Das Model wird durch JavaBeans repräsentiert.

Das Controller-Servlet nimmt die Requests der Clients entgegen, wertet sie aus und leitet sie an die entsprechende Action-Klasse weiter. Die Action-Klassen können als Erweiterungen des Controllers betrachtet werden, welche die anwendungsspezifische Ablauflogik enthalten. Das Controller-Servlet wird vom Framework bereitgestellt, braucht also nicht selbst programmiert zu werden. Die Action-Klassen benutzen die Dienste der Geschäftsschicht, um Daten des Models zu aktualisieren oder auszulesen. Danach wird die Kontrolle wieder an das Servlet übergeben, das die passende JSP zur Darstellung der Ergebnisse bestimmt. Die Daten der Geschäftsschicht werden dabei wie oben beschrieben in Value Objects gekapselt. Die in einer HTML-Form vom Benutzer eingegebenen Daten werden in sog. ActionForm-Beans zusammengefasst, die von Servlet anhand der Parameter des Requests automatisch gefüllt werden.

Die Konfigurationseinstellungen für das Controller-Servlet werden in einer XML-Datei festgelegt. Sie beinhaltet u. a. Informationen darüber, welche ActionForm-

---

<sup>8</sup> <http://jakarta.apache.org/struts/>

Beans abhängig zu einer bestimmten Action gehören und welche JSPs zur Darstellung der Ergebnisse und Fehlermeldungen aufzurufen sind.

## 6 Zusammenfassung und Ausblick

Im vorliegenden Beitrag wurde die Entwicklung eines Helpdesk-Systems zur Bearbeitung von Anliegen Studierender beschrieben. Dazu wurden die Konzepte herkömmlicher Helpdesks bzw. Helpdesk-Systeme auf den neuen Anwendungsbereich übertragen und mit Hilfe formaler Modelle innerhalb eines Fachkonzeptes spezifiziert. Abschließend erfolgte eine Darstellung der Realisierung einschließlich der Systemarchitektur.

Der Einsatz des Systems SMILE an der VGU ist für Juli 2003 geplant. Daher liegen zum gegenwärtigen Zeitpunkt (Stand 15. Mai 2003) noch keine Erfahrungsberichte vor.

Bei der Entwicklung wurde auf eine möglichst generelle Einsetzbarkeit des Systems an Hochschulen geachtet. Der Aufwand für dessen Anpassung an die Gegebenheiten einer Präsenzuniversität wird als gering eingeschätzt. So soll SMILE auch die Studenten der Fakultät für Informatik an der Otto-von-Guericke Universität Magdeburg bei der Lösung von Problemen unterstützen.

## Literatur

- [DePe00] Denninger, S.; Peters, I.: Enterprise JavaBeans. Addison-Wesley-Verlag: München et. al., 2000.
- [Dreo95] Dreo, G.: A Framework for Supporting Fault Diagnosis in Integrated Network and Systems Management. Shaker: Aachen, 1995.
- [EJB02] Enterprise JavaBeans Specification – Version 2.0, November 2002.  
<http://java.sun.com/products/ejb/docs.html#specs>.
- [Gam<sup>+</sup>96] Gamma et. al.: Entwurfsmuster: Elemente wiederverwendbarer objektorientierter Software. Addison-Wesley-Longman, Bonn et. al., 1996.
- [GöRo99] Göker, M.; Roth-Berghofer, T.: Development and Utilization of a Case-Based Help-Desk Support System in a Corporate Environment. In: Case-Based Reasoning Research and Development, Third International Conference, ICCBR-99, 1999.
- [JaUn98] Jagodic, J.; Ungerer, B.: Hilfsarbeiter – Trouble-Ticket-Systeme. iX Magazin für professionelle Informationstechnik, 1998(5): S. 100–105.
- [John92] Johnson, D.: NOC Internal Integrated Trouble Ticket System – Functional Specification Wishlist, 1992. <http://www.faqs.org/rfcs/rfc1297.html>.

- 
- [Knap99] Knapp, D.: A Guide to Help Desk Concepts. Course Technology: Cambridge et. al., 1999.
- [LeLü00] Lehner, F.; Lüders, R.: Helpdesk-Systeme und Call-Center-Anwendungen. Forschungsbericht 33, Lehrstuhl für Wirtschaftsinformatik III, Universität Regensburg, Juni 2000.
- [San<sup>+</sup>98] Santos, L.; Costa, P.; Simões P.: NetTrouble: A TTS for Network Management. In: IEEE International Telecommunications Symposium 1998, August 1998.
- [Scha99] Schade, O.: Problemsortierer – Freie Trouble-Ticket-Systeme. iX Magazin für professionelle Informationstechnik, 1999(9): S. 70–75.
- [Sch<sup>+</sup>02] Schmietendorf, A.; Dimitrov, E.; Dumke R.: Enterprise JavaBeans. Mitp-Verlag: Bonn, 2002.
- [ThSt96] Thomas, A. H.; Steele, R. M.: The Virtual Help Desk: Strategic Management Center. International Thomson Computer Press: New York et. al., 1996.
- [Woot01] Wooten, B.: Building & Managing a World Class IT Help Desk. Osborne/McGraw-Hill: Berkeley, 2001.