



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Masterarbeit

Konrad Glugla

Campus-Brett - Ein System zur Unterstützung  
sozialer Interaktion an Hochschulen

Konrad Glugla  
Campus-Brett - Ein System zur Unterstützung  
sozialer Interaktion an Hochschulen

Masterarbeit eingereicht im Rahmen der Masterprüfung  
im Studiengang Informatik (Master of Science)  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. rer. nat. Kai von Luck  
Zweitgutachter : Prof. Dr. rer. nat. Gunter Klemke

Abgegeben am 23. Dezember 2010

**Konrad Glugla**

**Thema der Masterarbeit**

Campus-Brett - Ein System zur Unterstützung sozialer Interaktion an Hochschulen

**Stichworte**

Enterprise 2.0, Social Computing, Schwarzes Brett, University 2.0, interaktive öffentliche Bildschirme, soziale Wahrnehmung

**Kurzzusammenfassung**

In dieser Arbeit wird der Einsatz von Social Computing Konzepten an Hochschulen untersucht. Der Trend von Web 2.0 und die Übertragung dieses Trends auf das Unternehmensumfeld, bezeichnet als Enterprise 2.0, haben gezeigt, wie der Aufbau und der Einsatz von sozialen Beziehungen positive Effekte bei der Kollaboration, der Innovation und des Knowledge Sharing erzielen können. Das Umfeld der Hochschule ist aufgrund ihrer stark kommunikativen Umgebung und ihrer Funktion als Bildungs- und Forschungsstätte offensichtlich für die Integration von Social Computing Methoden geeignet, diese sind jedoch aktuell auf den Campus der meisten Hochschulen kaum präsent.

Der Einsatz einer in den sozialen Kontext integrierten Austauschplattform soll soziale Interaktionen auf dem Campus unterstützen und zu einer engeren Zusammenarbeit zwischen Studenten einer Hochschule verhelfen. Dieses als Campus-Brett bezeichnete System, welches sich in seiner Anwendung stark am Schwarzen Brett orientiert, wird in dieser Arbeit entworfen und realisiert.

**Konrad Glugla**

**Title of the paper**

Campus Board - a system for supporting social interaction at universities

**Keywords**

Enterprise 2.0, social computing, bulletin board, University 2.0, interactive public displays, social awareness

**Abstract**

By means of this thesis the use of social computing concepts at universities will be examined. The trend of Web 2.0 and the transfer of this trend to the business environment, referred to as Enterprise 2.0, have shown positive effects on collaborative work, innovation and knowledge sharing through the use of social relationships. The university campus is apparently suitable for adapting social computing methods, due to their highly communicative environment and its function as an educational and research institution. Nevertheless, they are barely present on most university campuses.

Using an exchange platform, which is integrated in the social context, social interactions and cooperative work between students at the campus shall be increased. This system, called Campus Board, is based on the application of a bulletin board and will be designed and implemented in this work.

# Inhaltsverzeichnis

<b>Tabellenverzeichnis</b>	<b>6</b>
<b>Abbildungsverzeichnis</b>	<b>7</b>
<b>1. Einleitung</b>	<b>8</b>
1.1. Ziel dieser Arbeit . . . . .	9
1.2. Aufbau der Arbeit . . . . .	9
<b>2. Grundlagen</b>	<b>11</b>
2.1. Computer Supported Cooperative Work (CSCW) . . . . .	11
2.2. Enterprise 2.0 . . . . .	12
2.3. Liferay Portal . . . . .	13
<b>3. Analyse</b>	<b>15</b>
3.1. Informationspräsenz an Hochschulen . . . . .	16
3.1.1. Hochschulwebseiten . . . . .	16
3.1.2. HAW-Mailer . . . . .	18
3.1.3. Das Schwarze Brett . . . . .	18
3.1.4. Interaktives Infoboard im Department Informatik der HAW Hamburg . . . . .	25
3.2. University 2.0 . . . . .	27
3.3. Aktuelle Forschungsarbeiten / -ergebnisse . . . . .	31
3.3.1. CommunityMirrors . . . . .	31
3.3.2. Google Wave . . . . .	33
3.3.3. Sunspace . . . . .	34
3.3.4. The MobiSoC Middleware for Mobile Social Computing . . . . .	36
3.3.5. Weitere verwandte Projekte . . . . .	39
3.4. Fazit . . . . .	40
<b>4. Systemskizze</b>	<b>41</b>
4.1. Campus-Brett-Szenarien in Hochschulen . . . . .	43
4.1.1. Informationen für unterwegs . . . . .	43
4.1.2. Personalisierter Content . . . . .	43
4.1.3. Notification bei Aktualisierungen . . . . .	44

---

4.1.4. Erstellen von Inhalten außerhalb der Hochschule . . . . .	45
4.2. Sichtbarkeit der Inhalte . . . . .	45
4.3. Die 6 Komponenten von Enterprise 2.0 Technologien . . . . .	46
4.4. Taxonomie . . . . .	48
4.5. Interaktionen mit dem Campus-Brett . . . . .	50
4.5.1. Authentifikation am Campus-Brett . . . . .	50
4.5.2. Die Startseite . . . . .	51
4.5.3. Die Detailansicht . . . . .	53
4.5.4. Unterstützte Anzeigenerstellung . . . . .	55
4.6. Belohnungssystem . . . . .	57
4.7. Public vs Private . . . . .	57
4.8. Übersicht der Anforderungen und Systemwahl . . . . .	58
<b>5. Systementwurf und Realisierung</b>	<b>60</b>
5.1. Model View Controller Architektur . . . . .	60
5.2. Das Datenmodell . . . . .	61
5.2.1. Sortierung / Wert der Anzeigen . . . . .	62
5.3. Umsetzung als Struts Portlet . . . . .	65
5.3.1. ServiceBuilder zur Beschreibung des Datenmodells . . . . .	65
5.3.2. Layout und Seitenfluss . . . . .	68
5.3.3. Inhaltsvorschläge . . . . .	71
5.4. Fazit . . . . .	75
<b>6. Zusammenfassung und Ausblick</b>	<b>77</b>
<b>A. Quellcodes</b>	<b>80</b>
<b>B. Abbildungen</b>	<b>85</b>
<b>Literaturverzeichnis</b>	<b>87</b>

# Tabellenverzeichnis

3.1. Themen am Schwarzen Brett mit ihrer geschätzten Aufteilung . . . . .	24
4.1. Taxonomie im Campus-Brett . . . . .	49
5.1. Beispielhafter Ausschnitt aus Mapping-Tabelle CBCategories_CBEntries	74

# Abbildungsverzeichnis

2.1. Liferay Architekturskizze (Quelle: <a href="#">comundus (2010)</a> ) . . . . .	14
3.1. Abreißzettel an Anzeigen erleichtern die Weitergabe von Kontaktdaten . . . . .	19
3.2. Das Schwarze Brett an der Universität Hamburg . . . . .	20
3.3. Merkmale verschiedener Anzeigen am schwarzen Brett . . . . .	22
3.4. Angebotene Inhalte am Infoboard . . . . .	26
3.5. Das Infoboard im AI-Labor der HAW Hamburg . . . . .	26
3.6. Visualisierung der Verbindungen in einem sozialen Netzwerk (Quelle: <a href="#">Kreßner (2010)</a> ) . . . . .	29
3.7. Vision der CommunityMirrors. Personen interagieren gemeinsam am Wandbilschirm und arbeiten kollaborierend mit den Informationen (Quelle: <a href="#">Koch (2010)</a> ) . . . . .	32
3.8. Übersichtsbildschirm des Google Wave Client. Geöffnet ist ein Thema, das ein Voting beinhaltet. (Quelle: <a href="#">Siegler (2009-05-28)</a> ) . . . . .	34
3.9. Community Equity Model von Sunspace. Aktivitäten werden belohnt mit Punkten, wodurch sich ein Ranking ergibt. Die Top 10-User werden hervorgehoben. (Quelle: <a href="#">Reiser (2009b)</a> ) . . . . .	35
3.10. „Social State“ einer Community (Quelle: <a href="#">Borcea u. a. (2007)</a> ) . . . . .	37
4.1. Content To Go. Inhalte können vom Campus-Brett auf mobile Geräte übertragen werden . . . . .	43
4.2. Inhalte werden durch Personenprofile gefiltert . . . . .	44
4.3. Veränderungen an Anzeigen können nachverfolgt werden und z.B. per Push Notification auf Smartphones erscheinen . . . . .	44
4.4. Campus-Brett-Startseite . . . . .	51
4.5. Detailansicht einer Anzeige im Campus-Brett . . . . .	54
4.6. Skizze zur Funktionsweise beim Erstellen von Anzeigen . . . . .	56
5.1. Model View Controller-Konzept im Campus-Brett . . . . .	60
5.2. Entity Relationship Modell des Campus-Bretts . . . . .	61
5.3. UML Klassendiagramm für Personen und Anzeigen im Campus-Brett . . . . .	62
5.4. Modell-, Service- und Persistenzschicht des Campus-Bretts . . . . .	66
5.5. JSP-Seiten für die Darstellung der Inhalte . . . . .	68



---

5.6. Sequenzdiagramm für das Bearbeiten und Speichern einer Anzeige . . . . .	72
B.1. Screenshot der Campus-Brett Hauptseite mit Aufteilung der JSP-Seiten . . . .	85
B.2. Screenshot der Campus-Brett Detailansichtsseite mit Aufteilung der JSP-Seiten	86
B.3. Screenshot der Campus-Brett Anzeigenerstellungsseite mit Aufteilung der JSP-Seiten . . . . .	86

# 1. Einleitung

Das Internet und die Web 2.0-Bewegung haben in allen Bereichen des Lebens deutlich an Akzeptanz gewonnen. Die Welt rückt stetig näher zusammen. Informationen und Gedanken werden verstärkt und mit geringerer Hemmschwelle mit Hilfe des wachsenden Angebots an Kommunikationstechniken, seien es Plattformen im Internet, Instant Messaging Applikationen oder Smartphones mit Internetanbindung, ausgetauscht und geteilt. Für viele Menschen ist das tägliche Bloggen oder der Einsatz von Facebook und Co. zum gebräuchlichen Alltag geworden. So nutzen beispielsweise über 350 Mio. Menschen weltweit jeden Monat das soziale Netzwerk Facebook<sup>1</sup>. Diese als „Social Software“ bezeichneten Plattformen ermöglichen eine erleichterte Kontaktaufnahme / -pflege sowie einen direkteren (halböffentlichen) Informationsaustausch und versuchen durch die Vernetzung von Personen, oder auch Informationen, positive Netzwerkeffekte zu erzeugen.

Auch für Unternehmen ist die Entwicklung des schnelleren und direkteren Informationsaustauschs mittlerweile von existenzieller Bedeutung. So heisst es z.B. in [Buhse und Stamer \(2008\)](#) (S. 9), dass „Wissen, Kommunikation, die daraus entstehenden Netzwerke und eine darauf abgestimmte Unternehmensstruktur [...] mehr denn je [zählen]“. Die Übertragung der Konzepte aus Web 2.0 und „Social Computing“ für den Einsatz in Unternehmen formte den Begriff „Enterprise 2.0“, welcher u.a. im Jahre 2006 von Andrew P. McAfee geprägt worden ist<sup>2</sup>. Durch den Einsatz von „Social Computing“ im Unternehmen soll der Wissensaustausch der Mitarbeiter gefördert und somit die Stärke des gemeinschaftlichen Wissens genutzt werden<sup>3</sup>. Zudem versprechen sich Experten durch positive Netzwerkeffekte eine erhöhte Innovationsgeschwindigkeit<sup>4</sup>.

Obwohl vor allem Studenten sehr aktiv in Social Networking Plattformen, wie z.B. Facebook, StudiVZ, MySpace, Twitter etc., unterwegs sind, wird der Einsatz von Social Computing Konzepten auf den Campus der deutschen Hochschulen selten praktiziert. Die mangelnde Existenz dieser Konzepte verhindert in dieser stark kommunikativen Umgebung die in Social Networking Software beobachteten Effekte der Kollaboration, der Innovation und des

---

<sup>1</sup>Quelle: <http://facebookmarketing.de/features/was-bringt-facebook-2010>; Abrufdatum: 06.12.2010

<sup>2</sup>Siehe [McAfee \(2006b\)](#)

<sup>3</sup>Siehe [Surowiecki \(2004\)](#)

<sup>4</sup>Siehe [Buhse und Stamer \(2008\)](#) (S. 12)

Knowledge Sharing. Studenten sind in ihren Projekten und Semestergruppen verankert und haben nur einen eingeschränkten Überblick über die Geschehnisse / andere Projekte auf dem Campus. Ein Austausch außerhalb ihrer Semestergruppe wird nur bedingt geführt. Interessenspartner, z.B. Studenten mit ähnlichen Forschungszielen, bleiben oft unerkannt, obwohl sie sich fast jeden Tag am selben Ort, dem Campus, aufhalten.

## 1.1. Ziel dieser Arbeit

Ziel dieser Arbeit ist die Übertragung der Erkenntnisse aus der Enterprise 2.0- und der Social Computing-Forschung für einen Einsatz im Hochschulumfeld. Darüber hinaus sollen die neu gewonnenen Erkenntnisse in einer konkreten Anwendung implementiert werden. Hierfür sollen die Grundidee des Schwarzen Bretts aufgegriffen und soziale Interaktionen auf dem Campus durch eine optimierte Darstellung von digitalen Inhalten sowie einer Vernetzung von Personen und Informationen gefördert werden. Die Bereitstellung einer Social Networking-Plattform unterstützt die Kontaktaufnahme und den Informationsaustausch zwischen den Studenten. Dabei soll nicht nur der Austausch gefördert werden, sondern auch die „Awareness“ (das Gewahrsein) für Interessenspartner. Zusätzlich sollen personenbezogen relevante Inhalte mehr in das Sichtfeld der Studenten gelangen und so die Studenten beim Überblicken der Informationsmasse an deutschen Hochschulen unterstützen.

Über das traditionelle Einsatzgebiet des Schwarzen Bretts hinaus, nämlich den Austausch von Informationen in Form von Such- oder Verkaufswünschen, soll auch der Austausch von Wissen betrachtet werden. Durch einen effizienteren Wissensaustausch innerhalb der Hochschule und auch über deren Grenzen hinweg kann sich das Wissen vieler einzelner zu einem wertvolleren gemeinsamen Wissen ergänzen.

## 1.2. Aufbau der Arbeit

In Kapitel 2 werden zunächst Grundlagen aus Enterprise 2.0 und verwandten Themen erklärt, um zum Verständnis dieser Arbeit beizutragen.

In einer umfassenden Analyse wird die kommunikative Umgebung der Hochschule untersucht, das Forschungsgebiet Social Computing mit seinen verwandten Themen diskutiert und anschließend die Gedanken und Erkenntnisse auf die Hochschule übertragen. Den Abschluss der Analyse bildet die Betrachtung aktueller Forschungsarbeiten im Umfeld des Social Computing.

Das darauf folgende Kapitel 4 (Systemskizze) fasst die erarbeiteten Ideen zusammen und skizziert den Einsatz und den Funktionsumfang einer Social Computing-Plattform (Campus-Brett) auf dem Campus. Anhand dieser Systemskizze wird in Kapitel 5 die technische Umsetzung des Campus-Bretts entworfen und die Umsetzung von Teilen der Anwendung beschrieben.

In der abschließenden Zusammenfassung werden ein Ausblick auf mögliche nächste Schritte sowie weitere Einsatzmöglichkeiten der betrachteten Konzepte für das Umfeld der Hochschule gegeben.

## 2. Grundlagen

### 2.1. Computer Supported Cooperative Work (CSCW)

Der Begriff CSCW beschreibt den Einsatz von Computersystemen zur Unterstützung der Zusammenarbeit mehrerer Personen. Eine sehr allgemein gehaltene Definition dieses Begriffs findet sich in einer bereits im Jahre 1991 erschienenen Arbeit von Bowers und Benford<sup>1</sup>.

In its most general form, CSCW examines the possibilities and effects of technological support for humans involved in collaborative group communication and work processes

Die ersten Forschungsarbeiten zu CSCW gehen auf das Jahr 1984 zurück. In diesem Jahr fand unter der Leitung von Irene Greif und Paul Cashman ein Workshop statt, bei dem Teilnehmer aus unterschiedlichen Forschungsdisziplinen teilnahmen, um Ideen und Arbeitsergebnisse über die Unterstützung der Zusammenarbeit mithilfe von Computern auszutauschen<sup>2</sup>.

Eine wichtige Erkenntnis aus der CSCW-Forschung ist die Abhängigkeit der technischen Systeme mit den sozialen Systemen. Technische Lösungen, welche die Gruppenarbeit unterstützen sollen, müssen das Umfeld in welchem es eingesetzt werden soll verstehen und die in ihnen vorherrschenden sozialen Prozesse berücksichtigen. Die CSCW-Forschung beschäftigt sich deshalb mit dem Verstehen der sozialen Interaktion in Teams, Communities und Netzwerken sowie dem Entwurf, der Entwicklung und der Evaluierung von technischen Systemen, um die soziale Interaktion zu unterstützen<sup>3</sup>.

Technologien und Werkzeuge die aus der CSCW-Forschung entstehen werden als „Groupware“ bezeichnet. Groupware hat das Ziel eine Gruppe von Benutzern bei der Kommunikation, Zusammenarbeit und Koordination ihrer Aktivitäten zu unterstützen. Zu der Klasse Groupware zählen z.B. Nachrichtensysteme wie E-Mail, Konferenzsysteme (Audio- und Videokonferenzen) sowie Gruppeneeditoren zur gemeinsamen Arbeit an Dokumenten. CSCW-Systeme können in ihrer Anwendbarkeit klassifiziert werden in einer Kombination

---

<sup>1</sup>Siehe [Bowers und Benford \(1991\)](#)

<sup>2</sup>Siehe [Koch und Gross \(2006\)](#) und [Kurbel u. a. \(2009\)](#)

<sup>3</sup>Aus [Buhse und Stamer \(2008\)](#) (S. 44)

aus synchronen / asynchronen und zentralen / dezentralen Systemen<sup>4</sup>.

## 2.2. Enterprise 2.0

Der Begriff Enterprise 2.0 hat seinen Ursprung in der Web 2.0-Bewegung. Als Folge der großen Akzeptanz und dem Erfolg von Social Networking Plattformen wie Facebook, Xing oder Wikipedia, ist auch der Einsatz von Social Software (o.a. Social Computing) in Unternehmen verstärkt vorzufinden. Unter Social Software versteht man eine computergestützte Umgebung die es Personen ermöglicht sich zu treffen, Kontakte zu knüpfen, kollaborierend zu arbeiten und Communities aufzubauen. Positive Netzwerkeffekte des Social Networking, wie z.B. dem Auffinden von Interessenspartnern, dem gemeinsamen Arbeiten an einem Artikel oder dem gemeinsamen Lösen eines Problems, sollen der Produktivität eines Unternehmens helfen.

Andrew P. McAfee, einer der Pioniere des Enterprise 2.0, definiert diesen Begriff wie folgt<sup>5</sup>.

Enterprise 2.0 is the use of emergent social software platforms within companies, or between companies and their partners or customers.

Ist von Enterprise 2.0 die Rede, dann spricht man nicht nur von einem Stück Software, welches zur verbesserten Kommunikation in Unternehmen eingesetzt wird, sondern es beschreibt auch eine neue Unternehmenskultur in der die Mitarbeiter mehr Freiheiten genießen und ihre Meinungen und Ideen verstärkt in die tägliche Arbeit einbringen sollen. Mit dieser neuen Unternehmenskultur wird Unternehmen mehr Erfolg versprochen. In einigen Kreisen spricht man sogar von der Notwendigkeit dieses Wandels zur Erhaltung der Wettbewerbsfähigkeit<sup>6</sup>.

Zu den Zielen von Enterprise 2.0 zählen eine verbesserte Kommunikation, z.B. in Form von schnellerem direktem Wissensaustausch, und das Aufbauen von Netzwerken. Mitarbeiter sollen umfassend untereinander verbunden werden und Wissen miteinander teilen. Dazu ist es notwendig, dass Informationen nicht mehr in den gewohnten Bahnen von oben nach unten oder umgekehrt fließen und auf diesem Weg viele Filter durchlaufen, sondern dass sie sich weitgehend ungehemmt ausbreiten<sup>7</sup>.

---

<sup>4</sup>Vgl. [Nastansky u. a. \(2002\)](#)

<sup>5</sup>Siehe [McAfee \(2006a\)](#)

<sup>6</sup>Siehe u.a. [Buhse und Stamer \(2008\)](#) (S. 10)

<sup>7</sup>Beschreibungen sind z.T. übernommen worden aus [Buhse und Stamer \(2008\)](#) (S. 8ff)

## 2.3. Liferay Portal

Seit dem Jahre 2000 existiert unter dem Namen Liferay Portal eine Open-Source Portallösung, welche sich im Laufe der Jahre an dem Phänomen Enterprise 2.0 orientiert und weiterentwickelt hat. Die Einsatzbereiche dieser Portallösung reichen von einfachen personalisierten Service-Seiten über internes Prozess-Management bis hin zu Austauschplattformen mit unterstütztem kollaborativen Arbeiten.

Liferay Portal basiert auf einer Serviceorientierten Architektur (SOA), welche die Integration und die Entwicklung von Diensten, Funktionen und Komponenten vereinfacht und über Anwendungsgrenzen hinweg (wieder-)verwendbar macht. Eine maximale Skalierbarkeit und Wiederverwendbarkeit ist durch Decorator, Facade, Factory, Flyweight, MVC, Singleton und Value Object gegeben<sup>8</sup>.

Eine Liferay Portal Installation setzt sich aus Portlets zusammen, welche lose gekoppelt eingebunden werden. Der Funktionsumfang kann anhand eigens entwickelter Portlets, nach den Standards JSR-168 und JSR-286, erweitert werden.

Liferay Portal setzt sich grundlegend aus 3 Hauptkomponenten zusammen:

- **Liferay Portal**  
Liferay Portal stellt die Kernapplikation dar und ist die Basis für die anderen beiden Produktteile. Hier werden Kernfunktionalitäten wie z.B. die Benutzerauthentifizierung und das Taggen und Suchen von Web-Inhalten bereitgestellt.
- **Liferay CMS**  
Liferay CMS ist das in Liferay Portal integrierte Content-Management-System. Das Veröffentlichen und Verwalten von Inhalten im Liferay Portal wird über diese Komponente realisiert.
- **Liferay Collaboration**  
Die letzte Komponente stellt Unterstützungen für kollaboratives Arbeiten und dem Social Networking zur Verfügung. Zu den bereits integrierten Funktionalitäten zählen bspw. Blogs, Instant Messaging, Gruppenkalender und Umfragen.

---

<sup>8</sup>Siehe [comundus \(2010\)](#)

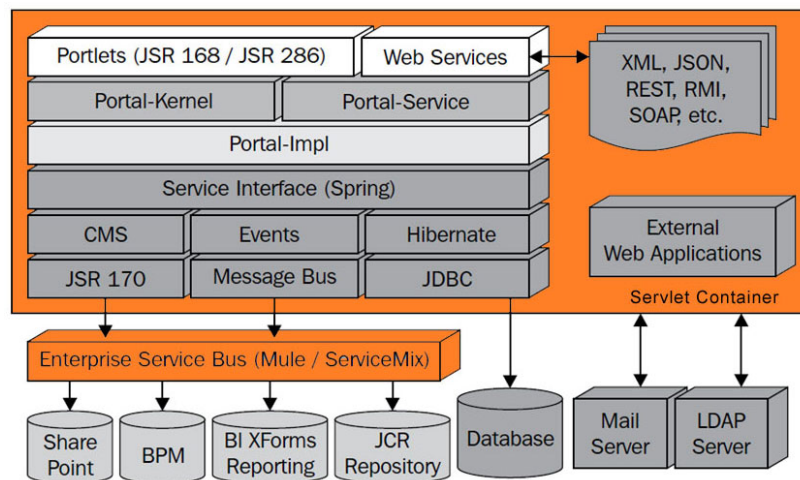


Abbildung 2.1.: Liferay Architekturskizze (Quelle: [comundus \(2010\)](#))

Der Einsatz neuester Java, J2EE und Web 2.0 Technologien ermöglicht die Umsetzung von umfangreichen Portallösungen mit komplexen Workflows. Dabei setzt Liferay Portal auf eine Reihe von Standards, wie AJAX und JSON, JSR-170 (Content Repository API for CMS), Hibernate, OpenSearch, WebDAV, WSRP, sowie Schlüsseltechnologien wie jQuery JavaScript Framework, Ruby on Rails, PHP, Spring und AOP, Velocity usw. Unterstützt werden dabei alle bekannten Applikationsserver (u.a. Apache Tomcat, GlassFish, JBoss, Jetty, WebSphere und WebLogic), Datenbankserver (u.a. IBM DB2, Apache Derby, HiperSonic, HSQLDB, MySQL, Oracle, PostgreSQL, Microsoft SQL Server und Sybase) sowie Betriebssysteme.

Das Portal ist für Benutzer über einen Webbrowser erreichbar, während externe Applikationen das Portal über Techniken wie z.B. REST, SOAP, RMI und JSON erreichen können<sup>9</sup>.

Liferay Portal ist Open-Source und damit frei erweiterbar. Üblicherweise werden neue Applikationen als Portlets entwickelt und eingebunden. Im Standardpaket der Liferay Portal Version 5.23 sind bereits knapp 80 Portlets integriert. Zu den integrierten Portlets gehören unter anderem ein Blog-Portlet, ein Wiki-Portlet, ein Message-Board-Portlet und ein Calendar-Portlet. Weitere frei verfügbare Portlets sind bei der Suche im Internet in großer Anzahl vorzufinden<sup>10</sup>. Die Anzahl der freien Entwickler ist nicht bekannt, lässt sich aber in etwa aus der Anzahl der Foren-Mitglieder<sup>11</sup> erschliessen, die momentan bei 13.198 liegt<sup>12</sup>.

<sup>9</sup>Beschreibungen der Technologien in [Yuan \(2009\)](#), [Liferay \(2010\)](#) und [comundus \(2010\)](#)

<sup>10</sup>Siehe beispielsweise auf <http://sourceforge.net/projects/lportal/files/> unter 'Plugins'; Abrufdatum: 22.07.2010

<sup>11</sup><http://www.liferay.com/community/forums>; Abrufdatum: 22.07.2010

<sup>12</sup>Stand: 22.07.2010



### 3. Analyse

Trotz der heutzutage zahlreich existierenden Kommunikationsmittel ergeht es Studenten häufig so, dass persönlich interessante oder auch wichtige Informationen im Hochschulumfeld, wie beispielsweise Klausuranmeldefristen oder Fristen für die Zahlung von Studiengebühren, nicht wahrgenommen werden. Kommunikationskanäle wie z.B. Webseiten, E-Mails oder Aushänge unterstützen den Hochschul-Alltag, sind dabei aber nicht durchgehend klar strukturiert, wodurch die Transparenz der relevanten Informationen nicht zwangsläufig gegeben ist. Baacke beschreibt in [Baacke \(1985\)](#) den Universitätscampus als eine „ganz und gar kommunikativ ausgelegte Umgebung [...]“, in der wohl Platz für den wissenschaftlichen Diskurs sowie für den Diskurs zwischen Studenten und Lehrenden sei, dessen „Distributions- und Kommunikationskanäle [...] bei näherem Hinsehen aber nicht aus[reichen], um [...] alles sagbar zu machen, was man zu sagen wünscht.“<sup>1</sup>. Dies ist ein Zustand, der heute nur noch bedingt zutrifft, denn durch das Medium Internet und seine starke Verbreitung in nahezu jedem Haushalt sind heutzutage Kommunikationskanäle in ausreichendem Maße vorhanden. Kommunikation über Internetforen oder E-Mail im Hochschulkontext ist für jeden Studenten zugänglich und wird auch mit sehr hoher Beteiligung von Studenten sowie Mitarbeitern und Professoren der Hochschule genutzt.

Der Zustand dieser kommunikativ ausgelegten Umgebung müsste heute etwas anders formuliert werden. Durch die Masse an Informationen die tagtäglich an Hochschulen entsteht erscheint es nahezu unmöglich, diese zu überblicken und die für einen persönlich interessanten bzw. relevanten Informationen heraus zu filtern. Internetforen werden von sogenannten Moderatoren „überwacht“, für den E-Mail-Verkehr gibt es verschiedene Verteiler, um die richtige Zielgruppe zu erreichen, und auch Webseiten haben eine gewisse Struktur, um auf der Suche nach bestimmten Informationen zu unterstützen. Bei all diesen Techniken kann jedoch nicht auf individuelle Interessen eingegangen werden, so dass bei der Informationsmasse, die an Hochschulen entsteht, meist nur Suchfunktionen helfen, um die Informationen herauszufiltern, nach denen man sucht.

In diesem Kapitel wird zu Beginn der status quo der Informationspräsenz an deutschen Hochschulen, vorwiegend am Beispiel der HAW Hamburg, analysiert. Anschließend wird versucht, die Erkenntnisse aus der Social Networking- und Enterprise 2.0-Forschung für

---

<sup>1</sup>Siehe auch [Fischer \(2009\)](#) (S. 25)

den Einsatz auf dem Hochschul-Campus zu übertragen, mit dem Ziel, eine effizientere Informationsdarstellung bzw. -kommunikation zu erreichen. Abschließend werden einige aktuelle Forschungsprojekte in diesem Umfeld betrachtet.

### 3.1. Informationspräsenz an Hochschulen

Im Umfeld einer Hochschule entsteht tagtäglich eine unüberschaubare Menge an Informationen. Informationen in Form von Wissen, wie z.B. Projekt- / Forschungsberichte, zur Organisation, z.B. Klausuranmeldefristen, aber auch Informationen privater Natur, welche meist in Form von Aushängen am Schwarzen Brett landen. Die folgende Analyse bezieht sich auf Untersuchungen, die primär an der Universität Hamburg und der HAW Hamburg durchgeführt wurden.

Zum Austausch und zur Verbreitung von Informationen an Hochschulen werden außerhalb der Vorlesungsveranstaltungen bevorzugt die Medien Internet (Webseiten), E-Mail und das Schwarze Brett genutzt. Der Postweg, über welchen hauptsächlich die Studienbescheide verschickt wurden, wurde an der HAW Hamburg im April 2009 verabschiedet. Ihm folgte die Einführung eines elektronischen Studentenausweises, des Hochschulportals Helios (elektronisches Informations- und Organisationssystem) und eines neuen Mail-Zugangs für alle Studenten.

Wissensinformationen wie z.B. Projektankündigungen und Projektergebnisse werden auf den Internetseiten der HAW Hamburg<sup>2</sup> veröffentlicht, über E-Mail-Verteiler verbreitet oder auf eigens organisierten Webseiten von Professoren publiziert. Da an einer Hochschule ständig mehrere Projekte laufen, steigt die Masse an Informationen dramatisch und sorgt so für Schwierigkeiten bei der Verwaltung sowie einer übersichtlichen und geordneten Darstellung der Informationen. Für den Informationsbetrachter ist es nahezu unmöglich, alle Informationen zu überblicken bzw. die für ihn relevanten Informationen heraus zu filtern.

Folgend werden die genutzten Kommunikationskanäle im Hochschulumfeld betrachtet und auf bestimmte Verhaltensmuster beim Einsatz analysiert.

#### 3.1.1. Hochschulwebseiten

Suchen Studenten nach Informationen rund um das Studium und den Campus, ist die Webseite der Hochschule die erste Anlaufstelle. Auf den Seiten der HAW Hamburg wer-

---

<sup>2</sup><http://www.haw-hamburg.de>; Abrufdatum: 22.10.2010

den Informationen angeboten wie z.B. Mensapläne, Veranstaltungspläne, Angebote für Abschlussarbeiten, Kontakte zu Professoren und viele mehr. Hochschulwebseiten haben den Ruf, unübersichtlich und wenig benutzerfreundlich zu sein, was u.a. in einer Studie der Psychologin Ulrike Duma an der Universität Mainz belegt wurde<sup>3</sup>. Auch die Webseiten der HAW Hamburg zeigen ähnliche Mängel auf. Viele Informationen sind schwer auffindbar und häufig nur über viele Klicks erreichbar. So ist der Speiseplan der Mensa am „Campus Berliner Tor“ erreichbar über „Technik und Informatik“ -> „Informatik“ -> „Studierende“ -> „Speiseplan“ begründet durch die Tatsache, dass sich die Mensa räumlich im Gebäude des Departments Informatik befindet. Der Link zum Speiseplan ist zudem leicht zu übersehen, da er sich in einer Vielzahl von anderen Links (insgesamt 18) im unteren Bereich versteckt hält. Wird ein Lageplan der Hochschule gesucht, navigiert man über „Studium“ -> „Team Studierendenzentrum“ -> „Lageplan“. Dies sind zwei Beispiele prominenter Themen, die auf Hochschulwebseiten gesucht werden und im Optimalfall mit minimalem Aufwand auffindbar sein sollten.

Eine eingebaute Suchfunktion auf den Webseiten der HAW Hamburg erleichtert die Suche nach Informationen. Diese basiert auf der Google Search Engine und bietet eine große Hilfestellung bei der gezielten Suche nach Informationen. Da es sich hierbei aber um eine Keyword-Suche handelt, werden eventuell für den Suchenden interessante Seiten nicht aufgezeigt, sofern die Person nicht die richtigen Begriffe eingibt. Beispielsweise wird die Seite mit dem Thema „Stellenausschreibungen“ bei der Suche nach dem Begriff „Jobs“ nicht aufgedeckt.

Die Benutzerfreundlichkeit bei der Erkundung der Webseiten wird zusätzlich negativ beeinflusst durch die unterschiedliche Gestaltung der Webseiten zwischen den Departments. So ist z.B. auf den Hauptseiten die Navigation auf der linken Seite, beim Klick auf das Department Medientechnik ist diese jedoch auf einmal oben zu finden. Eine einheitliche Struktur ist für den Besucher nicht erkennbar und erschwert somit das „Erforschen“ der Webseiten.

Ein weiteres Beispiel für Mängel in der Benutzerfreundlichkeit ist die Seite der Stellenausschreibungen des Departments Informatik<sup>4</sup>, welche thematisch vor allem bei Studenten sehr beliebt ist. Hier werden Angebote zu Abschlussarbeiten, Praktika, Studentenjobs und Festanstellungen publiziert. Ohne weitere Strukturierung werden diese Angebote lediglich als Links, vorwiegend zu PDF-Dateien, aufgelistet und mit einem Kurztitel des Angebots beschrieben. Die einzige Strukturierung, die hier angeboten wird, ist die Sortierung nach Publikationsdatum der Anzeige.

---

<sup>3</sup>Siehe [Duma und Hecht \(2006\)](#)

<sup>4</sup><http://www.informatik.haw-hamburg.de/691.html>; Abrufdatum: 23.10.2010

### 3.1.2. HAW-Mailer

An der HAW Hamburg steht den Studenten mit dem „HAW-Mailer“ ein zentraler E-Mail Service zur Verfügung. Dieser ist im Internet über einen herkömmlichen Webbrowser erreichbar. Neben den eigens geführten Konversationen zwischen Studenten und Professoren, erhält der Student über diesen Mail-Zugang Informationen rund um das Hochschulleben. Häufig genutzt wird dieser Kommunikationskanal von der Hochschulverwaltung und vom AStA (Allgemeiner Studierendenausschuss). Informationen die im Postfach der Studenten landen sind z.B. Studiengebührenbescheide, Rückmeldedefristen, AStA-Wahlen und Ankündigungen zu Vorträgen an der Hochschule.

Um die jeweiligen Zielgruppen zu erreichen existieren mehrere E-Mail-Verteiler, wie z.B. „alle Studierende des Departments Informatik“. Die sehr allgemein gehaltene Gruppierung durch die Verteiler hat zur Folge, dass häufig Nachrichten im Postfach von Studenten landen, welche sie persönlich nicht betrifft oder nicht interessiert. Um Informationen über diesen Kommunikationskanal zu erhalten muss der Student aktiv auf diesen Service zugreifen. Bei geringerer Aktivität in diesem Postfach, z.B. weil dieses nicht zu ihrer bevorzugten E-Mail-Adresse gehört, können aus diesem Grund wichtige Informationen, wie z.B. Studiengebührenbescheide, leicht übersehen werden.

### 3.1.3. Das Schwarze Brett

Ein Schwarzes Brett ist ein Gegenstand, häufig eine Tafel oder eine Pinnwand, welcher an halböffentlichen Orten, vor allem an Hochschulen, aufgestellt wird und als Kommunikations- und Austauschplattform dient. An einem Schwarzen Brett werden Informationen ohne weitere Genehmigung angebracht. Aufgestellt wird diese Einrichtung üblicherweise an hochfrequentierten Aufenthaltsorten wie z.B. in Eingangshallen oder in der Nähe einer Mensa. Die Größe eines Schwarzen Bretts beträgt oft  $10m^2$  oder mehr und bietet somit viel Platz für Anzeigen. Durch die sehr auffällige Größe und den hochfrequentierten Aufstellort wird das Schwarze Brett zum Blickfang.

Der Einsatz des Schwarzen Bretts hat sich seit geraumer Zeit<sup>5</sup> aufgrund seiner einfachen Anwendung und der auffälligen Präsenz in der Öffentlichkeit bewährt. Informationen werden auf Zetteln verfasst und am Schwarzen Brett angepinnt bzw. angeklebt.

An Hochschulen wird das Schwarze Brett hauptsächlich von Studenten genutzt, um

---

<sup>5</sup>Der Ursprung des Schwarzen Bretts ist nicht eindeutig geklärt. Laut Wikipedia geht der erste Einsatz eines Schwarzen Bretts in der Wissenschaft auf das Jahr 1687 zurück (siehe [http://de.wikipedia.org/wiki/Schwarzes\\_Brett](http://de.wikipedia.org/wiki/Schwarzes_Brett); Abrufdatum: 05.11.2010).

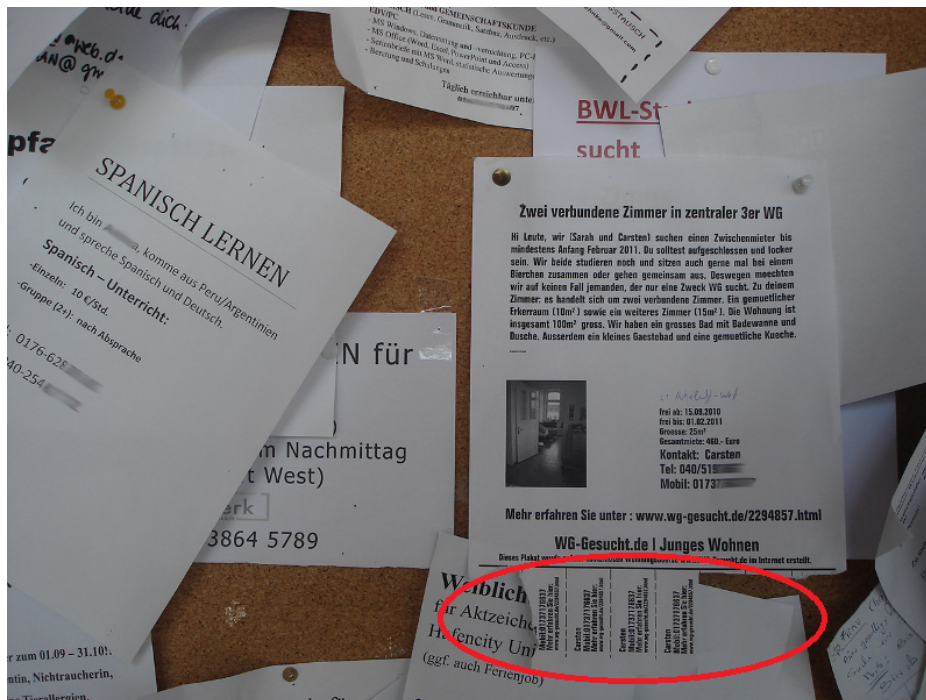


Abbildung 3.1.: Abreißzettel an Anzeigen erleichtern die Weitergabe von Kontaktdaten

Gesuche oder Angebote zu veröffentlichen. Andere Nutzer sind Professoren und Mitarbeiter der Hochschule sowie externe Personen, die häufig Jobangebote veröffentlichen. Am Schwarzen Brett angebrachte Informationen werden von vorbeigehenden Personen peripher wahrgenommen, gezielt von Interessenten gelesen sowie durch Wartezeiten oder sonstige Gründe bedingt „zufällig“ durchstöbert. Bei Entdeckung einer interessanten Anzeige notiert der Interessent wichtige Daten wie z.B. Kontaktdaten auf seinem Schreibblock oder Mobiltelefon oder hält sie direkt per Fotohandy fest. Um die Kontaktaufnahme zu erleichtern, werden Anzeigen häufig mit sogenannten Abreißzetteln versehen (siehe Abbildung 3.1), welche zur Mitnahme vorgesehen sind. Eine ebenfalls gängige Praxis ist die direkte Kontaktaufnahme per Mobiltelefon nach Sichtung der Anzeige.

Wichtig für die Akzeptanz und den Erfolg eines Schwarzen Bretts ist die Aktualität sowie die inhaltliche Relevanz der Anzeigen. Um dies zu gewährleisten ist es nötig, das Schwarze Brett regelmäßig zu überwachen und alte bzw. unpassende Beiträge zu entfernen. Im Hochschul Umfeld kümmert sich üblicherweise der Hausmeister um diese administrative Aufgabe.

Trotz der Administration ist ein häufiger Kritikpunkt die mangelnde Aktualität sowie die Unübersichtlichkeit am Schwarzen Brett. Fehlende Datumsangaben bei angebrachten Anzeigen und nicht selbstständiges Entfernen von abgelaufenen Anzeigen erschweren das Herstellen von Ordnung und Aktualität. Oft herrscht am Schwarzen Brett das totale Chaos.



Abbildung 3.2.: Das Schwarze Brett an der Universität Hamburg

Dies verdeutlicht Abbildung 3.2 am Beispiel eines Schwarzen Bretts an der Universität Hamburg im Foyer vor der Mensa. Anzeigen hängen hier kreuz und quer, überlappen sich oder sind sogar komplett verdeckt. Zudem liegen einige Zettel auf dem Boden. Abbildung 3.3(B) zeigt eine beispielhafte Anzeige, welche vier andere Anzeigen gleichzeitig verdeckt. Dieses Chaos führt zum Problem der unsicheren Informationsweitergabe. Der Anzeigenersteller hat nicht die Gewissheit, dass seine Anzeige das Publikum (die Studenten) erreicht. Dies kann durch die vorher genannten Ereignisse, dem mutwilligen oder unabsichtlichen Verdecken der Anzeige und dem nicht vorhersehbaren Verschwinden der Anzeige, z.B. durch Vandalismus oder Windzügen, die durch die Eingangshalle ziehen, geschehen. Der negative Effekt für das Publikum ist, dass verdeckte Anzeigen, die eventuell von Interesse wären, nicht gesehen werden.

Die chaotische Anordnung der Anzeigen ist ein typischer Charakter des Schwarzen Bretts, wodurch es letztlich so authentisch wirkt. Das Publikum bekommt das Gefühl, mit gleichgesinnten Menschen (indirekt) zu interagieren. Zusätzlich bekommt das Publikum, sofern es auf der Suche nach Gegenständen ist, das Gefühl einer „Schnäppchenjagd“ (vergleichbar mit einem Flohmarktbesuch). Das Schwarze Brett hat hierdurch einen sehr menschlichen Charakter, was gelegentlich zusätzlich z.B. mit persönlichen Fotos des Erstellers (siehe Abschnitt 3.1.3, Gestaltung der Anzeigen) in den Anzeigen unterstützt wird.

Schwarze Bretter im Hochschulumfeld sind traditionell, nicht zuletzt unterstützt durch

die Simplizität, beliebt und häufig genutzt. Vor allem die Überlagerung mehrerer Anzeigen ist ein Indiz für die rege Benutzung des Schwarzen Bretts. Trotz der immensen Größe herrscht an nahezu jedem Schwarzen Brett Platzmangel.

### Gestaltung der Anzeigen

Bei der Erstellung von Anzeigen für das Schwarze Brett sind der Kreativität keine Grenzen gesetzt. Betrachtet man die gestalterischen Merkmale der Anzeigen, so lassen sich drei Haupttypen erkennen, die sich aus den Zielen der Anzeige ableiten:

- **Auffallend (kreativ)**

Im Vordergrund steht hier, dass die Anzeige als Blickfang wirkt und so möglichst viele Leute erreicht. Hierbei wird meistens mit kreativen Mitteln wie z.B. Farben und Fotos gearbeitet. Auch die Handschrift zählt als ein kreatives Mittel.

- **Seriös (sachlich)**

Meist handelt es sich hierbei um Jobangebote, bei denen Seriösität im Vordergrund steht. Diese Art von Anzeigen sind übersichtlich gestaltet, enthalten oft viele Informationen und sind maschinell erstellt.

- **Zweckorientiert (schlicht)**

Ohne aufwendig kreativem Layout werden hier die wichtigsten Informationen, häufig in einer Auflistung, zusammengefasst. Dies ist die einfachste Form einer Anzeige und entsteht aus Gründen von Zeitmangel oder geringerer Motivation. Trotzdem erreicht auch dieser Typ einen Großteil des Publikums und erfüllt somit seinen Zweck.

Der Typ der Anzeige ist nicht immer eindeutig zu identifizieren. In vielen Fällen überschneiden sich die Typen. Abbildung 3.3 zeigt Ausschnitte von Schwarzen Brettern, auf denen Anzeigen mit typischen Gestaltungsmerkmalen zu sehen sind.

Wird eine Anzeige am Schwarzen Brett angebracht, verfolgt der Verfasser zunächst das Ziel, sein Publikum zu erreichen. Eine Taktik ist, die Anzeige möglichst zentral zu positionieren. Um aufzufallen, kann zusätzlich gestalterisch gearbeitet werden. Oft wird bei Anzeigen mit Farben und Fotos gearbeitet, welche die Aufmerksamkeit auf sich ziehen sollen. Ein kreatives Beispiel dafür zeigt Abbildung 3.3(A), auf dem ein aus Farbkarton selbst gebasteltes Gesuch zu sehen ist. Typisch für ein Gesuch / Angebot sind hier auch die Abreißzettel, rot eingekreist, um die Kontaktaufnahme zu erleichtern.

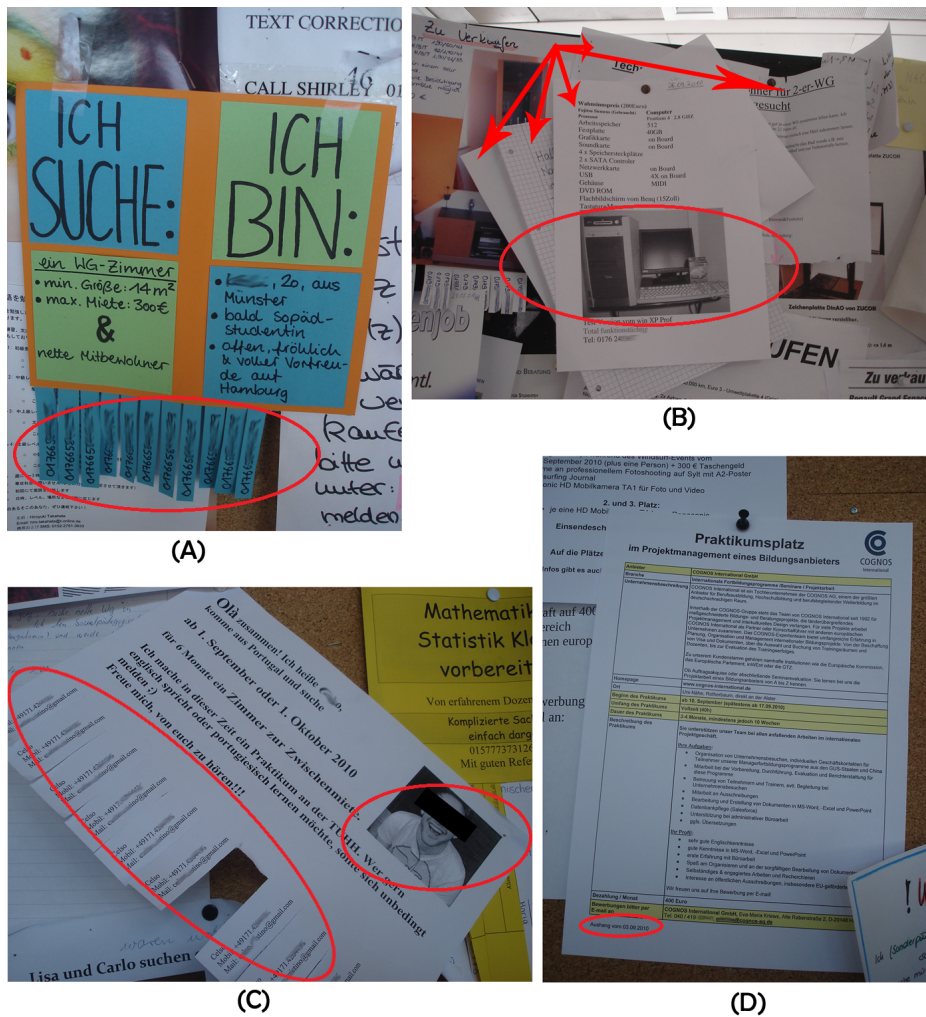


Abbildung 3.3.: Merkmale verschiedener Anzeigen am schwarzen Brett

Der Einsatz von Fotos in Anzeigen kann verschiedene Ziele haben. Zum einen dienen sie als Blickfang und unterstützen eventuell den Zweck der Anzeige. Auf Abbildung 3.3(B), die einen angebotenen Gegenstand zeigt, ist diese Zweckorientiertheit erkennbar. Um weiter aufzufallen, wurde in diesem Beispiel die Anzeige sehr zentral positioniert und ein Verdecken von fünf anderen Anzeigen in Kauf genommen.

Eine andere Wirkung erzielen z.B. persönliche Fotos wie in Abbildung 3.3(C). Bei dieser Anzeige handelt es sich um ein WG-Gesuch. Durch ein fröhliches Foto des Suchenden wird versucht, die Anzeige und somit auch sich selbst sympathisch wirken zu lassen. Auch hier werden wieder zum Zwecke der leichten Kontaktaufnahme Abreißzettel eingesetzt.



Abbildung 3.3(D) zeigt ein typisches Jobangebot, in diesem Fall ein Praktikumsangebot, am Schwarzen Brett. Das Layout ist übersichtlich und wenig aufwändig gestaltet, wodurch eine gewisse Seriösität, unterstützt durch das Firmenlogo, erzielt wird. Die Datumsangabe am unteren Rand der Anzeige unterstützt die Aktualität der Anzeige für Bewerber sowie für den Administrator des Schwarzen Bretts.

### **Digitales Schwarzes Brett**

Mit dem Erfolg des Internets findet auch das Schwarze Brett seinen Weg in die Digitalität. Viele Hochschulen verfügen mittlerweile über ein online verfügbares Schwarzes Brett, welches entweder von Mitarbeitern der Hochschule oder von Studenten administriert wird. Die Gestaltung dieser Internetdienste ist in den meisten Fällen recht simpel und häufig dem Aufbau eines Forums nachempfunden. Anzeigen werden ohne aufwändiges Layout, häufig rein textbasiert, eingetragen und vorgegebenen Kategorien zugeordnet. Die Aufteilung in Kategorien und Unterkategorien, welche bei fast allen untersuchten digitalen Schwarzen Brettern beobachtet wurde, verschafft Ordnung und soll dem Anwender helfen, schnell an die gewünschten Informationen zu kommen. Typisch ist eine Aufteilung in Angebote und Gesuche mit Unterkategorien wie z.B. Wohnung, Jobs, Bücher, Verschiedenes. In diesen Kategorien kann der Anwender navigieren, um die Ergebnisse nach seinen Wünschen zu filtern.

Um der Einfachheit des Schwarzen Bretts gerecht zu werden, wird in den meisten Fällen auf Login-Formalitäten verzichtet, wodurch theoretisch jeder ohne weitere (umständliche) Sicherheitsprüfungen in der Lage ist, Einträge zu veröffentlichen. Der Nachteil dieser Simplifizierung ist die Gefahr von Mißbrauch, die eine erhöhte administrative Tätigkeit notwendig macht. Für die Kontaktaufnahme hinterlassen die Anwender hauptsächlich ihre E-Mail Adresse. Auch Telefonnummern werden in diesem Zusammenhang häufig preisgegeben.

Ein weiterer Nachteil von digitalen Schwarzen Brettern ist, dass sie nicht direkt im Sichtfeld der Studenten sind. Personen müssen Kenntnis von diesem Internetdienst haben und aktiv auf diesen zugreifen. Der Effekt des zufälligen Entdeckens interessanter Anzeigen ist hier deutlich geringer als beim herkömmlichen Schwarzen Brett auf dem Campus.

### **Themen am Schwarzen Brett**

Das Schwarze Brett in der Hochschule eignet sich thematisch nicht für jede Form von Anzeigen. Eine Anzeige über einen Senioren-Fussballclub würde in diesem Umfeld sicherlich auf wenig Resonanz stoßen. Ein sehr beliebtes Thema am Schwarzen Brett der Hochschulen

ist die Wohnungs- / Zimmersuche. Eigene Beobachtungen am Schwarzen Brett ergaben in etwa folgende Aufteilung der Themen:

Thema	Anteil
Wohnungs-/ Zimmersuche	40%
Angebote/ Gesuche von Gegenständen	20%
Sprachkurse/ -partner	10%
Nachhilfe	10%
Jobangebote	10%
Eventankündigungen	5%
Sonstiges	5%

Tabelle 3.1.: Themen am Schwarzen Brett mit ihrer geschätzten Aufteilung

Auch in den online angebotenen Schwarzen Brettern findet eine ähnliche Aufteilung der Themenbereiche statt. Angebote und Gesuche von Produkten werden häufig noch detaillierter aufgeteilt, z.B. in Computer, Fahrräder, Bücher etc. Nach Vergleich mehrerer Onlineportale ergeben sich zusätzlich folgende Themen, für die das Schwarze Brett eingesetzt wird.

- Mitfahrgelegenheiten
- Kontakte
- Abschlussarbeiten

Für die Auswertung der Themenbereiche wurden u.a. folgende online angebotene Schwarze Bretter analysiert (Stand: Oktober 2010).

- Anzeigen-Onlinedienst der Universität Leipzig<sup>6</sup>
- Schwarzes Brett für Studenten in Aachen<sup>7</sup>
- Internet-Flohmarkt der Georg-August-Universität Göttingen<sup>8</sup>
- Schwarzes Brett der Universität Osnabrück<sup>9</sup>
- Das Schwarze Brett der Humboldt-Universität zu Berlin<sup>10</sup>
- Schwarzes Brett der Technischen Universität München<sup>11</sup>

<sup>6</sup><http://dsb.uni-leipzig.de/>; Abrufdatum: 16.10.2010

<sup>7</sup><http://www.campuslife.de/anzeigen>; Abrufdatum: 16.10.2010

<sup>8</sup><http://www.uni-goettingen.de/de/61905.html>; Abrufdatum: 16.10.2010

<sup>9</sup><http://www.uni-osnabrueck.de/SchwarzesBrett/standard/schwarzesbrett.php>; Abrufdatum: 16.10.2010

<sup>10</sup><http://www2.hu-berlin.de/schwarzesbrett/>; Abrufdatum: 16.10.2010

<sup>11</sup>[http://portal.mytum.de/schwarzesbrett/folder\\_listing](http://portal.mytum.de/schwarzesbrett/folder_listing); Abrufdatum: 16.10.2010

### 3.1.4. Interaktives Infoboard im Department Informatik der HAW Hamburg

Im Department Informatik der HAW-Hamburg wird den Studenten im 11.Stock ein interaktiver Informationsservice über einen Single-Touch Bildschirm angeboten. An diesem Infoboard können Studenten folgende statische und interaktive Informationen abrufen<sup>12</sup>:

- Speiseplan der Mensa
- Übersicht wichtiger Termine
- Details zu angebotenen Wahlpflichtfächern
- Veranstaltungspläne
- Übersicht der aktuellen Belegung der Rechner in den PC-Pools

In Abbildung 3.4 ist die Darstellung der angebotenen Dienste zu sehen. Der Benutzer navigiert am Infoboard per Fingerberührung über Buttons im oberen Bereich des Bildschirms. Die Darstellung der Inhalte ist, unterstützt durch die Größe des Bildschirms, sehr übersichtlich gestaltet und spiegelt eine kleine Auswahl, für Studenten der HAW Hamburg, interessanter Dienste wider. Besonders die Übersicht zur aktuellen Belegung der freien Rechner in den PC-Pools und der aktuelle Speiseplan der Mensa sind sehr beliebt.

Das Infoboard im 11.Stock ist im Aufenthaltsraum vor den Fahrstühlen aufgestellt. Durch ihre auffällige Platzierung in der Mitte des Raumes (siehe Abbildung 3.5) wirkt er als Blickfang. Studenten können gezielt nach Informationen suchen, z.B. bei Ankunft im 11. Stock auf der Suche nach einem freien Rechner, oder beim Warten auf den Fahrstuhl in den Inhalten stöbern. Zudem können Inhalte beim Vorbeigehen peripher wahrgenommen werden.

Die simpel gehaltene Struktur und Aufmachung des Infoboards, sowie die intuitive Bedienung per Fingertipp, machen Informationen auf dem Campus schnell und direkt für jedermann sichtbar.

---

<sup>12</sup>Stand: 23.11.2010

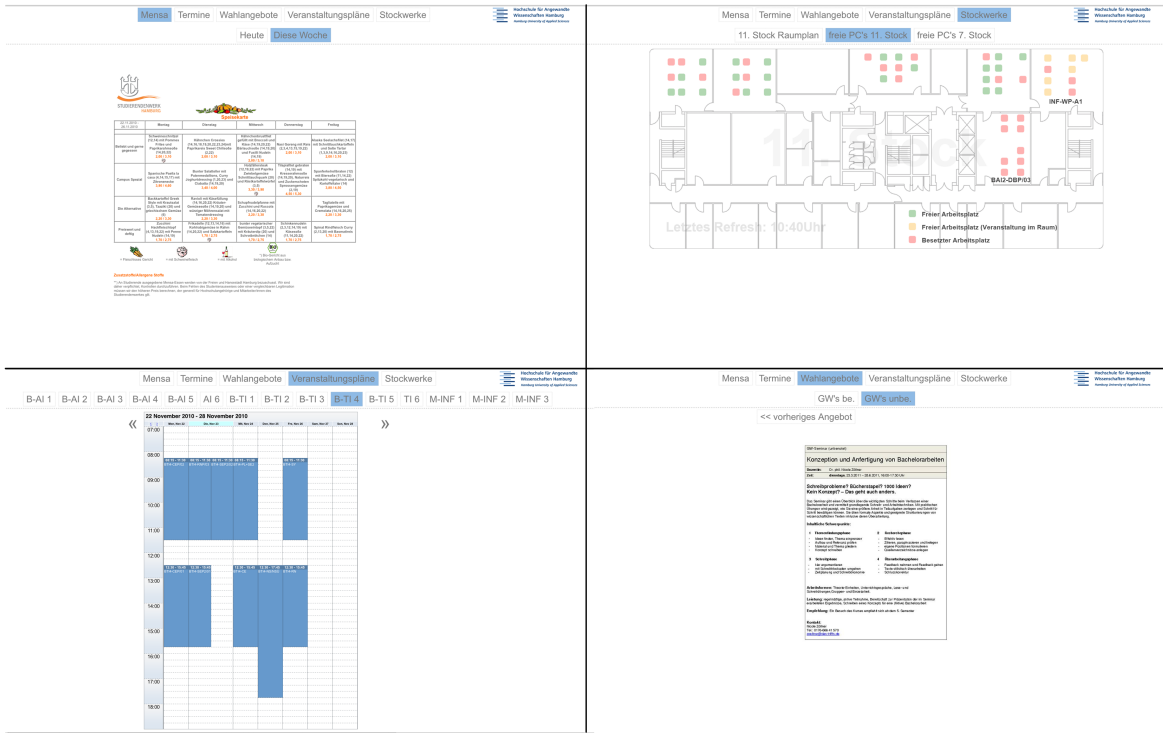


Abbildung 3.4.: Angebotene Inhalte am Infoboard

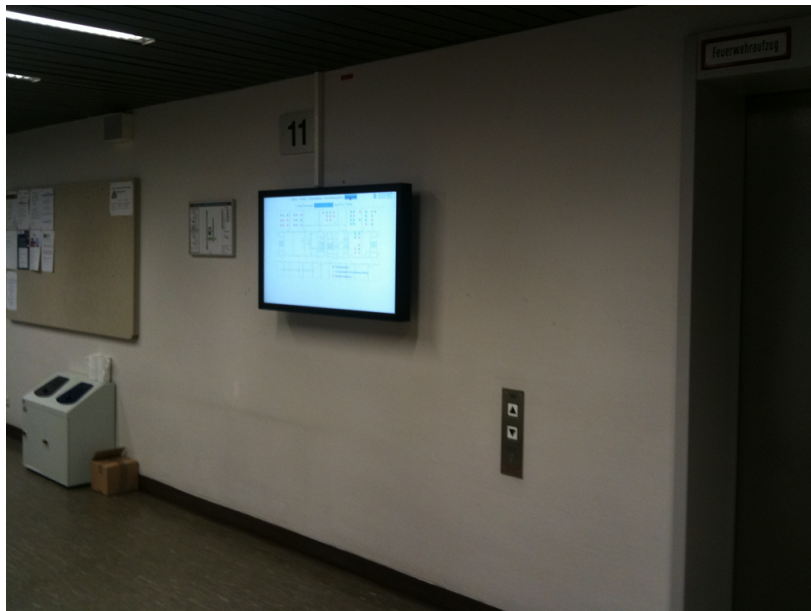


Abbildung 3.5.: Das Infoboard im AI-Labor der HAW Hamburg

## 3.2. University 2.0

Wie zu Beginn dieses Kapitels beschrieben, ist die Hochschule eine äußerst kommunikativ ausgelegte Umgebung. Kommunikation auf dem Campus existiert in Form von Austausch privater Interessen, z.B. über ein Schwarzes Brett oder einem Internetforum, sowie in Form von Wissensaustausch, welcher vorwiegend in Form von wissenschaftlichen Arbeiten und Projektberichten in Bibliotheken, Hochschulwebseiten etc. zu finden ist.

An Hochschulen laufen permanent Projekte und es wird gleichzeitig in etlichen Themenbereichen geforscht. Häufig existieren viele Forschungsteams nebeneinander und arbeiten nicht optimal zusammen. Verwandte Forschungsarbeiten, auch hochschulübergreifend, werden nicht wahrgenommen, wodurch möglicherweise vielversprechende Kollaborationen verloren gehen. Wissen und Erkenntnisse aus anderen Teams oder Projekten bleiben verborgen und können sich daher nicht optimal ergänzen. Als Beispiel dient hier das Veröffentlichen von wissenschaftlichen Arbeiten an der HAW Hamburg. Üblicherweise werden diese von dem jeweils betreuenden Professor auf einer selbst verwalteten Webseite veröffentlicht. Dabei sind Querverweise zu themenverwandten Arbeiten nicht direkt ersichtlich.

In [Unsworth \(2008\)](#) (S. 227) wird die Hochschule wie folgt beschrieben:

Universities are, at their core, organizations that cultivate knowledge, seeking both to create new knowledge and to preserve and convey existing knowledge, but they are remarkably inefficient and therefore ineffective in the way that they leverage their own information resources to advance that core activity.

John Unsworth beschreibt hier die Universität als eine Einrichtung, die primär mit dem Gegenstand „Wissen“ hantiert. Universitäten verwalten Wissen und generieren neues. In diesem Zusammenhang kritisiert der Autor die unzureichende Berücksichtigung des Web 2.0-Gedanken im Hochschulumfeld. Universitäten verfallen zu monolithischen Informationssystemen und wirken so innovationshemmend. Ein verbesserter Informationsaustausch innerhalb aber auch zwischen den verschiedenen Universitäten wird hier notwendig und soll durch, wie er sie nennt, „Seamy“ Systeme erreicht werden. Durch „unsichtbares“ Verknüpfen und Aufarbeiten von Informationen, oder auch Personendaten, entstehen Synergieeffekte und können zusätzlich neue, vorher unbemerkte Sichtweisen auf ein Thema hervorrufen. Als Beispiele werden Services genannt wie Tag Clouds, welche aus den Titeln der Hochschul-Publikationen generiert werden, wodurch aktuell beliebte Themen sichtbar werden, oder Suchfunktionen, die nach Auflistung der Informationsergebnisse eine Übersicht von Personen darstellen, die mit dem gesuchten Themenbereich in Verbindung gebracht werden<sup>13</sup>.

---

<sup>13</sup>Eine Demoanwendung (BRAIN) aus einem Projekt von John Unsworth, welche automatisch Querverweise aus einem Dokument generiert, ist zu finden unter <http://brain.lis.uiuc.edu>; Abrufdatum: 09.09.2010

Ein weiterer wichtiger Aspekt für die Hochschule im „Social Computing Zeitalter“ ist die Bildung und Weiterentwicklung von (virtuellen) Communities und den daraus resultierenden positiven Netzwerkeffekten. Das mangelnde Bewusstsein (Awareness) von Interessenspartnern im eigenen Umfeld wird auch in [Borcea u. a. \(2007\)](#) (S. 1) kritisiert:

Currently, people living or working in the same places routinely miss opportunities to leverage interpersonal affinities (e.g., shared interests and backgrounds) for friendship, learning, or business through a simple lack of awareness.

Durch den Aufbau bzw. die Stärkung von sozialen Beziehungen / Netzwerken, entstehen automatisch neue Wege der Kollaboration, Innovation und des Knowledge Sharing<sup>14</sup>. Ein gutes Beispiel dieses Effekts ist die Social Networking Plattform Facebook<sup>15</sup>. Dort können Geschehnisse in einem Netzwerk von Freunden beobachtet und z.B. eigene Fragen gepostet werden, welche wiederum von Personen aus dem Netzwerk gesehen und beantwortet werden können. Je größer und breiter gestreut dieses Netzwerk aufgebaut ist, desto höher ist die Chance, Personen zu finden, welche die gestellte Frage zielgerecht beantworten können.

Der Soziologe Mark Granovetter hebt in diesem Zusammenhang besonders die Bedeutung von schwachen Beziehungen hervor<sup>16</sup>. Starke Beziehungen zwischen Menschen entstehen aus langfristigen, häufigen und länger andauernden Interaktionen, schwache Beziehungen aus nicht häufigen, eher beiläufigen Interaktionen<sup>17</sup>. Granovetter ist der Meinung, dass Personen, die sich vorwiegend in derselben Personengruppe bewegen, zu selten neue Impulse bekommen. Schwache Beziehungen, sog. lose Kontakte, bieten ein noch größeres Potenzial für Innovationen und Knowledge Sharing, als es starke Beziehungen schon haben. Der Vorteil von schwachen Beziehungen besteht darin, dass Wissen und Ideen in diesem Umfeld mit größerer Wahrscheinlichkeit aufgrund unterschiedlicher schulischer / beruflicher Laufbahnen breiter gefächert sind als es im engen Freundeskreis, in dem Wissen häufig viele Redundanzen aufweisen, der Fall ist. Dieses Szenario lässt sich direkt auf die Hochschule übertragen. Hat beispielsweise der Informatik-Student A eine Frage im Themengebiet BWL, hat in seinem sozialen Netzwerk aber nur einen BWL-Studenten B, so erreicht Student A durch das soziale Netzwerk von Student B, in welchem sich eine Vielzahl von BWL-Studenten befinden, mehr Themenexperten und erhöht so seine Chance, eine Lösung oder neue Ideen für sein Problem zu erhalten.

Wie eng die Beziehungen in sozialen Netzwerken sind, belegte Stanley Milgram 1967 mit dem „small world experiment“<sup>18</sup>. Bei diesem Experiment sollten ausgewählte US-Bürger

<sup>14</sup>Vgl. [Buhse und Stamer \(2008\)](#) (S. 23ff)

<sup>15</sup><http://www.facebook.com>; Abrufdatum: 10.09.2010

<sup>16</sup>Siehe [Granovetter \(1973\)](#)

<sup>17</sup>Vgl. [Buhse und Stamer \(2008\)](#) (S. 25)

<sup>18</sup>Siehe [Milgram \(1967\)](#)

versuchen, eine willkürlich ausgewählte Zielperson in den USA zu erreichen, indem sie jeweils Nachrichten in ihrem Bekanntenkreis weiterleiten. Aus diesem Experiment entwickelte Milgram das Konzept der „six degrees of separation“, welches besagt, dass jede Person im Durchschnitt über etwa 6 andere Personen von jeder beliebigen Person auf der Welt entfernt ist. Mit einem effizienten Einsatz sozialer Netzwerke könnten demnach Personen aus nahezu jedem Wissensbereich auf schnellem Wege erreicht und zur Lösungs- / Ideenfindung herangezogen werden. Abbildung 3.6 visualisiert beispielhaft die Verbindungen in einem sozialen Netzwerk.

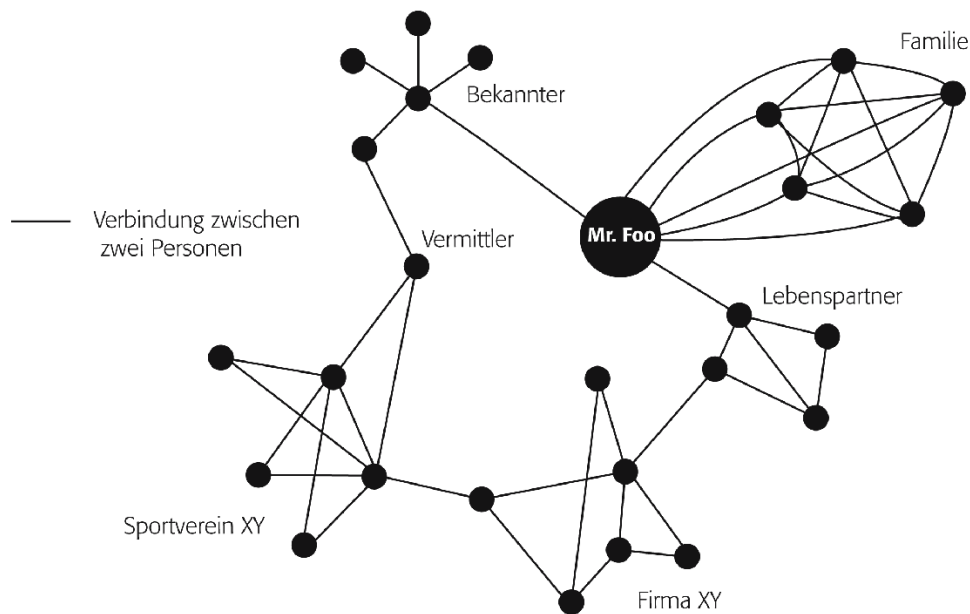


Abbildung 3.6.: Visualisierung der Verbindungen in einem sozialen Netzwerk (Quelle: [Kreßner \(2010\)](#))

Zur Realisierung positiver Netzwerkeffekte in Social Computing-Plattformen sind „kritische Masse“-Effekte von zentraler Bedeutung. Die kritische Masse beschreibt die Anzahl von Mitgliedern, die nötig ist, um ein Netzwerk für deren Anwender attraktiv werden zu lassen, denn „erst durch die steigende Gesamtuserzahl entsteht ein steigender direkter derivativer (abgeleiteter) Nutzen in Form von größerem Angebot, der für die Attraktivität des Netzwerks entscheidend ist“ ([Kopp \(2006\)](#) (S. 14)). Häufig liegt die kritische Masse bei etwa 10% der potenziellen Nutzer<sup>19</sup>. Ist die kritische Masse erreicht, kommt es in der Regel zum sog.

<sup>19</sup>Vgl. [Clement und Albers \(2005\)](#) und [Rogers \(1995\)](#)

„Bandwaggon-Effekt“<sup>20</sup>, der eigenständigen Entwicklung des Netzwerks.

Ist der Nutzen eines neuen Services nicht direkt ersichtlich, kommt es in der frühen Phase oft zu einer Situation des gegenseitigen Beobachtens und der abwartenden Haltung, auch „Pinguin-Effekt“ genannt. Auftretende Aufwandskosten bei der Nutzung des neuen Services können diesen Effekt unterstützen. Eine der wichtigsten Lehren aus CSCW ist, „dass immer ein Ausgleich zwischen Aufwand und Nutzen angestrebt werden sollte, um eine hohe Akzeptanz sicherzustellen. [...] Mit einer hohen Motivation besteht die Möglichkeit, einen eventuellen Zusatzaufwand zu kompensieren.“ (Buhse und Stamer (2008) (S. 54ff))

Die Motivation zur Beteiligung in einem Netzwerk lässt sich auch ohne Erreichen des direkten derivativen Nutzens steigern. Ein Beispiel hierfür ist das eingeführte „Value-System“ in der Community-Plattform Sunspace<sup>21</sup>. Nutzer erhalten durch ihre Aktivitäten Belohnungen in Form von „virtuellen Abzeichen“, „virtuellen Punkten“ etc. und bekommen somit einen „Wert“ in der Community. Solche Belohnungssysteme werden vor allem in Computerspielen mit großem Erfolg immer stärker verwendet. Aktuell prominente Beispiele sind „World of Warcraft“ und „Starcraft 2“, bei denen ergänzend zu den normalen Spielinhalten sog. Erfolge („Achievements“)<sup>22</sup> erreicht werden können, die die Motivation zusätzlich steigern.

---

<sup>20</sup>Siehe [http://en.wikipedia.org/wiki/Bandwagon\\_effect](http://en.wikipedia.org/wiki/Bandwagon_effect); Abrufdatum: 27.10.2010

<sup>21</sup>Siehe Reiser (2009a)

<sup>22</sup>siehe [http://en.wikipedia.org/wiki/Achievement\\_\(video\\_gaming\)](http://en.wikipedia.org/wiki/Achievement_(video_gaming)); Abrufdatum: 27.10.2010



### 3.3. Aktuelle Forschungsarbeiten / -ergebnisse

Die Themen Social Computing und Enterprise 2.0 erlebten in den vergangenen Jahren einen regelrechten Boom. Aus diesem Grund existiert auch eine Vielzahl von Projekten und Anwendungen in diesem Bereich. Im Folgenden werden einige aktuelle und anregende Projekte näher betrachtet, welche sich für den Einsatz an Hochschulen adaptieren lassen. Abschließend wird noch auf weitere Projekte verwiesen, die sich im näheren Umfeld dieser Arbeit bewegen.

#### 3.3.1. CommunityMirrors

Zu einer verbesserten Informationsdarstellung sowie eines effektiveren Informationsaustauschs entstand an der Universität der Bundeswehr München<sup>23</sup> ein Projekt mit dem Namen CommunityMirrors<sup>24</sup>. Als CommunityMirrors werden interaktive, auf Touch-Interaktion basierende Wandbildschirme, welche an halböffentlichen Orten eingesetzt werden sollen, bezeichnet.

Der Einsatz von CommunityMirrors ist begründet durch den derzeit wenig optimalen Zustand der Informationsdarstellung. Primär durch das Phänomen Web 2.0 sind heutzutage wie beschrieben Informationen in scheinbar unbegrenztem Volumen im Internet vorhanden. Inhalte sind auf Servern weltweit abgelegt und anhand von „Keywords“ über Suchmaschinen wie z.B. Google, zumindest theoretisch, auffindbar. Durch einen stetigen Informationszuwachs ist in vielen Bereichen jedoch ein „Information Overload“ vorhanden, der dazu führt, dass relevante Inhalte nur erschwert über die klassischen Suchmechanismen auffindbar sind. Zusätzlich kann die richtige Wahl an „Keywords“ ein Hindernis darstellen, da beispielsweise verschiedene Personen ein und dasselbe Thema mit unterschiedlichen Titeln versehen.

Der Einsatz von Social Software, wie z.B. Wikis, die den Informationsaustausch fördern, ist auch in Unternehmen immer häufiger vorzufinden. Der Informationsaustausch soll durch CommunityMirrors weiter optimiert werden. Inhalte werden durch das Herstellen von Beziehungen zwischen der Information und den Informationslieferanten innerhalb ihres Kontexts potenziellen Informationskonsumenten sichtbar gemacht. Das automatisierte Filtern und Strukturieren der Inhalte ermöglicht das zufällige und z.T. periphere Finden von Informationen statt des gezielten Suchens.

CommunityMirrors sollen durch ihre Integration in den sozialen Kontext die Kommunikation der Mitarbeiter fördern. Durch das Aufstellen der Wandbildschirme an (halb-)öffentlichen

<sup>23</sup><http://www.unibw.de/inf/fakultaet/>; Abrufdatum: 22.08.2010

<sup>24</sup><http://www.communitymirrors.net/>; Abrufdatum: 22.08.2010

Orten sind Informationen direkt und für mehrere Personen gleichzeitig sichtbar. Auf CommunityMirrors kann ohne definiertes Ziel in der dargestellten Information gestöbert werden. Mehrere Personen können gleichzeitig diese Informationen betrachten, wodurch der aktive und gemeinschaftliche (Informations-)Austausch am Bildschirm gefördert wird.

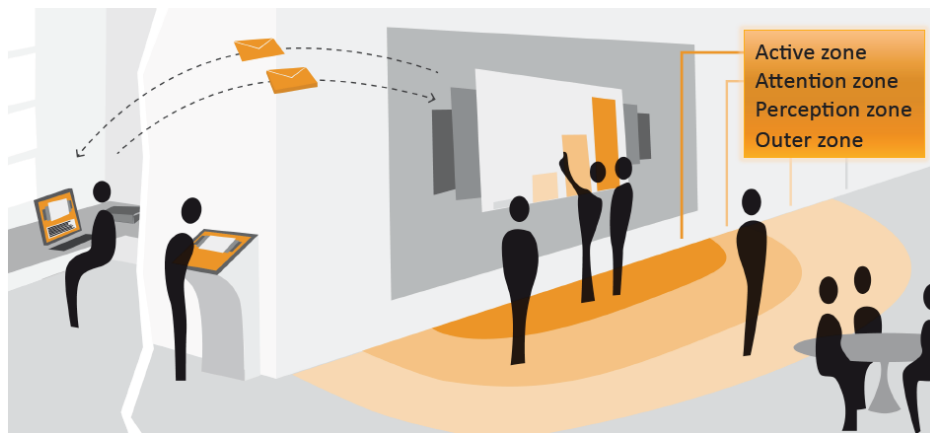


Abbildung 3.7.: Vision der CommunityMirrors. Personen interagieren gemeinsam am Wandbildschirm und arbeiten kollaborierend mit den Informationen (Quelle: Koch (2010))

Eine prototypische CommunityMirror-Anwendung wurde in Ott u. a. (2010b) als SocialNetworkingMirror vorgestellt. Diese Anwendung visualisiert anhand von Profilbildern soziale Netzwerke und zeigt mit optisch ansprechenden Mitteln Beziehungen zwischen Personen und Personengruppen mit gleichen Interessen auf. Profilbilder bewegen sich in dieser graphischen Umgebung auch ohne Interaktion frei über den Bildschirm und erzeugen so einen Serendipity-Effekt, also das zufällige Finden von Personen mit ähnlichen Interessen. Bei Berührung eines Profilfotos wird eine digitale Visitenkarte angezeigt. Zusätzlich werden Verbindungen dieser Person zu anderen Personen sichtbar. Aufgrund der Besonderheiten des Touchscreen-Interfaces geschieht eine Suche nach Kontakten statt mit manueller Texteingabe über eine auswahlbasierte Logik. Werden Personen mit dem Interesse an „Enterprise 2.0“ gesucht, navigiert man beispielsweise über „Suche“ > „Interessen“ > „E“ > „Enterprise 2.0“.

Um Informationen einfach zugänglich und „erlebbar“ zu machen, stehen bei CommunityMirror-Anwendungen eine intuitive Bedienbarkeit und die Erzielung von Serendipity Effekten im Vordergrund.

### 3.3.2. Google Wave

Google Wave<sup>25</sup> ist eine neue Kommunikations- und Kollaborationsplattform von Google und kann als Web 2.0- bzw. Enterprise 2.0-Lösung interpretiert werden<sup>26</sup>. Google Wave vereint verschiedene Konzepte der Kommunikation, wie z.B. E-Mail und Instant Messaging, in einer einzigen Plattform und setzt das Hauptaugenmerk auf die kollaborative Arbeit mit Dokumenten und anderen Inhalten. Um dies zu unterstützen, wird besonders darauf geachtet, die vorhandenen Konzepte zu optimieren und interaktiv sowie benutzerfreundlich zu gestalten.

Auf der Startseite des Google Wave Clients finden sich zunächst eine Übersicht der Kontakte sowie eine Übersicht der Themen, sogenannte Waves, in die man involviert ist. Die Übersicht der Themen hat einen Look ähnlich wie bei Google Mail<sup>27</sup>, wobei die Themen zusätzlich mit den Fotos der involvierten Personen dargestellt werden und über „number indicators“ (kleine Zahlen in einem grünen Kreis) visuell verdeutlicht wird, ob ein Thema neuen Inhalt bzw. Veränderungen enthält. Eine Wave besteht nicht ausschließlich aus Textinhalten, sondern eignet sich ebenso für den Austausch und die Kollaboration mit jeglicher Form von Inhalten, z.B. Google Maps, Spiele, Bilder oder auch Eventeinladungen.

Eine Wave hat die Form einer Diskussion oder eines gemeinsam genutzten Dokuments. Teilnehmer können sich Nachrichten schreiben oder auch in Echtzeit kommunizieren, ähnlich wie beim Chatting, wobei hier zusätzlich jeder getippte Buchstabe sofort sichtbar wird. In einer Wave können auch private Nachrichten verfasst werden, welche nicht für alle Teilnehmer sichtbar sind. Veränderungen in einer Wave können, nach dem Prinzip der Versionskontrolle, beliebig zurückverfolgt werden.

Google Wave wurde im Mai 2009 bei der Entwicklerkonferenz Google I/O vorgestellt. Wenig später wurde die Google Wave API als Open Source zur Verfügung gestellt, in der Hoffnung, dass freie Entwickler eine Reihe von Gadgets, Erweiterungen etc. rund um das Wave-Konzept implementieren und somit Google Wave noch attraktiver gestalten. Im August 2010 wurde die Entwicklung von Google Wave als eigenständiges Produkt aufgrund mangelnder Akzeptanz, so die offizielle Begründung, eingestellt.

---

<sup>25</sup><http://wave.google.com>; Abrufdatum: 29.08.2010

<sup>26</sup>Vgl. Schwarzer (2010)

<sup>27</sup><http://mail.google.com>; Abrufdatum: 22.08.2010

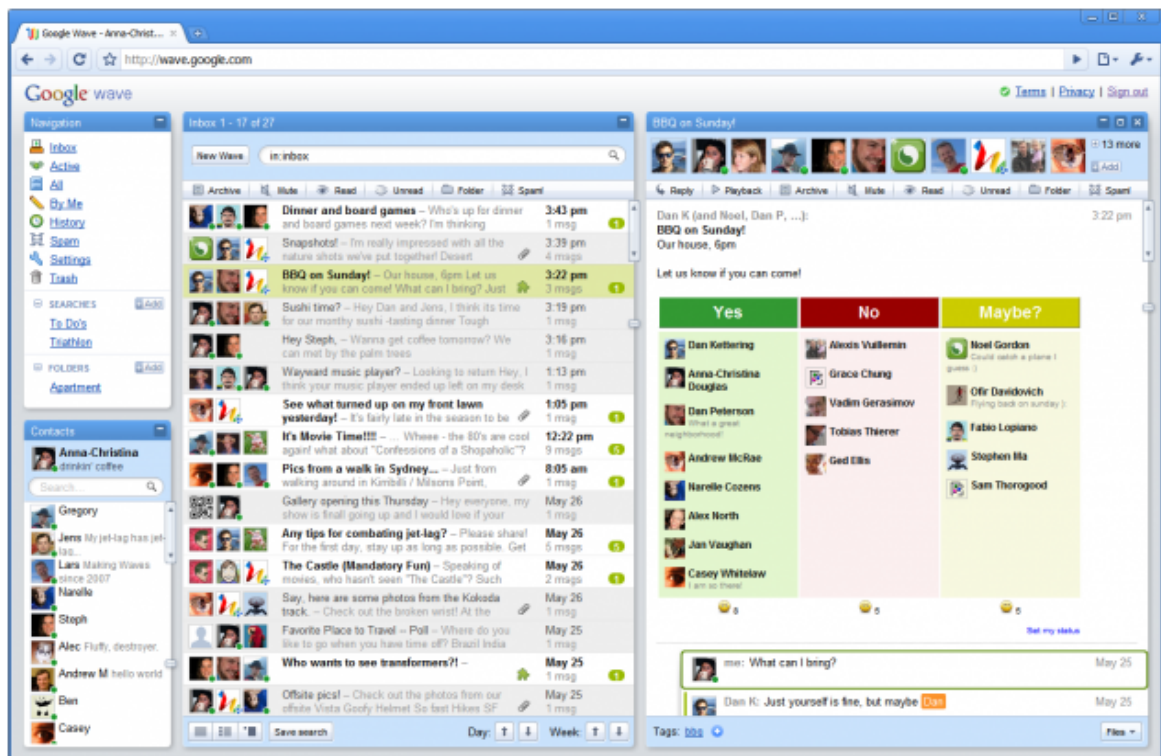


Abbildung 3.8.: Übersichtsbildschirm des Google Wave Client. Geöffnet ist ein Thema, das ein Voting beinhaltet. (Quelle: Siegler (2009–05–28))

### 3.3.3. Sunspace

Die Enterprise 2.0-Umgebung „Sunspace“ der Firma Sun Microsystems ist eine web-basierte Anwendung und baut auf die Wiki-Funktionalitäten, dem Erstellen und Editieren von zentral gespeicherten Inhalten, von Atlassian's „Confluence“<sup>28</sup> auf.

In Sunspace können mehrere unabhängige Communities gebildet werden. Von Nutzern erstellte Inhalte sind über Suchfunktionen Community-übergreifend auffindbar. Inhalte werden von Nutzern mit Tags versehen, was die Suche nach relevanten Informationen optimiert.

Sunspace beinhaltet viele Facebook-ähnliche Funktionen. Beispielsweise verfügt jeder Nutzer über ein Benutzerprofil, in welchem Kontaktinformationen, Aktivitäten, Rollen etc. dargestellt werden. Ebenso können Nutzer eine Kontaktliste pflegen, wodurch sie in Sunspace mit Personen verschiedener Communities verknüpft sind.

<sup>28</sup><http://www.atlassian.com/software/confluence/>; Abrufdatum: 12.08.2010

Zu den Community-Funktionen von Sunspace zählen News zur Anzeige aktueller Themen, das Austauschen / Teilen von Dateien und URLs sowie Wikis und Foren.

Eines seiner Hauptaugenmerke setzt „Sunspace“ auf ein Community Equity Model. Dieses Belohnungs- / Bewertungssystem dient der Förderung der Nutzeraktivität, aber auch zur besseren Strukturierung der Daten. Inhalte werden von Nutzern der Community bewertet und bekommen so einen „Wert“. Nicht nur Inhalte erhalten einen Wert, sondern auch Nutzer werden statistisch erfasst. So gibt es ein Ranking-System für Beitragserstellung (contribution) und die Teilnahme in der Community (participation). Es zählt dabei nicht nur die Menge der erstellten Beiträge, sondern auch die durch das Bewertungssystem ermittelte Qualität. Eine Belohnung der Nutzer erfolgt dabei durch eine Übersicht der Top 10-Nutzer auf der Startseite.

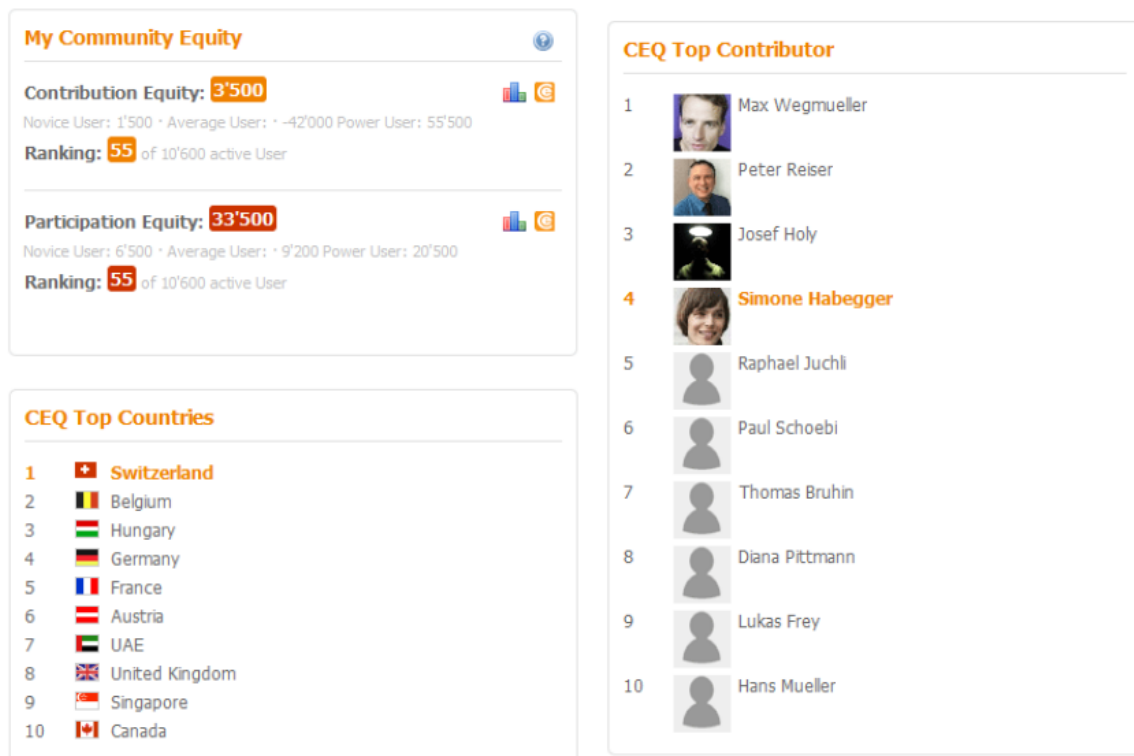


Abbildung 3.9.: Community Equity Model von Sunspace. Aktivitäten werden belohnt mit Punkten, wodurch sich ein Ranking ergibt. Die Top 10-User werden hervorgehoben. (Quelle: Reiser (2009b))

Zusätzlich zu den Bewertungen der Inhalte werden Aufruf-Statistiken erfasst, wodurch häufig gelesene oder editierte / kommentierte Inhalte einen „höheren“ Wert erhalten. Inhalte werden gemessen an ihren Werten sortiert.

Bei der Suche nach Inhalten werden zusätzlich Personen aufgelistet, die mit den Inhalten der Suchergebnisse im Zusammenhang stehen, sei es als Autor oder anderweitiger Beteiligung. Dies ermöglicht das Auffinden von sogenannten Themen-Experten.

Weitere Funktionalitäten bietet Sunspace über Plugins an. So kann jeder Nutzer sein Sunspace beliebig individuell gestalten bzw. erweitern, indem er Plugins aktiviert oder deaktiviert. Zum derzeitigen Funktionsumfang zählen u.a. Mywiki (persönliche Wikipedia-Umgebung, um Informationen / Wissen festzuhalten), MyBlog (Weblog-Umgebung zum Abgeben von Statusmeldungen oder Verfassen kurzer Artikel), Plazes (Location-based Service der auf einer Weltkarte darstellt wo Personen sich zur Zeit aufhalten) und BuzzBox (Anzeige von Statusnachrichten der Personen aus den zugehörigen Communities).

### 3.3.4. The MobiSoC Middleware for Mobile Social Computing

MobiSoC ist eine Middleware für Mobile Social Computing Applications (MSCAs), entwickelt von Studenten des New Jersey Institute of Technology<sup>29</sup>. MSCAs werden motiviert aus dem Wandel von den physischen Communities zu verstärkt virtuellen Communities, wie sie in Facebook, MySpace etc. vorzufinden sind, der jedoch fehlenden Awareness von Personen mit selben / ähnlichen Interessen an Orten, an denen man sich ständig aufhält (zu Hause, bei der Arbeit, auf dem Campus etc.). Zusätzlich sind sich Personen oft nicht bewusst über nahegelegene Orte oder soziale Events, welche sie normalerweise besuchen würden.

Untersuchungen haben ergeben, dass Personen mehr und mehr bereit sind, Profilinformationen sowie Aufenthaltsorte mitzuteilen, sei es aus der persönlichen Intention heraus, oder um im Gegenzug Services zu nutzen. Mit MSCAs sollen durch Mobile Computing Algorithmen, Wireless Technologien, und Real-time Location Systemen Personen geholfen werden, wieder eine stärkere Verbindung zu ihren physischen Communities und ihrer Umgebung zu erlangen. Beispiele, die den Einsatz von MSCAs begründen, sind Szenarien wie „Sind Freunde von mir in der Mensa / Kantine?“, „Ist jemand in meiner Umgebung der gerne Tennis spielen würde?“, „Wo verbringen Informatik Studenten auf dem Campus die meiste Zeit?“.

MobiSoc bietet eine Plattform für das Erfassen, Verwalten und Teilen von sozialen Stati physischer Communities. Der „Social State“ ergibt sich aus Daten von Personenprofilen,

<sup>29</sup><http://www.njit.edu/>; Abrufdatum: 15.10.2010

Ortsprofilen, Person-zu-Person Beziehungen sowie Person-zu-Ort Beziehungen. Dabei entwickelt er sich kontinuierlich weiter durch neue Benutzerprofile, neue soziale Verbindungen, ortsgebundene Informationen oder auch Eventinformationen, die erstellt werden. Für das Erstellen von Ortsprofilen sollen Ortsinformationen möglichst permanent über das Mobiltelefon des Benutzers o.ä. erfasst werden<sup>30</sup>.

Beziehungen zwischen Personen oder von Personen zu Orten können selbst bestimmt (konfiguriert), zusätzlich aber auch von „lernenden“ Algorithmen erkannt und erweitert werden. Ähnliche Interessen zwischen Personen können beispielsweise erkannt werden, wenn Personen häufig dieselben Orte besuchen. Diese „Social State Learning“ Algorithmen gehören zur Kernkomponente der Middleware und können auch Informationen in Personenprofilen und Ortsprofilen ergänzen, wie es in Abbildung 3.10 illustriert wird.

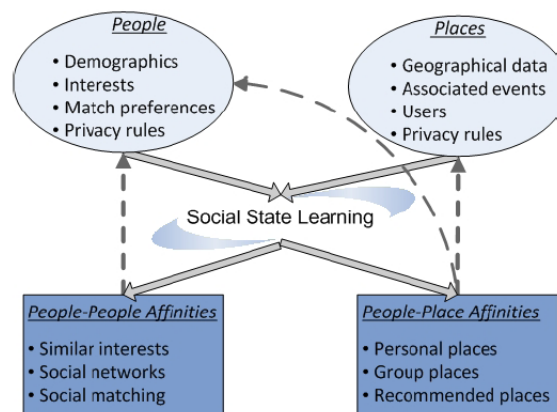


Abbildung 3.10.: „Social State“ einer Community (Quelle: Borcea u. a. (2007))

In Borcea u. a. (2007) werden zwei implementierte MSCAs, Tranzact und Clarissa, vorgestellt. Tranzact ist eine Applikation für ortsbasierte „ad hoc social collaboration“. Clients senden Anfragen, um Echtzeitinformation zu einem Ort zu bekommen. Ein Student möchte beispielsweise erfahren, welche Speisen momentan in der Mensa angeboten werden und sendet diese Anfrage über Tranzact, worauf die Applikation in den sozialen Kontakten des Studenten nach Personen sucht, die sich in der Mensa aufhalten. Nach Prüfung einiger Parameter, z.B. ob die Zielperson bereit ist, solch ein Event zu empfangen, denn eventuell möchte die Zielperson in diesem Umfeld ungestört sein, werden Request und Response über einen Eventbasierten Kommunikationsmechanismus gesendet.

Die zweite Applikation mit dem Namen Clarissa ist eine sog. „social matching applicati-

<sup>30</sup>Privatsphäre und Datenschutz soll hierbei nicht verletzt werden und wird in Borcea u. a. (2007) weiter behandelt

on“. Diese Applikation sucht nach Interessenspartnern oder Freunden im Aufenthaltsort. Beispielsweise hat ein Student zwischen zwei Vorlesungen eine zweistündige Pause und sucht nach Personen, mit denen er diese Zeit auf dem Campus verbringen kann. Die Applikation sucht nun nach Personen in den sozialen Kontakten sowie nach Personen, welche gemeinsame Interessen mit dem suchenden Studenten teilen. Die Treffer werden anschließend noch anhand einer Gewichtung der Faktoren gefiltert und sortiert. So bekommen z.B. Interessenstreffer im Bereich Sport und Musik die höchste Gewichtung. Diese Anfrage wird abschließend als Event eingetragen und kann von den potenziellen Treffern empfangen werden.



### 3.3.5. Weitere verwandte Projekte

Im Umfeld der in dieser Arbeit behandelten Themen existieren noch viele weitere sehr interessante Projekte, über die an dieser Stelle ein kurzer Überblick gegeben werden soll. Das ActiveCampus<sup>31</sup> Projekt an der University of California, San Diego, untersucht den Einsatz von mobilen Anwendungen auf dem Campus. Hier wird u.a. eine Applikation vorgestellt, bei der Studenten beim Aufenthalt auf dem Campus eine Übersichtskarte der Umgebung auf ihrem mobilen Gerät (Smartphone, PDA) angezeigt bekommen, welche angereichert ist mit Informationen um Departments, Labore, Aufenthaltsorte von Freunden und laufende Veranstaltungen zu identifizieren. Selber werden die Forschungsgebiete von ActiveCampus als context-aware und community-oriented computing beschrieben.

In Churchill u. a. (2004) wird der Einsatz von öffentlichen Bildschirmen als Kommunikationsmedium zum interaktiven Informationsaustausch in Teams, Gruppen, Organisationen und Communities untersucht. Sogenannte „Plasma Poster“ werden z.B. in den Räumlichkeiten eines Unternehmens aufgestellt und sollen von den Mitarbeitern zur Inhaltserstellung und -betrachtung genutzt werden. Dabei wurden Verhaltensmuster beobachtet wie z.B. der Führung von Konversationen vor den Plasma Postern und dem gemeinsamen Erforschen der Inhalte. Zu den am häufigsten ausgetauschten Themen am Plasma Poster zählten die Bereiche Technologien, Nachrichten und Humor.

GeoNotes<sup>32</sup> ermöglicht mithilfe von mobilen Geräten das Erstellen von virtuellen Notizen am Aufenthaltsort, welche von anderen Personen die denselben Ort zu späterer Zeit besuchen eingesehen werden können.

„ibiblio“<sup>33</sup> nutzt den Community-Gedanken um eine öffentliche Internetbibliothek mit frei verfügbarer Software und Informationen aufzubauen. Benutzer können alle vorhandenen Inhalte frei nutzen und können einen eigenen Beitrag zu den Inhalten leisten.

Mit Social Serendipity<sup>34</sup> wird die Awareness von Interessenspartnern der Umgebung gesteigert. Über eine mobile Applikation werden per Bluetooth und mithilfe von Benutzerprofilen die direkte soziale Interaktion mit Personen in naher Umgebung, welche dieselben Interessen teilen, vorgeschlagen.

---

<sup>31</sup> Siehe Griswold u. a. (2004)

<sup>32</sup> Siehe Espinoza u. a. (2001)

<sup>33</sup> <http://www.ibiblio.org/>; Abrufdatum: 27.11.2010

<sup>34</sup> Siehe Eagle und Pentland (2005)

### 3.4. Fazit

Das sehr umfangreiche und sehr aktiv geführte Forschungsgebiet der sozialen Netzwerke macht mit ihren gewonnenen Erkenntnissen eine Übertragung dieses Forschungsgebiets auf den Hochschulcampus, auf welchem das hohe Aufkommen von Informationen nur suboptimal kommuniziert wird, interessant. Bei der Weitergabe von Informationen auf dem Campus werden die Inhalte ungefiltert und ohne Berücksichtigung individueller Interessen dargestellt. Weitere Kritikpunkte bezogen auf den mangelnden Einsatz des Web 2.0-Gedankens auf dem Campus, wie der fehlenden Verknüpfung der in den Hochschulen publizierten Inhalte, verurteilen die Hochschule als innovationshemmend. Der Einsatz von Social Computing Konzepten, mit dem verbundenen Ausbau von sozialen Beziehungen, kann dem entgegenwirken. Inhalte können nach Interessen gefiltert werden, Personen mit selben Interessen, welche sich häufig am selben Ort aufhalten, werden aufgedeckt, und kollaborierendes Arbeiten kann technisch unterstützt werden. Die Awareness für interessante Personen und Inhalte vor Ort kann durch eine stärkere Vernetzung und Aufbereitung der digitalen Daten erzielt werden und gleichzeitig zu einer erhöhten sozialen Interaktion auf dem Campus führen. Potenzial bei der Steigerung dieser Awareness und der sozialen Interaktion bietet zusätzlich die Integration technischer Installationen in den sozialen Kontext, wie sie z.B. bei den CommunityMirrors und den Plasma Postern praktiziert werden. Während bei vielen der momentan genutzten digitalen Kommunikationskanäle Inhalte im Hintergrund verschwinden und ein aktives Abrufen der Daten notwendig macht, wird bei den (halb-)öffentlichen Bildschirmen die periphere Wahrnehmung und das zufällige Entdecken von Inhalten gefördert und eine schnelle und einfache Interaktion mit den Inhalten ermöglicht.

Beim Einsatz von Social Networking Plattformen ist auch die Motivation ein wichtiger Erfolgsfaktor um eine rege Beteiligung und somit positive Netzwerkeffekte zu erzielen. Konzepte zur Steigerung der Motivation wurden z.B. beim Community Equity Model von Sunspace aufgezeigt.

## 4. Systemskizze

Die Analyse aus Kapitel 3 hat gezeigt, wie wichtig die Kommunikation auf dem Campus ist und wie stark frequentiert Kommunikationsmittel wie z.B. das Schwarze Brett sind. Der effiziente Umgang mit der Kommunikation und der Darstellung von Informationen wird auf den Hochschulcampus kaum praktiziert, trotz vorhandener Techniken und Konzepte, die bereits seit längerer Zeit aus Web 2.0, Social Computing und verwandten Themen in anderem Umfeld, z.B. in Unternehmen, erfolgreich eingesetzt werden. In Unternehmen werden verstärkt Tools aus der Enterprise 2.0-Forschung eingesetzt, um Kommunikation, Kreativität und Innovation zu fördern.

Mit dem Campus-Brett wird die Grundidee des Schwarzen Bretts aufgegriffen, um mithilfe neuer technischer Mittel seine Schwächen zu minimieren sowie erweiterte Einsatzmöglichkeiten zu erschließen. Hierfür sollen aktuell erfolgreiche technische Lösungen wie z.B. Touchscreens und Smartphones zum Einsatz kommen und Konzepte aus Social Computing in das Campus-Brett mit einfließen. Im weiteren Verlauf dieser Arbeit wird der Begriff „Campus-Brett“ synonym für die technische Installation, einem touchbasierten Wandbildschirm sowie für die Software, die auf der Installation läuft, benutzt.

Studenten sollen mit dem Campus-Brett die Möglichkeit erhalten sich auszutauschen, Wissen zu teilen und eine bessere Übersicht über die Aktivitäten an der Hochschule zu bekommen. Der Benutzer hat zwei Möglichkeiten das Campus-Brett einzusetzen um sein Problem oder sein Anliegen zu lösen:

1. Durch visuelle Erforschung direkt am Campus-Brett.
2. Durch Erstellung von Notizen, welche im Campus-Brett sichtbar werden.

Zusätzlich zur gezielten Suche nach Informationen ist ein Ziel des Campus-Bretts, dass das System automatisch nach Lösungen von Problemen bzw. Anliegen einer Person sucht. Das System sucht dabei in einem sozialen Netzwerk nach Personen, welche bei den angepinn-ten Problemen / Anliegen Hilfestellung leisten könnten und informiert diese, dass auf dem Campus-Brett eine Notiz erstellt wurde, die für sie interessant sein könnte.

Bei der visuellen Erforschung am Campus-Brett wird der Student unterstützt durch gefilterte Inhalte anhand von Benutzerprofilen. Befindet sich ein Student im 1. Semester des

Informatik-Studiums, so interessiert er sich nicht für eine Lerngruppe von Elektrotechnikern im 5. Semester. Werden bestimmte Anzeigen im Detail betrachtet, so sollen Anzeigen mit ähnlichen Inhalten sichtbar werden. Auf diese Weise bekommt der „erforschende“ Student schnell eine Übersicht über die Inhalte, für die er sich interessiert, und erlebt zusätzlich durch eine intuitive Touch-Bedienung und visuelle Aufbereitung der Inhalte eine Art „Entdeckungstour“.

Bei selbst erstellten Notizen werden, wie bereits zuvor erwähnt, zusätzlich zur Darstellung auf dem Campus-Brett automatisch Personen informiert, die sich für die Notiz interessieren könnten. Erstellt Student X eine Notiz mit dem Titel „Suche Spieler für Hobbyfußballmannschaft“, wird Student Y, der in seinem Profil „Fußball“ in seinen Interessen eingetragen hat, über diese neue Notiz z.B. per E-Mail oder per Push Notification auf sein Smartphone benachrichtigt.

## 4.1. Campus-Brett-Szenarien in Hochschulen

Zur Verdeutlichung der Ideen für das Campus-Brett werden nachfolgend Beispielszenarien aufgezeigt und erläutert.

### 4.1.1. Informationen für unterwegs

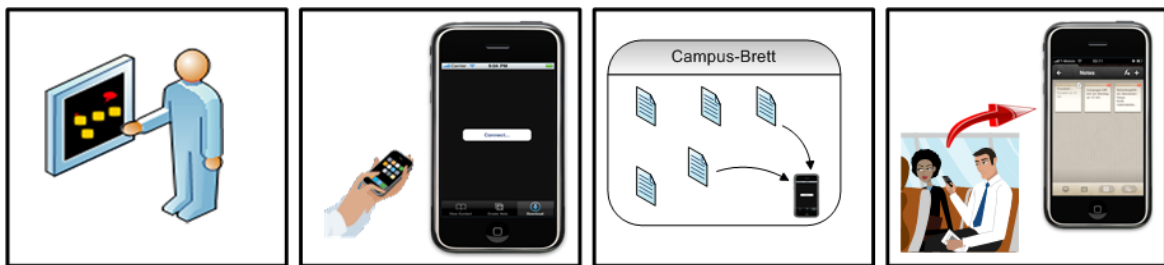


Abbildung 4.1.: Content To Go. Inhalte können vom Campus-Brett auf mobile Geräte übertragen werden

Beim Stöbern im Inhalt des Campus-Bretts stößt der Student auf einen interessanten Artikel über ein Forschungsprojekt an seiner Hochschule. Zur Interaktion mit dem Campus-Brett startet der Student die „Campus-Brett Applikation“ auf seinem Smartphone. Zwischen dem Campus-Brett und dem Smartphone wird automatisch eine Verbindung über Bluetooth oder Wifi aufgebaut. Am Campus-Brett wird nun sein angemeldetes Smartphone sichtbar (siehe Abbildung 4.1, Bild 3). Der Student zieht nun über die Touch-Bedienung am Wandbildschirm den gesichteten Artikel auf das Smartphone-Symbol, wodurch ein Kopiervorgang zum Smartphone eingeleitet wird. Zusätzlich zu diesem Artikel bemerkt er noch eine Zahlungserinnerung für die aktuell fälligen Studiengebühren. Die Notiz, auf der die Überweisungsdaten notiert sind, wird ebenfalls auf das Smartphone „gezogen“.

Den Artikel, welcher nun auf seinem Smartphone gespeichert ist, liest er auf dem Heimweg in der U-Bahn. Zuhause angekommen, startet er sein Online Banking und überträgt die Daten für die Überweisung der Studiengebühren aus der Zahlungserinnerungsnotiz.

### 4.1.2. Personalisierter Content

Auf dem Campus-Brett werden nach einem Zufallsprinzip abwechselnd verschiedene Inhalte gleichzeitig angezeigt. Ein Student meldet sich per Studentenausweis am Campus-Brett an

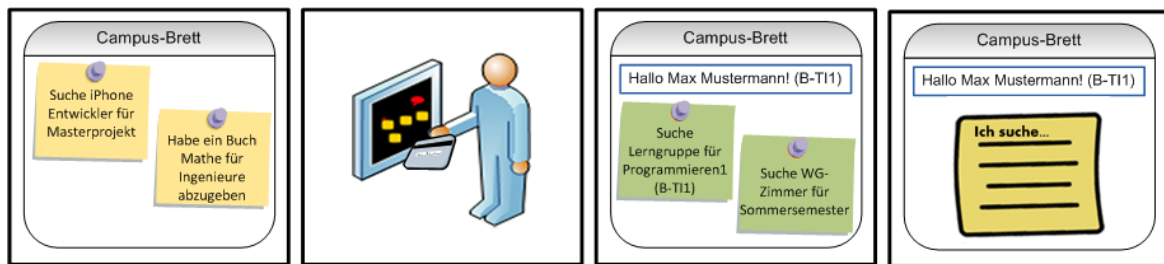


Abbildung 4.2.: Inhalte werden durch Personenprofile gefiltert

und wird mit seinem Namen begrüßt. Durch ein hinterlegtes Profil sind einige Eckdaten des Studenten bekannt: Er befindet sich im 1. Semester im Studiengang „Technische Informatik“, belegt die Fächer Programmieren 1, Elektrotechnik 1 etc. Sein Profil hat er um die Information „Habe freies WG-Zimmer“ erweitert.

Im Vordergrund dargestellte Inhalte werden nach Anmeldung an das persönliche Profil angepasst. Durch die Authentifikation per Studentenausweis ist er nun auch befugt, neue Inhalte zu erstellen. Da er noch ein Notebook für sein Studium sucht, kann er direkt mit der Suche beginnen.

### 4.1.3. Notification bei Aktualisierungen

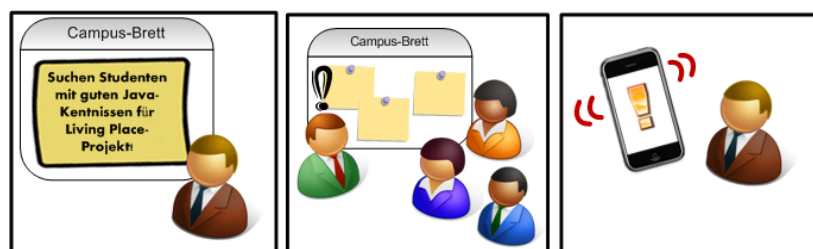


Abbildung 4.3.: Veränderungen an Anzeigen können nachverfolgt werden und z.B. per Push Notification auf Smartphones erscheinen

Auf der Suche nach Studenten, die an einem Projekt der Hochschule mitarbeiten möchten, erstellt Student X eine Notiz auf dem Campus-Brett. Im Vorbeigehen entdecken einige Studenten diese. Student Y ist sehr interessiert daran und aktiviert die Funktion „Ich bin dabei!“. Student Z hat auch Interesse, aber wenig Zeit. Er kommentiert die Notiz: „Wie groß wird der Zeitaufwand sein?“.

Das Feedback auf die erstellte Notiz erhält Student X per Push Notification direkt auf sein Smartphone. Er sieht den Kommentar von Student Z und schickt ihm eine Antwort über sein Smartphone.

#### 4.1.4. Erstellen von Inhalten außerhalb der Hochschule

Ähnlich wie beim bloggen oder twittern erstellt Person X Inhalte über das Smartphone wie z.B. „Ich komme heute zu spät“ oder „Ich bin heute krank“. Personen, die mit Person X verknüpft sind, erhalten diese Nachricht direkt auf ihr Smartphone oder auf dem Startbildschirm des personalisierten Campus-Bretts. Verknüpfte Personen sind z.B. alle Studenten eines Dozenten.

Geht es in der Benachrichtigung um die Abwesenheit eines Dozenten, so landet die Nachricht standardmäßig priorisiert auf dem Startbildschirm der Campus-Brett-Installation nahe des Vorlesungsraums.

## 4.2. Sichtbarkeit der Inhalte

Wie bereits in der Analyse erwähnt, ist die auffällige Präsenz in der Öffentlichkeit einer der Erfolgsfaktoren des Schwarzen Bretts. Inhalte sind im Vorbeigehen sichtbar und können von mehreren Personen gleichzeitig vor Ort betrachtet und diskutiert werden.

Angeregt von den Ideen der CommunityMirrors<sup>1</sup>, soll das Campus-Brett in Form von interaktiven Wandbildschirmen, anders als bei herkömmlichen digitalen Schwarzen Brettern, bei denen die Inhalte mit eingeschränkter Sichtbarkeit an Desktoprechnern betrachtet werden, als eine Art „Informationsstrahler“ in den sozialen Kontext integriert und Inhalte sichtbar, greifbar und erlebbar gemacht werden. In den Inhalten kann z.B. beim Warten auf den Fahrstuhl oder während Pausenzeiten in der Kaffee-Ecke ohne definiertes Ziel gestöbert werden. Zudem wird durch die öffentliche Positionierung der Wandbildschirme der aktive und gemeinschaftliche Austausch vor dem Bildschirm gefördert<sup>2</sup>. Das Campus-Brett kann somit ähnlich positive Effekte wie das Schwarze Brett erzielen und hat zusätzlich die Chance, im Vergleich zum Schwarzen Brett das „Erleben“ von Inhalten durch touchbasierte Interaktionen noch attraktiver zu gestalten.

---

<sup>1</sup>Siehe Abschnitt 3.3.1

<sup>2</sup>Vgl. Ott u. a. (2010b) (S. 682)

### 4.3. Die 6 Komponenten von Enterprise 2.0 Technologien

Inhalte in Form von Anzeigen sind der zentrale Bestandteil des Campus-Bretts. Inhalte werden von Studenten erstellt, strukturiert abgespeichert und auf dem Campus-Brett visuell dargestellt. In McAfee (2006b) werden die sechs Komponenten von Enterprise 2.0 Technologien vorgestellt und dafür das Akronym „SLATES“ verwendet. SLATES steht für Search, Links, Authoring, Tags, Extensions und Signals und beschreibt die Technologien, die eine Enterprise 2.0- bzw. Web 2.0-Lösung beinhalten muss. Auch das Campus-Brett soll die SLATES erfüllen, welche im folgenden näher erläutert werden. Erläuterungen zu den Begriffen wurden z.T. aus McAfee (2006b) übernommen.

- **Search**

In einer Informationsplattform müssen Informationen, nach denen gesucht wird, auffindbar sein. Layout und Navigationshilfen können dabei helfen, Informationen schneller zu finden. Untersuchungen haben jedoch ergeben, dass Anwender verstärkt Keyword-Suchen bevorzugen. Dies gilt auch für Portale im Intranet. Eine Befragung von Forrester Research hat ergeben, dass bei Intranet-Portalen ohne Suchfunktion mehr als die Hälfte der Befragten Probleme hatte, die gesuchten Informationen zu finden<sup>3</sup>. Auch beim Campus-Brett kann mit zunehmender Beteiligung die Informationsmasse schnell zunehmen und eine Keyword-Suche somit einen zusätzlichen Komfort leisten.

- **Links**

Links bieten die Möglichkeit, Inhalte zu strukturieren und wichtige bzw. beliebte Inhalte zu identifizieren. Beispielsweise wären die „besten“ Seiten jene, zu denen am häufigsten verlinkt wurde.

Im Campus-Brett sollen die Anwender nicht die Möglichkeit erhalten, Inhalte zu verlinken. Stattdessen bekommen Inhalte einen „Wert“. Inhalte werden durch die Anzahl von Aktivitäten (Views, Comments, Follows) und ein Bewertungssystem strukturiert, um somit „beliebte“ / „prominente“ Inhalte in den Vordergrund zu bringen. Ein genauerer Entwurf dieses Modells wird in Abschnitt 5.2.1 behandelt.

- **Authoring**

Ein Kernbestandteil von Web 2.0-Anwendungen ist die Erstellung von Inhalten durch ihre Anwender. Internet Blogs und Wikipedia haben gezeigt, dass Internetnutzer das Bedürfnis haben sich mitzuteilen und Inhalte zu verfassen. Obwohl im Umfeld einer Hochschule viele Inhalte von Studenten verfasst werden, existieren häufig keine Plattformen auf denen sie ihre Inhalte veröffentlichen und diese mit ihren Kommilitonen austauschen können.

---

<sup>3</sup>Vgl. McAfee (2006b) (S. 23)



Das Campus-Brett soll den Studenten die Möglichkeit geben auf möglichst einfache Art und Weise Inhalte zu erstellen. Dabei soll das Layout der Anzeigen, ähnlich wie beim traditionellen Schwarzen Brett, individuell gestaltbar sein. Zusätzlich soll es möglich sein, fertige Anzeigen als Bilder und PDF-Dateien im Campus-Brett zu publizieren.

- **Tags**

Eine Befragung von Forrester Research zeigte, dass Firmen-Intranet-Nutzer sich häufig eine bessere Kategorisierung von Inhalten wünschen<sup>4</sup>. Durch das Anbringen von Tags, einfachen kurzen Beschreibungswörtern, durch die Benutzer entsteht eine Kategorisierung der Inhalte, welche sich über die Zeit entwickelt. Dieses Kategorisierungssystem wird „folksonomy“ genannt und ist das Gegenteil von der „taxonomy“, bei welcher ein von Experten entwickeltes Kategorisationsschema vorgegeben wird. Gegenüber den „taxonomies“ haben „folksonomies“ den Nachteil, dass sie üblicherweise nicht mehrstufig sind und eventuell Redundanzen aufweisen können. Der wesentliche Vorteil ist jedoch, dass sie die Informationsstrukturen und Beziehungen reflektieren, wie sie von den Benutzern gesehen werden.

Im Campus-Brett sollen Benutzer die Möglichkeit erhalten, Tags an ihren Anzeigen anzubringen. Tags erhöhen die Sichtbarkeit der Inhalte, es werden Beziehungen zwischen Anzeigen mit selben / ähnlichen Tags hergestellt und die angebrachten Tags werden auch von der Suchfunktion gefunden. Eine Tag Cloud deckt zudem häufig eingesetzte Tags und somit „beliebte“ Anzeigen auf<sup>5</sup>.

- **Extensions**

Unter Extensions wird hier das Generieren von Inhaltsvorschlägen verstanden. Ein prominentes Beispiel liefert das Versandhaus Amazon<sup>6</sup>, bei welchem Algorithmen eingesetzt werden, um den Besuchern Vorschläge im Stile von „Wenn Sie dies mögen, interessieren Sie sich vielleicht auch für diese Artikel“ zu generieren.

Das Campus-Brett stellt Verbindungen zwischen den Anzeigen her. Anzeigen mit ähnlichen Inhalten, selben Kategorien oder auch z.B. mit Autoren derselben Semestergruppe haben eine gewisse „gewichtete“ Verbindung zueinander. So können Vorschläge generiert und dem Benutzer zu einem schnelleren Überblick über „interessante“ Anzeigen verholfen werden.

Eine weitere Zusatzfunktion soll Anzeigen anhand von Benutzerprofilen filtern. In

---

<sup>4</sup>Vgl. McAfee (2006b) (S. 24)

<sup>5</sup>Siehe Abschnitt 4.5.2

<sup>6</sup><http://www.amazon.de>; Abrufdatum: 24.09.2010

einer Profilübersicht werden Daten erfasst wie z.B. Studiengang, Semestergruppe, belegte Kurse etc., welche im Optimalfall automatisiert aus Verzeichnisdiensten der Hochschule gezogen werden, so dass der Student bei seiner ersten Anmeldung über seinen Studentenausweis ein vorkonfiguriertes Profil vorfindet. Dieses soll zusätzlich mit einer Liste von Interessen erweitert werden, welche der Benutzer in Form von Tags eintragen kann. All diese Informationen dienen einem „Smart Filter“-Algorithmus als Input zur Generierung einer Übersicht von „interessanten“ Vorschlägen für den jeweils eingeloggtten Benutzer. (Mehr dazu in Abschnitt 4.5.2.)

- **Signals**

Trotz Suchfunktionen, Inhaltsvorschlägen und Kategorisierungssystemen könnten Benutzer aufgrund einer schnell steigenden Informationsmasse Schwierigkeiten bekommen, alle Neuigkeiten in den für sie interessanten Anzeigen zu verfolgen. Nachzuschauen, ob neue Inhalte zum Thema „Smart Home“ verfasst oder Kommentare an eigenen Anzeigen angebracht wurden, kann zu einem Vollzeitjob werden. Daher sollen Benutzer informiert (signalisiert) werden, wenn neue „interessante“ Inhalte auftauchen. Diese „Signale“ werden beispielsweise in Form von E-Mail-Benachrichtigungen oder RSS<sup>7</sup> realisiert. Beim Campus-Brett sollen zusätzlich Push Notifications an das angemeldete Smartphone des Studenten gesendet werden, welche den neuen Textinhalt anzeigen und auf die neue / editierte Anzeige, welche auch auf dem Smartphone darstellbar ist, weiterleiten sollen.

## 4.4. Taxonomie

Aus den in der Analyse festgestellten Themen am Schwarzen Brett (siehe Abschnitt 3.1.3) kann für das Campus-Brett eine Einteilung in Kategorien vorgenommen werden. Eine Kategorisierung strukturiert die Inhalte, macht sie leichter auffindbar und ermöglicht es, ähnliche Inhalte durch den Einsatz von Algorithmen miteinander zu verknüpfen. Beliebte Themen am Schwarzen Brett sind das Anbieten und Suchen von Wohnungen / WG-Zimmern sowie das Anbieten und Suchen von Gegenständen. Hier lässt sich bereits eine verschachtelte Struktur ausmachen. Zusammen mit den eigenen Überlegungen, welche Themen am Campus-Brett ausgetauscht werden sollen, ergibt sich die in Tabelle 4.1 dargestellte Taxonomie.

---

<sup>7</sup>RSS dient zur strukturierten Veröffentlichung von Änderungen auf Webseiten. Nachrichten sind in einem standardisierten XML-Format aufgebaut und bestehen, ähnlich wie bei einem Nachrichtenticker, aus kurzen Informationsblöcken, die aus einer Schlagzeile mit kurzem Textanriss und einem Link zur Originalseite bestehen (siehe <http://de.wikipedia.org/wiki/RSS>; Abrufdatum: 25.09.2010).

<b>Ebene 1</b>	<b>Ebene 2</b>	<b>Ebene 3</b>
Angebot	- Wohnung/Zimmer  - Buch  - Nachhilfe      - Sprachpartner      - Mitfahrgelegenheit  - Sonstiges  - ...	- Mathe  - Elektrotechnik  - ...   - Englisch  - Spanisch  - ...
Gesuch	s.o. (selbe Struktur wie Angebot)	
Projekt	- Informatik      - Elektrotechnik    - BWL  - ...	- INF-M1  - INF-M2  - ...   - ...   - ...
Lerngruppe	- Informatik  - Elektrotechnik  - ...	- ...  - ...
Klausuren	- ...	- ...
Jobs		
Sonstiges		

Tabelle 4.1.: Taxonomie im Campus-Brett

## 4.5. Interaktionen mit dem Campus-Brett

Ein Erfolgsfaktor des altbewährten Schwarzen Bretts ist die Einfachheit. Anzeigen können ohne großen Aufwand auf einem Stück Papier erstellt und an das Schwarze Brett gepinnt werden. Bei der Umsetzung des Campus-Bretts muss diese Anforderung ebenfalls berücksichtigt werden, denn ein zu großer Einsatzaufwand würde die Akzeptanz erheblich senken. Interaktionen am Campus-Brett sollen unkompliziert sowie unterhaltsam sein. Auch bei [Sledgianowski und Kulviwat \(2008\)](#) werden die einfache Nutzbarkeit und die „Playfulness“ als wichtige Erfolgsfaktoren von Social Networking Software identifiziert.

Interaktionen am Campus-Brett sollen erleichtert werden durch intuitive Bedienmöglichkeiten per Touchscreen. Die Intuitivität und hohe Akzeptanz von Touch-Bedienungen haben vor allem Anwendungen auf dem iPhone, wie z.B. dem Blättern in Fotos oder dem Zoomen mit zwei Fingern in Google Maps, verdeutlicht. Weitere Interaktionen, welche die Bedienung am Campus-Brett vereinfachen und zu einem „Erlebnis“ machen sollen, sind das Zusammenklicken von Anzeigen nach dem Baukasten-Prinzip zur Minimierung von Schreibarbeiten und die Anmeldung per Studentenausweis, welche im weiteren Verlauf dieser Arbeit genauer erläutert werden.

### 4.5.1. Authentifikation am Campus-Brett

Für den Anmeldevorgang am Campus-Brett ist zusätzlich zum herkömmlichen Login-Vorgang mit Benutzernamen und Passwort eine vereinfachte Authentifizierung vorgesehen. Eine „kontaktlose“ Anmeldung per Mitglieds- / Studentenausweis oder auch per Smartphone soll die Hürden umständlicher Login-Formalitäten und das lästige Einprägen von Passwörtern umgehen sowie Akzeptanz und Motivation beim Einsatz des Campus-Bretts steigern.

Die Studentenausweise an der HAW Hamburg, welche im April 2009 einheitlich eingeführt wurden, verfügen über einen integrierten RFID-Chip, der sich mit einem RFID-Lesegerät eindeutig identifizieren lässt. Das Auslesen der RFID-Tags geschieht dabei kontaklos mithilfe elektromagnetischer Wellen.

Eine Authentifikation per Smartphone könnte mit dem Herstellen einer Verbindung über Bluetooth oder Wi-Fi durchgeführt werden. Identifiziert wird das persönliche Gerät über die Erkennung von eindeutigen Attributen wie z.B. Geräteerkennung, Betriebssystemversion, Geräte-ID, IMEI und ICCID. Eine Möglichkeit der Authentifizierung unterstützt beispielsweise das iPhone mit dem Simple Certificate Enrollment Protocol (SCEP), welches eine kabellose Anmeldung von Identitätszertifikaten auf dem iPhone ermöglicht. Für diesen Vorgang

navigiert der Benutzer initial über sein iPhone zu einer URL und wird dann über eine HTTP-Authentifizierung oder über bestehende Verzeichnisdienste bestätigt. Sobald der Benutzer authentifiziert ist, werden Konfigurationsprofile und Sicherheitszertifikate zwischen iPhone und Server ausgetauscht und somit ein sicherer Anmeldevorgang ermöglicht<sup>8</sup>.

#### 4.5.2. Die Startseite

Die Exploration am Campus-Brett beginnt auf einer Startseite, welche in einer Übersicht mehrere angepinnte Anzeigen gleichzeitig darstellt. Diese Startseite wird in Abbildung 4.4 skizziert und setzt sich zusammen aus einer Pinnwand, einer Suchfunktion, Filterfunktionen, einer Tag Cloud, sowie einer Profilübersicht. Die Profilübersicht ist nur bei eingeloggtem Zustand sichtbar und wird ansonsten ausgeblendet.

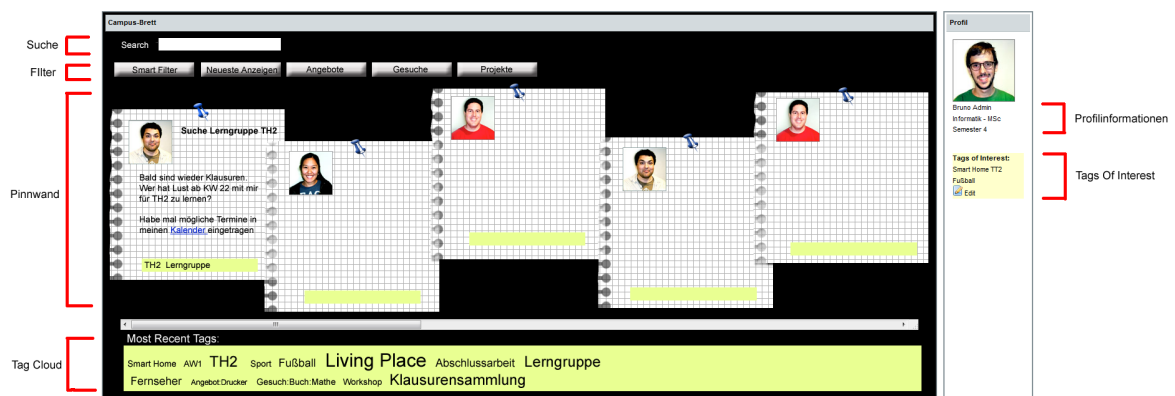


Abbildung 4.4.: Campus-Brett-Startseite

Die rot markierten Bereiche werden nachfolgend im Detail erläutert.

#### Suche

Die Keyword-Suche ist direkt über die Startseite erreichbar und wird gut sichtbar im oberen Bereich platziert. Das Starten einer Suche aktualisiert die Anzeigen auf der dargestellten Pinnwand.

<sup>8</sup>Eine detailliertere Beschreibung mit Implementierungshinweisen ist verfügbar unter [http://images.apple.com/chde/iphone/business/docs/iPhone\\_OTA\\_Enrollment\\_Configuration.pdf](http://images.apple.com/chde/iphone/business/docs/iPhone_OTA_Enrollment_Configuration.pdf); Abrufdatum: 23.11.2010

## Filter

Die in der Datenbank gespeicherten Anzeigen können anhand verschiedener Funktionen gefiltert werden. Ist ein Student beispielsweise auf der Suche nach aktuellen Projekten an der Hochschule, so sind Inhalte zu diesem Thema schnell erreichbar über einen Klick auf den Filter „Projekte“. Der „Smart Filter“ sucht potentiell interessante Anzeigen für den eingeloggteten Benutzer anhand von Profilinformationen und der „Tags Of Interest“. Der Algorithmus gestaltet sich in etwa wie folgt:

```
1 user = getActiveUser();
2 tagsOfInterest = user.getTagsOfInterest();
3
4 entriesTagMatches = getEntriesWithTags(tagsOfInterest);
5 results.add( sort(entriesTagMatches) );
6
7 similarUser= getUsersWithSimilarSemester(
8                                     user.getCourse(), user.getSemester());
9 for similarUser from i=0 to i=similarUser.length();
10 DO user = similarUser[i];
11     entriesAuthorMatches = getEntriesWithAuthor(user)
12     results.add( sort(entriesAuthorMatches) );
```

Die Anzeigen, die nach diesem Algorithmus in der „results“-Liste landen, werden vor der Darstellung in einem weiteren Algorithmus sortiert. Dies geschieht bei jedem Aufruf neu und enthält einen gewissen Zufallsfaktor, so dass ein Serendipity-Effekt, also das zufällige Entdecken von „interessanten“ Anzeigen, gefördert wird. Der Sortieralgorithmus wird in Abschnitt [5.2.1](#) näher beschrieben.

## Pinnwand

Ähnlich wie beim traditionellen Schwarzen Brett sind die Anzeigen „chaotisch“ angeordnet und überlappen sich teilweise. Automatisch hinzugefügte Profilfotos an den Anzeigen sorgen für eine „sympathische“ Authentizität. Auf den Anzeigen, welche möglichst individuell gestaltbar sein sollen, sind am unteren Rand die vergebenen Tags sichtbar. Auf einer Multitouch-Oberfläche kann auf der Pinnwand per Zieh- bzw. Schiebebewegung frei durch die Anzeigen navigiert werden.

## Tag Cloud

Die Darstellung einer Tag Cloud auf der Startseite erhöht die Sichtbarkeit von prominenten Themen am Campus-Brett. Die Größe der Schrift visualisiert die Prominenz eines Tags. Je beliebter ein Tag ist, desto größer wird er in der Tag Cloud dargestellt.

## Profilinformationen

Meldet sich ein Student am Campus-Brett an, erscheint im rechten Bereich eine Profilübersicht. Die Profilinformationen sind vom eingeloggtten Benutzer frei editierbar. Hierzu gehören:

- Name
- Geburtsdatum
- Studiengang
- Semester

## Tags Of Interest

Anhand von Tags kann der Benutzer seine Interessen definieren. Die im Profil eingetragenen „Tags Of Interest“ sind als eine Art Abonnement von Themen zu verstehen und werden von Algorithmen wie z.B. dem Smart Filter genutzt, um den Benutzer über Themen zu informieren, die für ihn interessant sind.

### 4.5.3. Die Detailansicht

Bei Interesse an einer Anzeige kann diese per Fingertipp / Klick in der Detailansicht betrachtet werden. Diese Ansicht wird in Abbildung 4.5 skizziert. Auch hier werden die rot markierten Bereiche im Folgenden kurz erläutert.

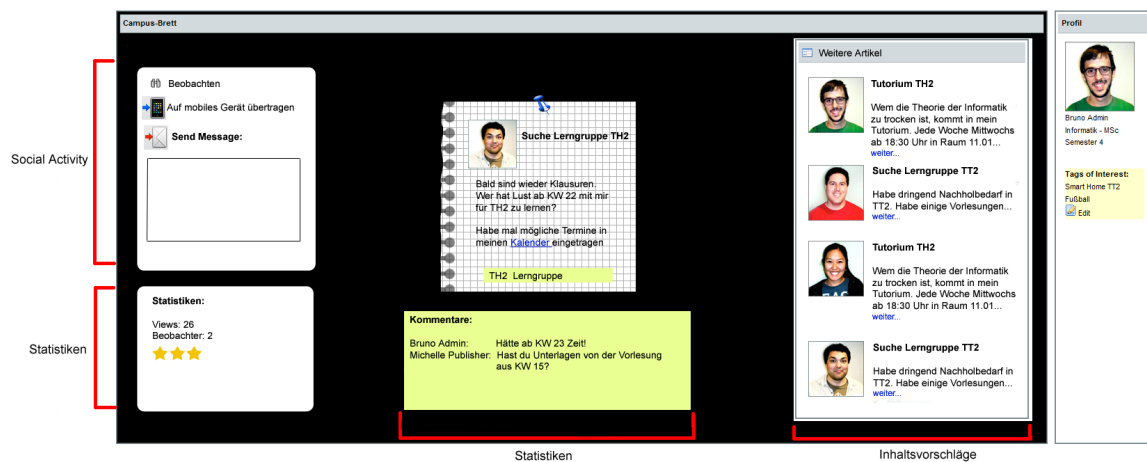


Abbildung 4.5.: Detailansicht einer Anzeige im Campus-Brett

## Social Activity

Aktivitäten am Campus-Brett werden registriert und spielen eine Rolle bei der Bestimmung zur Beliebtheit einer Anzeige, auf die z.B. im Sortieralgorithmus (siehe Abschnitt 5.2.1) zurückgegriffen wird. Der Benutzer kann in diesem Feld mehrere soziale Aktivitäten in Bezug auf die betrachtete Anzeige ansteuern. Zu den hier dargestellten Aktivitäten zählen:

- **Beobachten**  
Wird eine Anzeige beobachtet, wird der Benutzer über Änderungen an der Anzeige informiert (per E-Mail, Push Notification etc.). Hierzu zählen auch neu verfasste Kommentare.
- **Auf mobiles Gerät übertragen**  
Stellt eine Verbindung zu einem Smartphone her und macht die Anzeige auf dem Smartphone offline verfügbar.
- **Send Message**  
Sendet eine private Nachricht an den Ersteller der Anzeige.

## Statistiken

Statistiken zu einer Anzeige werden im linken Bereich dargestellt und zum Teil grafisch, z.B. durch Sterne bei der Bewertung, aufbereitet. Diese Übersicht gibt dem Benutzer Aufschluss über Beliebtheit und Qualität der Anzeige.



## Kommentare

Jede Anzeige verfügt über ein Feld für Kommentare. Interessiert sich ein Benutzer für eine Anzeige oder hat er Rückfragen, so kann er sich im Kommentarfeld zu Wort melden. Kommentare dienen zur Kontaktaufnahme mit dem Autor sowie den Interessenten. Sie werden bei neu verfassten Kommentaren durch einen Benachrichtigungsdienst darüber in Kenntnis gesetzt. Die Benachrichtigung erfolgt über eine einfache E-Mail und / oder eine Push Notification auf das Smartphone. Kommentare an Anzeigen sind für alle Benutzer ohne Einschränkung sichtbar.

## Inhaltsvorschläge

In der Detailansicht einer Anzeige werden im rechten Bereich Inhaltsvorschläge generiert. Anzeigen werden durch folgende Faktoren verknüpft, welche jeweils eine Gewichtung haben:

- gleiche / ähnliche Tags
- gleiche / ähnliche Kategorien
- gleicher Autor

Die Faktoren zur Erkennung von „ähnlichen“ Anzeigen werden auch in dieser Reihenfolge gewichtet. Anzeigen mit ähnlichen Tags haben z.B. eine höhere Chance, in den Inhaltsvorschlägen zu landen als Anzeigen in der gleichen Kategorie, aber ohne ähnliche Tags. Aufgrund des Platzmangels und aus Gründen der übersichtlichen Darstellung soll hier nur eine kleine Auswahl von ähnlichen Anzeigen dargestellt werden, welche die „besten“ Treffer bei der Suche nach ähnlichen Anzeigen widerspiegelt. Die „besten“ Treffer werden in Abschnitt [5.2.1](#) beschrieben.

Der Benutzer kann nach Auswahl eines Vorschlags diesen in der Detailansicht betrachten, auf welcher wieder neue Inhaltsvorschläge sichtbar werden. So kann der Benutzer von Anzeige zu Anzeige springen und mehrere Anzeigen mit ähnlichen Inhalten „entdecken“.

### 4.5.4. Unterstützte Anzeigenerstellung

Das Verfassen von längeren Texten am Touch-Bildschirm, z.B. über eine virtuell eingeblendete Tastatur, ist, im Vergleich zur Eingabe über eine haptische Tastatur, nur bedingt komfortabel. Aus diesem Grund soll anhand der erkannten wiederkehrenden inhaltlichen und

gestalterischen Merkmale der Anzeigen am Schwarzen Brett (siehe Abschnitt 3.1.3) ein Modul entwickelt werden, welches das Erstellen von Anzeigen am Campus-Brett erleichtert. In Abbildung 4.6 ist die Idee zur halbautomatisierten Anzeigenerstellung skizziert.

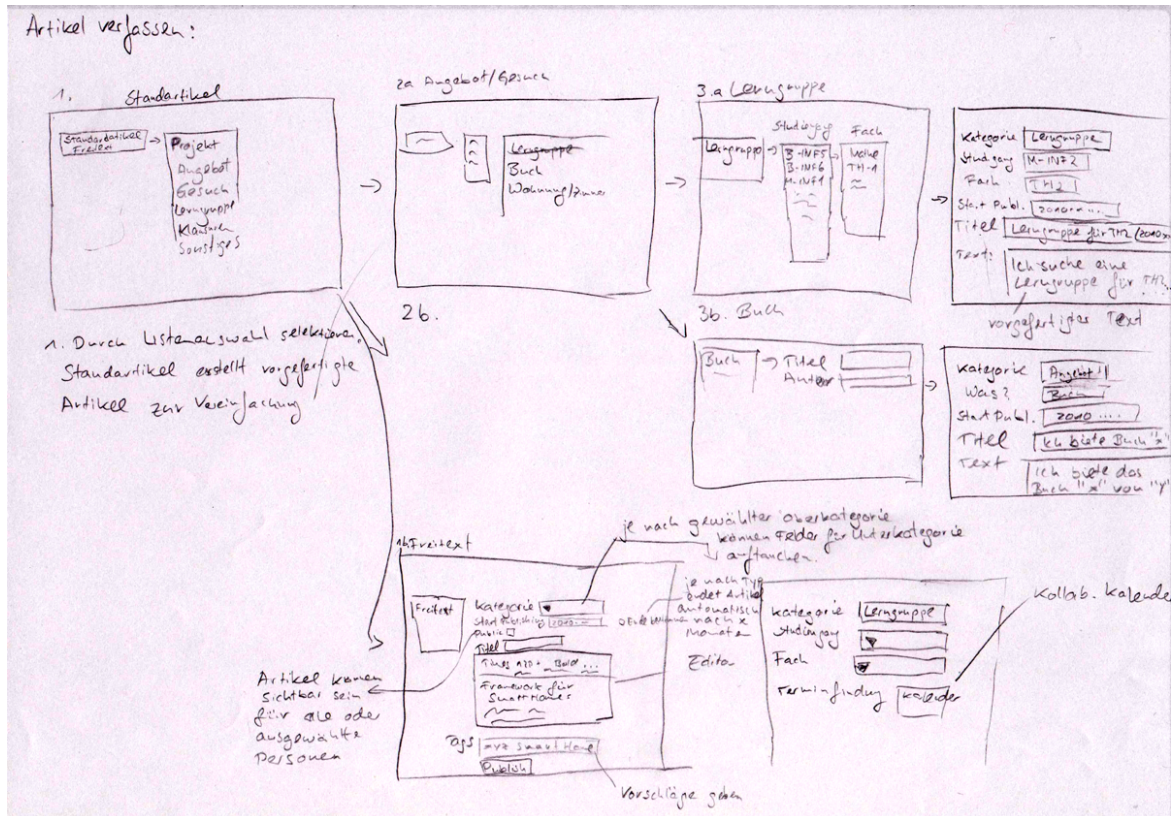


Abbildung 4.6.: Skizze zur Funktionsweise beim Erstellen von Anzeigen

Der Benutzer entscheidet anhand der vorgegebenen Taxonomien, zu welchem inhaltlichen Typ die zu erstellende Anzeige gehört. Zu Beginn werden nur die Hauptkategorien zur Auswahl dargestellt und je nach Auswahl werden nach und nach die Unterkategorien dazu aufgeklappt (siehe Abbildung 4.6, (1)->(2a)->(3a)). Nach Erreichen der untersten Ebene wird eine automatisch erstellte Anzeige angezeigt. Als Beispiel könnte bei der Auswahl „Suche->Sprachpartner->Spanisch“ ein Standardtext wie „Hallo, ich suche eine/n Sprachpartner/in um mein Spanisch aufzubessern. Ich biete deutsch, meldet euch!“ generiert werden. Zusätzlich könnte die Darstellung einer spanischen Flagge die Anzeige grafisch unterstützen.

## 4.6. Belohnungssystem

Zur Steigerung der Motivation soll, ähnlich wie bei Konzepten aus Sunspace<sup>9</sup>, ein „Schaukasten“ mit errungenen Abzeichen in der Profilübersicht der Benutzer angezeigt werden. Benutzer des Campus-Bretts können durch ihre Aktivitäten sogenannte Achievements, ähnlich wie aktuell häufig bei Computerspielen eingesetzt, erringen und diese für andere sichtbar machen. Beispiele für Abzeichen wären z.B. „Frühaufsteher“ (bei mehreren aufeinanderfolgenden Logins vor 9 Uhr), „Streber“ (wenn sich der Benutzer eine Woche lang jeden Tag einloggt) oder „Profi-Seller“ (bei Erreichen von insgesamt 100 selbstverfassten Angeboten). Um die erreichbaren Abzeichen sichtbar zu machen sollen diese in einer Übersichtsseite, mit ihrer detaillierten Beschreibung, angezeigt werden, und so einen Anreiz für eine erhöhte Aktivität am Campus-Brett geben. Zusätzlich sollen „unsichtbare“ Abzeichen (Hidden Achievements) eine zusätzliche Motivationssteigerung durch den Überraschungseffekt erzielen.

## 4.7. Public vs Private

Die Präsentation von persönlichen Daten an halböffentlichen Orten bringt Probleme bei der Wahrung der Privatsphäre mit sich. Steht ein Benutzer im eingeloggten Zustand vor dem Wandbildschirm ist es eher unerwünscht, dass das komplette Profil den umgebenden Personen sichtbar gemacht wird. Zudem ist jede Person in unterschiedlichem Umfang bereit persönliche Daten preiszugeben. Auch wenn der Benutzer beim Einsatz des Campus-Bretts, wie auch beim Einsatz eines Schwarzen Bretts, einen Teil seiner Privatsphäre zwangsweise aufgeben muss, z.B. bei der Preisgabe von Kontaktdaten, muss der Benutzer Kontrolle darüber haben welche Informationen preisgegeben werden. Diese Anforderungen sollen berücksichtigt werden durch die Realisierung des Campus-Brett in 2 verschiedenen Modi, einem Public- und einem Private-Mode, und der Konfigurierbarkeit der preisgegebenen Daten.

Das Campus-Brett läuft auf dem öffentlichen Wandbildschirm im Public-Mode. In diesem Modus sollen private Daten, z.B. die Profilübersicht des eingeloggten Benutzers, nicht angezeigt werden. Diese sollen wiederum im Private-Mode, welcher bei der Nutzung am Desktop-Rechner aktiv ist, dargestellt und editierbar sein. Die 2 verschiedenen Modi unterscheiden sich nicht nur in der Präsentation, sondern auch beim Funktionsumfang. So ist z.B. das Eingeben von längeren Texten über eine haptische Tastatur komfortabler als die Eingabe über eine virtuelle Tastatur am Touch-Bildschirm<sup>10</sup>. Aus Gründen des verbesserten Komforts

---

<sup>9</sup>Siehe [Reiser \(2009a\)](#)

<sup>10</sup>Siehe auch [Ott u. a. \(2009\)](#) (S. 110)

sollen daher bestimmte Interaktionen, wie das Erstellen von Anzeigen, hauptsächlich am Desktop-Rechner durchgeführt werden.

## 4.8. Übersicht der Anforderungen und Systemwahl

Zu den Kernfunktionalitäten des Campus-Bretts gehören das Erstellen und Publizieren von Anzeigen. Für den Umgang mit Inhalten, Erstellen, Editieren, Löschen etc., wird ein Content-Management-System (CMS) benötigt. Zudem sollen mehrere Benutzer in verschiedenen Rollen (Student, Professor, Mitarbeiter) und Gruppen (Studiengang, Semester) Zugriff auf diese Daten haben. Um unnötige Programmierarbeiten zu vermeiden, soll für diese Funktionalitäten auf ein existierendes CMS aufgebaut werden. Heutige Content-Management-Systeme unterstützen die genannten Anforderungen und bieten häufig einen weitaus größeren Funktionsumfang.

Für die Umsetzung des Campus-Bretts wurde die Portallösung Liferay Portal (siehe Grundlagen in Abschnitt 2.3) ausgewählt. Vor der Evaluierung dieses Produkts sollen noch einmal alle Anforderungen zusammengefasst werden:

- Inhalte erstellen, speichern, editieren, publizieren und löschen
- User Management
  - Benutzerrollen vergeben
  - Benutzer in Gruppen einteilen
- Suchfunktion
- Aktivitäten überwachen (Views, Downloads etc.)
- Anzeigen bewerten
- Inhalte taggen
- Inhaltsvorschläge generieren (Inhalte verknüpfen)
- Benachrichtigungen bei Aktualisierungen
- Touchbasierte Navigation
- Authentifikation per RFID-Ausweis

Die in Liferay Portal integrierten Social Computing Werkzeuge ermöglichen die Bildung von Benutzergruppen und rollenbasierte Authorisation. Benutzer können je nach Gruppe / Rolle verschiedene Sichten und Zugriffe auf Inhalte und Services bekommen. Das integrierte CMS gibt Unterstützung bei diversen Veränderungen der Inhalte bis hin zur Versionierung. Inhalte

in Liferay Portal können mit den zwei eingebauten Search Engines, Apache Lucene<sup>11</sup> und Apache Solr, gefunden werden.

Weitere interessante Punkte im Funktionsumfang von Liferay Portal sind u.a. (aus [Yuan \(2009\)](#)):

- WYSIWYG Editing
- Dynamic Tagging (ermöglicht Taggen von Webinhalten, Dokumenten etc.)
- Control Panel (ermöglicht zentrales Administrieren von Inhalten, Benutzern, Communities, Rollen)
- Activity tracking
- Announcements and Alerts (Broadcasting von Nachrichten an verschiedene Gruppen und Benutzer)

Liferay Portal deckt einen Großteil der gestellten Anforderungen ab und eignet sich somit als Grundlage für die Umsetzung des Campus-Bretts. Die Tatsache, dass dieses Produkt komplett Open-Source angeboten wird, gibt zusätzlich die Möglichkeit, falls notwendig, Anpassungen auf jeder Ebene, z.B. am Benutzermodell, durchzuführen.

---

<sup>11</sup><http://lucene.apache.org>; Abrufdatum: 25.09.2010

## 5. Systementwurf und Realisierung

Die in der Systemskizze erarbeiteten Konzepte und Anforderungen an das Campus-Brett sollen nun in einen technischen Systementwurf überführt, und anschließend anhand dieses Entwurfs realisiert werden. Für diese Schritte wird auf die Open-Source Portallösung Liferay Portal (siehe Abschnitt 2.3) zurückgegriffen.

### 5.1. Model View Controller Architektur

Für die Umsetzung mit Liferay Portal wird das Campus-Brett als Struts Portlet entwickelt. Mit dem Open-Source Framework Apache Struts<sup>1</sup> wird die Entwicklung von Java Webapplikationen unterstützt und die Model View Controller (MVC) Architektur verfolgt, welche die Datenbankzugriffe, das Seitenlayout und den Kontrollfluss voneinander trennt. Das Modell repräsentiert dabei den Business- bzw. den Datenbank-Code, die View kümmert sich um die Darstellung der Daten und die Entgegennahme von Benutzerinteraktionen, während der Controller Benutzeraktionen auswertet und die Abläufe steuert.

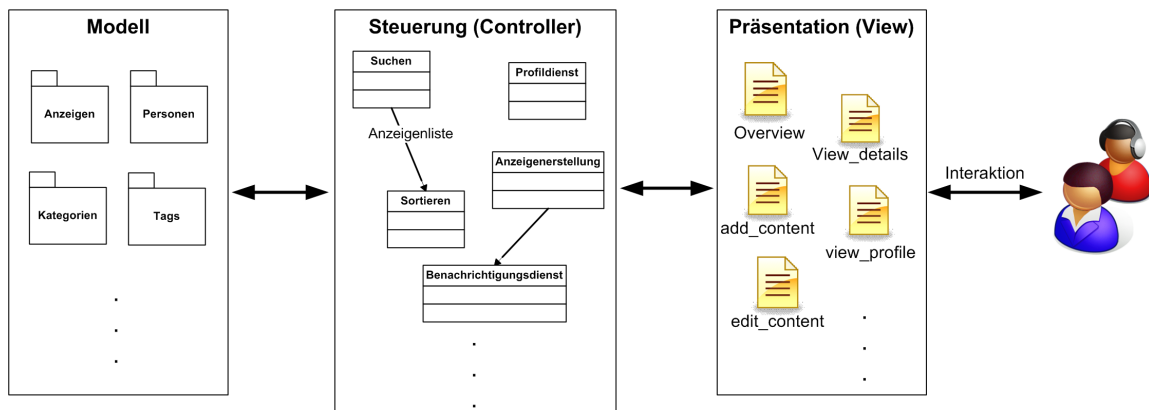


Abbildung 5.1.: Model View Controller-Konzept im Campus-Brett

<sup>1</sup>Siehe <http://struts.apache.org/>; Abrufdatum: 26.09.2010

Abbildung 5.1 illustriert die Verteilung dieser Aufgaben im Campus-Brett. Der Benutzer interagiert mit der Präsentationsschicht, z.B. auf der in Abschnitt 4.5.2 erwähnten Startseite. Interaktionen werden von der Präsentationsschicht (View) verarbeitet und an die Steuerung zur Auswertung weitergeleitet. Die Steuerung startet Anfragen an die Modellschicht, erstellt Ergebnisse und / oder löst eine Weiterleitung in der Präsentationsschicht aus. Als Beispiel aktiviert ein Benutzer im eingeloggtten Zustand auf der Startseite den Button „Smart Filter“. Diese Aktion wird im Controller von einem Filtermodul verarbeitet, welches eine Anfrage an die Modellschicht startet. Über diese Anfrage hat die Steuerung Zugriff auf die Profildaten des eingeloggtten Benutzers. Nach Auswertung der Interessen des Benutzers liefert eine erneute Anfrage an die Modellschicht eine Liste von Anzeigen. Diese Liste wird in der Steuerungsschicht vom Sortiermodul (siehe Abschnitt 5.2.1) weiterverarbeitet und als Ergebnis von der Präsentationsschicht dargestellt.

## 5.2. Das Datenmodell

Anzeigen am Campus-Brett gehören immer genau einem Autor und verfügen über einen Inhalt bestehend aus Text und evtl. Bildern. Ergänzt um all die in der Systemskizze vorgestellten Konzepte, ergibt sich das ERM aus Abbildung 5.2.

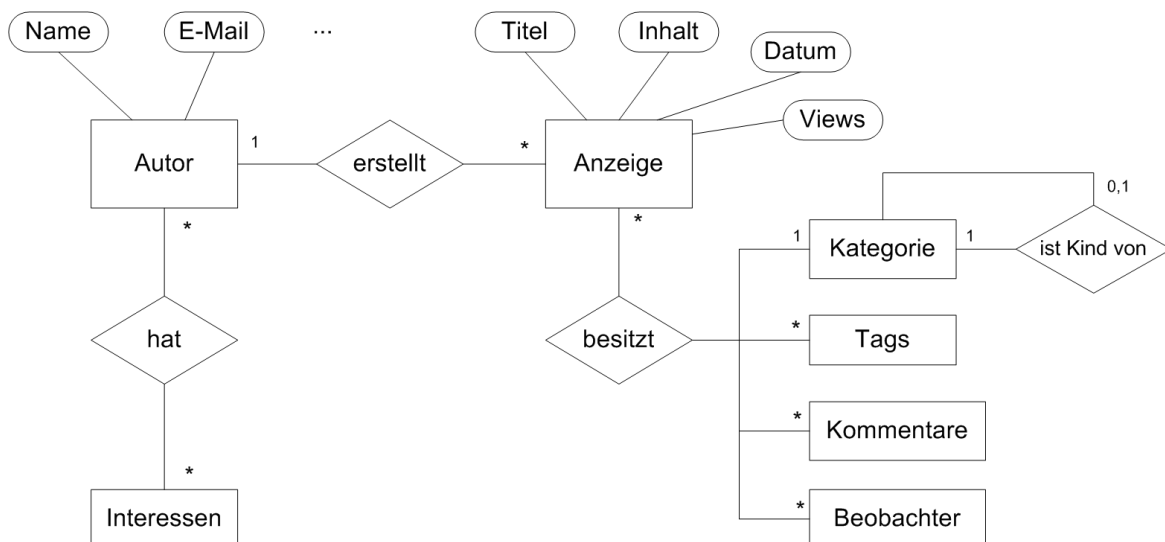


Abbildung 5.2.: Entity Relationship Modell des Campus-Bretts

Eine Anzeige befindet sich in einer in den Taxonomien definierten Kategorie. Durch die Verschachtelung des Kategorisierungsschemas kann dies eine Unterkategorie von meh-

renen übergeordneten Kategorien sein. Weiter ist eine Anzeige mit den Objekten Tags, Kommentare und Beobachter verknüpft.

Ein Autor kann über mehrere Interessen verfügen, welche im Campus-Brett genutzt werden um Inhaltsvorschläge zu generieren (siehe Abschnitt 4.5.3).

Die Objekte Person und Anzeige werden in Abbildung 5.3 in einem UML-Klassendiagramm, in leicht vereinfachter Form, dargestellt. Der Autor, welcher über die Anzeige direkt erreichbar ist, ist eine Person welche konkret einen der 3 Personentypen Professor, Student oder Mitarbeiter beschreibt. Je nach Personentyp verfügt diese Person über verschiedene Attribute. Beispielsweise verfügen alle Studenten über die Attribute Studiengang und Semester, über welche sie untereinander in Verbindung gebracht werden können.

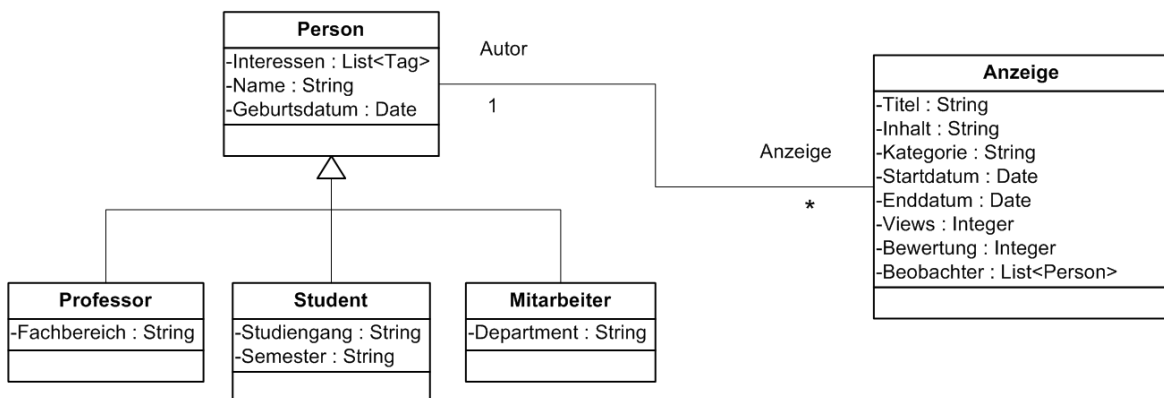


Abbildung 5.3.: UML Klassendiagramm für Personen und Anzeigen im Campus-Brett

An dieser Stelle könnte das Modell weiter entwickelt werden um z.B. bessere Verbindungen zwischen Professoren und Studenten herstellen zu können. Studenten könnten sich bei Vorlesungen eintragen, bei welchen Professoren sich wiederum als Dozent eintragen usw. Diese Mechanismen werden im weiteren Verlauf dieser Arbeit jedoch nicht konkretisiert, stattdessen wird lediglich der Einsatz des Campus-Bretts durch Studenten betrachtet.

### 5.2.1. Sortierung / Wert der Anzeigen

Sollen Anzeigen auf dem Campus-Brett dargestellt werden, sorgt ein Algorithmus für die Sortierung der Anzeigen. Die Sortierung erfolgt zum Zeitpunkt des Seitenaufrufs und ist aufgrund eines eingebauten Zufallsfaktors variabel. Die Reihenfolge der Anzeigen wird anhand eines ermittelten numerischen Werts festgelegt, der die Beliebtheit, die Relevanz und die



Qualität der Anzeige widerspiegelt. Je höher dieser Wert, desto weiter vorne landet die Anzeige in der Darstellung. Diese „Wertigkeit“ einer Anzeige wird als *severity* bezeichnet und berechnet sich wie folgt:

$$severity = activities\_severity + ratings\_severity + (weight\_severity * random) - age \quad (5.1)$$

### Activities Severity

Aktivitäten an einer Anzeige erhöhen die Wertigkeit. Registrierte Aktivitäten sind:

- Views
- Downloads
- User Feedback (Kommentare)

Bei allen 3 Faktoren gilt, je höher die Anzahl, desto höher die Beliebtheit der Anzeige. Dies könnte z.B. folgende Berechnung hervorrufen:

$$activities\_severity = \frac{views}{100} + downloads + comments \quad (5.2)$$

### Ratings Severity

Anzeigen können von Benutzern mit bis zu 5 Sternen bewertet werden. Die Bewertung spielt eine wichtige Rolle bei der Identifizierung von qualitativ hochwertigen Anzeigen. Die Anzahl der Bewertungen spielt ebenfalls eine entscheidende Rolle bei der Repräsentativität der Bewertung und soll in der Berechnung die Gewichtung verstärken. Die Bewertung kann die Gewichtung einer Anzeige auch negativ beeinflussen, wenn sie schlechter als 2.5 Sterne ausfällt.

$$ratings\_severity = (rating - 2.5) * rating\_count \quad (5.3)$$

### Weight Severity

Wird eine Liste von Anzeigen sortiert, wurde diese Liste aus einer Gemeinsamkeit bei den Tags, in den Kategorien oder beim Autor zusammengestellt. Diese Faktoren bekommen eine unterschiedliche Gewichtung, da z.B. Anzeigen mit selben Tags interessanter sind als Anzeigen des gleichen Autors. Diese Gewichtung kann zum Zeitpunkt des Sortierens frei vergeben werden. Eine mögliche Verteilung wäre:

- TAGS\_WEIGHT = 0.5
- CATEGORY\_WEIGHT = 0.4
- AUTHOR\_WEIGHT = 0.1

Ein zufällig generierter Multiplikationsfaktor sorgt für eine gewisse Variation bei der Sortierung durch diese Faktoren. So haben auch Treffer in den Autoren eine Chance vor den Treffern bei den Tags zu landen.

### Age

Die Wertigkeit einer Anzeige wird mit steigendem Alter reduziert. Dieser Faktor soll nur einen kleineren Einfluss auf den Wert einer Anzeige haben. Beispielsweise könnte der Werteverfall verzögert und / oder um einen Faktor abgeschwächt werden.

$$\text{if}(\text{age} > 14)\{\text{age} = \frac{\text{age} - 14}{2}\} \quad (5.4)$$

## 5.3. Umsetzung als Struts Portlet

Das Campus-Brett wird als Struts Portlet in Liferay Portal entwickelt. Wie bereits zu Anfang dieses Kapitels erwähnt, implementiert Struts das MVC-Konzept, wodurch die Präsentation von der Geschäftslogik getrennt wird. Zudem unterstützt Struts in der Präsentationsschicht ein zentrales Seitenfluss-Management, welche die Skalierbarkeit erhöht und die Modularisierung des Coding-Prozesses ermöglicht. Details hierzu werden in Abschnitt 5.3.2 behandelt.

### 5.3.1. ServiceBuilder zur Beschreibung des Datenmodells

Liferay Portal stellt ein Tool namens ServiceBuilder zur Verfügung, welches die Bereitstellung von Java Services erleichtert. Der ServiceBuilder automatisiert für Portlets die Erstellung von Interfaces und Klassen zur Beschreibung eines Datenmodells und der Datenbankinteraktion. Auf die bereitgestellten Java Services kann auf verschiedene Weisen zugegriffen werden, so z.B. lokal vom Java Code oder remote über Web Services<sup>2</sup>. Der ServiceBuilder wird über eine XML-Datei gesteuert und generiert SQL-Code, Java Beans, Hibernate Configuration, Spring Configuration, Axis Web Services, JSON JavaScript Interfaces und mehr.

Das Campus-Brett verwaltet Einträge die von Benutzern erstellt werden. Das Datenmodell für diese Einträge ist in der Entwurfsphase in Abschnitt 5.2 entwickelt worden. In der Implementierungsphase wurde eine Anzeige mit dem Synonym `CBEntry` benannt. Ein Eintrag `CBEntry` besteht aus Metadaten, wie z.B. `category`, `userid` (des Autors) und `createDate`, sowie aus den Feldern `title` und `content` die den Inhalt beschreiben (vgl. Abbildung 5.3). Mithilfe des ServiceBuilders werden Interfaces und Klassen generiert, welche die Arbeit mit diesem Datenmodell erleichtern. Der ServiceBuilder wird über die Datei `service.xml` angesteuert. Die Definition des Anzeigenmodells durch die `service.xml` ist im Anhang in Listing A.1 zu finden.

In Abbildung 5.4 wird das Klassendiagramm mit den vom ServiceBuilder generierten Klassen, einschließlich ihrer Basisklassen, dargestellt. Die in gelb markierten Klassen werden nur beim ersten Durchlauf des ServiceBuilders generiert. Weitere Durchläufe, z.B. bei Veränderungen des Datenmodells, überschreiben diese Dateien nicht. In diesen Klassen findet die Implementierung des Verhaltens statt. Beispielsweise wird in der `CBEntryPersistenceImpl` mit der `update`-Methode definiert, welche Daten in die Datenbank geschrieben werden sollen. Die tatsächliche Kommunikation mit der Datenbank wird in der Persistenzschicht automatisch abgewickelt.

---

<sup>2</sup>Siehe auch Yuan (2009) (S. 78ff)



Die wichtigsten Klassen aus diesem Modell werden nachfolgend näher erläutert.

### **com.liferay.portal.model.BaseModel**

Dieses Interface ist das Basis-Interface aller Modelle in Liferay. Hierüber kann identifiziert werden, ob es sich beim aktuellen Objekt um ein neues handelt, oder ob es sich bereits in der Persistenzschicht befindet. Es dient somit zur Entscheidungsfindung, ob ein neuer Eintrag erstellt, oder lediglich ein `update` auf einen vorhandenen Eintrag durchgeführt werden muss.

### **com.ext.portlet.campusBrett.model.CBEntryModel**

Dieses Interface dient als Repräsentation eines Eintrags in der Persistenzschicht und definiert Methoden für die direkte Interaktion mit der Datenbank.

### **com.ext.portlet.campusBrett.model.impl.CBEntryImpl**

In dieser Klasse werden Hilfsmethoden für die im Campus-Brett erstellten Anzeigen implementiert. Sie beinhaltet z.B. eine Methode um den Textinhalt einer Anzeige, welche in der Datenbank mit Layoutbeschreibung als HTML-Code gespeichert ist, zu extrahieren und diesen für die Vorschau der Inhaltsvorschläge mit verändertem Layout aufzubereiten.

### **com.ext.portlet.campusBrett.service.impl.CBEntryLocalServiceImpl**

Diese Klasse implementiert die Dienste die für die Interaktion mit Anzeigen angeboten werden sollen. Zu diesen Diensten zählen:

- Anzeigen erstellen (`addEntry`)
- Anzeigen editieren (`updateEntry`)
- Anzeigen abrufen (`getEntry`)
- Anzeigen löschen (`deleteEntry`)
- Anzeigen suchen (`search`)

Die `reIndex`-Methode aktualisiert die Inhalte für die Search Engine. Diese muss bei jeder inhaltlichen Veränderung der Anzeige ausgeführt werden. Beim Indexing werden Zeichenfolgen die für die Suchfunktion unnötig sind, wie z.B. Interpunktion, Pronomen und Adverbien, entfernt.

### com.ext.portlet.campusBrett.service.persistence.CBEntryPersistenceImpl

In der Persistenzschicht werden die Transaktionen mit der Datenbank abgewickelt. Diese sind durch die Beschreibung des Datenmodells mit der `service.xml` ohne weitere Codebearbeitung funktionsfähig.

### 5.3.2. Layout und Seitenfluss

Die Präsentation wird in Liferay Portal mit der Web-Programmiersprache Java Server Pages (JSP) realisiert. JSP dient zur einfachen Erzeugung von dynamischen HTML-Seiten und erlaubt es Java-Code und spezielle JSP-Aktionen in diese einzubetten. Zusätzlich zu den JSP-Aktionen stellt Liferay Portal eine eigene Tag-Bibliothek bereit. Diese Bibliothek unterstützt z.B. bei der Kommunikation mit der Steuerung (z.B. über `<liferay-portlet:renderURL>`), beim Erstellen von Bedienoberflächen (z.B. `<liferay-ui:tabs>`), und stellt viele weitere Dienste zur Verfügung (z.B. über `<liferay-util:include>`).

Abbildung 5.5 zeigt die im Campus-Brett-Projekt erstellten JSP-Seiten.

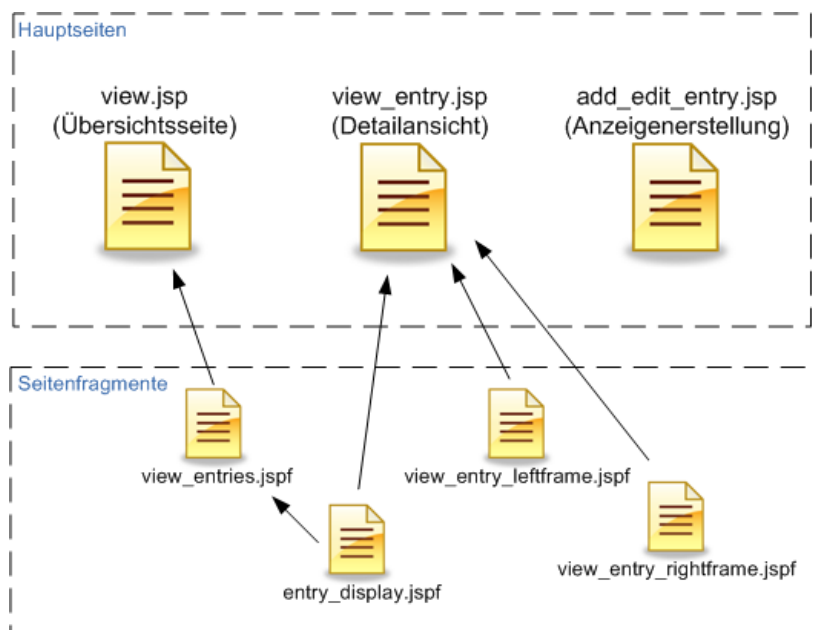


Abbildung 5.5.: JSP-Seiten für die Darstellung der Inhalte

Das Campus-Brett besteht aus 3 Hauptseiten. Der Übersichtsseite (Startseite), der

Anzeigen-Detailansicht und der Seite für die Erstellung von Anzeigen. Zur Wiederverwendbarkeit von Codeabschnitten sowie zur übersichtlicheren Gestaltung des Layoutcodes wurden einige Abschnitte in Fragmentdateien ausgelagert. So wird beispielsweise das Layout einer einzelnen Anzeige in der Datei `entry_display.jspf` beschrieben und von verschiedenen Seiten eingebunden.

Der aktuelle Entwicklungsstand<sup>3</sup> der einzelnen Seiten wird anhand von Screenshots des Systems im Anhang B illustriert.

Bei Struts Portlets werden Seitenfluss und Seitenlayout voneinander getrennt behandelt und zentral in XML-Dateien organisiert. Diese sind:

- `struts-config.xml` - Definitionen für den Seitenfluss
- `tiles-defs.xml` - Definitionen für die Seitenlayouts

Die Definitionen des Seitenflusses in der `struts-config.xml` zeigt Listing 5.1 am Beispiel der Anzeigenerstellung.

Listing 5.1: Definition des Seitenflusses in der `struts-config.xml`

```
1 <action path="/ext/campusBrett/add_edit_entry"  
2     type="com.ext.portlet.campusBrett.action.EditEntryAction">  
3   <forward name="portlet.ext.campusBrett.add_edit_entry"  
4     path="portlet.ext.campusBrett.add_edit_entry" />  
5   <forward name="portlet.ext.campusBrett.edit_entry"  
6     path="portlet.ext.campusBrett.edit_entry" />  
7   <forward name="portlet.ext.campusBrett.error"  
8     path="portlet.ext.campusBrett.error" />  
9 </action >
```

Das Erstellen und Editieren wird in einer Aktion zusammengefasst und mit der Java-Klasse `EditEntryAction` verbunden. Diese Klasse befindet sich serverseitig in der Steuerungsschicht und entscheidet ob es sich bei der ausgewählten Aktion um eine neue Anzeige handelt, oder ob eine Operation auf einer bereits vorhandenen Anzeige durchgeführt wurde. Leitet der Benutzer einen Bearbeitungswunsch auf eine Anzeige ein, findet über die Steuerung eine Weiterleitung zu `portlet.ext.campusBrett.edit_entry` statt. Diese Weiterleitung ist in Listing 5.2 definiert.

Listing 5.2: Weiterleitung durch `EditEntryAction.java`

```
1 public ActionForward render(ActionMapping mapping, ActionForm form,  
2     PortletConfig portletConfig, RenderRequest renderRequest,  
3     RenderResponse renderResponse) throws Exception {
```

<sup>3</sup>Stand 13.12.2010

```

4  try {
5      ActionUtil.getEntry(renderRequest);
6  } catch (Exception e) {
7      if (e instanceof NoSuchEntryException
8          || e instanceof PrincipalException) {
9          SessionErrors.add(renderRequest, e.getClass().getName());
10         return mapping.findForward("portlet.ext.campusBrett.error");
11     } else {
12         throw e;
13     }
14 }
15 long entryId = ParamUtil.getLong(renderRequest, "entryId");
16 // Bei vorhandener entryId wird auf edit_entry weitergeleitet
17 if (entryId > 0) {
18     return mapping.findForward(getForward(renderRequest,
19                                         "portlet.ext.campusBrett.edit_entry"));
20 } else {
21     return mapping.findForward(getForward(renderRequest,
22                                         "portlet.ext.campusBrett.add_edit_entry"));
23 }
24 }

```

Wird vom Benutzer die Aktion „Speichern“ auf einer Anzeige ausgelöst, so ruft die JSP-Seite die Aktion `/ext/campusBrett/add_edit_entry` (siehe Listing 5.3) auf, welche serverseitig von der Java-Klasse `EditEntryAction` als Aktion verarbeitet wird (siehe Listing 5.4).

Listing 5.3: Submit-Aktionen in der `add_edit_entry.jsp`

```

1 <form
2   action="<portlet:actionURL>
3       <portlet:param name="struts_action"
4           value="/ext/campusBrett/add_edit_entry" />
5       </portlet:actionURL>"
6   method="post"
7   name="<portlet:namespace />fm"
8   onSubmit="<portlet:namespace />saveEntry(false); return false;">
9
10  ...

```

Listing 5.4: Verarbeitung der Action in `EditEntryAction.java`

```

1 public void processAction(ActionMapping mapping, ActionForm form,
2                          PortletConfig portletConfig, ActionRequest actionRequest,
3                          ActionResponse actionResponse) throws Exception {

```



```
4 ...
5 // entryId ist nur vorhanden wenn entry bereits existiert. ansonsten -1
6 long entryId = ParamUtil.getLong(actionRequest, "entryId");
7 ...
8 if (entryId <= 0) {
9     // Add entry
10    entry = CEntryServiceUtil.addEntry(title, categoryId, content,
11                                       serviceContext);
12 } else {
13     // Update entry
14    entry = CEntryLocalServiceUtil.updateEntry(entryId, title, content);
15 }
16
17 PortletURL portletURL = ((ActionResponseImpl) actionResponse)
18     .createRenderURL();
19
20 portletURL.setParameter("struts_action", "/ext/campusBrett/view_entry");
21 portletURL.setParameter("entryId", String.valueOf(entry.getEntryId()));
22
23 actionResponse.sendRedirect(portletURL.toString());
24 }
```

Wie hier zu erkennen ist wird in dieser Aktion entschieden, ob in der Persistenzschicht ein neuer Eintrag erstellt werden muss, oder ob ein `update` auf einen Eintrag durchgeführt werden kann. Nach Abschluss der Operation wird der Präsentationsschicht eine Weiterleitung zu `/ext/campusBrett/view_entry` gesendet. Welche JSP-Seite letztendlich dahintersteckt ist in der `tiles-defs.xml` definiert (siehe Listing A.2).

Die in diesem Abschnitt beschriebenen Vorgänge werden zur Verdeutlichung nochmal als Sequenzdiagramm in Abbildung 5.6 dargestellt.

### 5.3.3. Inhaltsvorschläge

Auf der Detailansichtsseite (`view_entry.jsp`) werden, wie bereits in der Systemskizze in Abschnitt 4.5.3 erwähnt, zu der dargestellten Anzeige Inhaltsvorschläge generiert. Diese Inhaltsvorschläge werden auf dem Server in der Steuerung zusammengestellt und der Präsentationsschicht zur Verfügung gestellt. Dort werden sie in einem Tabellenlayout aufbereitet und dem Benutzer zur Interaktion zur Verfügung gestellt. Der Code hierfür ist in Listing A.3 einsehbar.

Serverseitig kümmert sich die Java-Klasse `CampusBrettEntryLinker` um das Finden

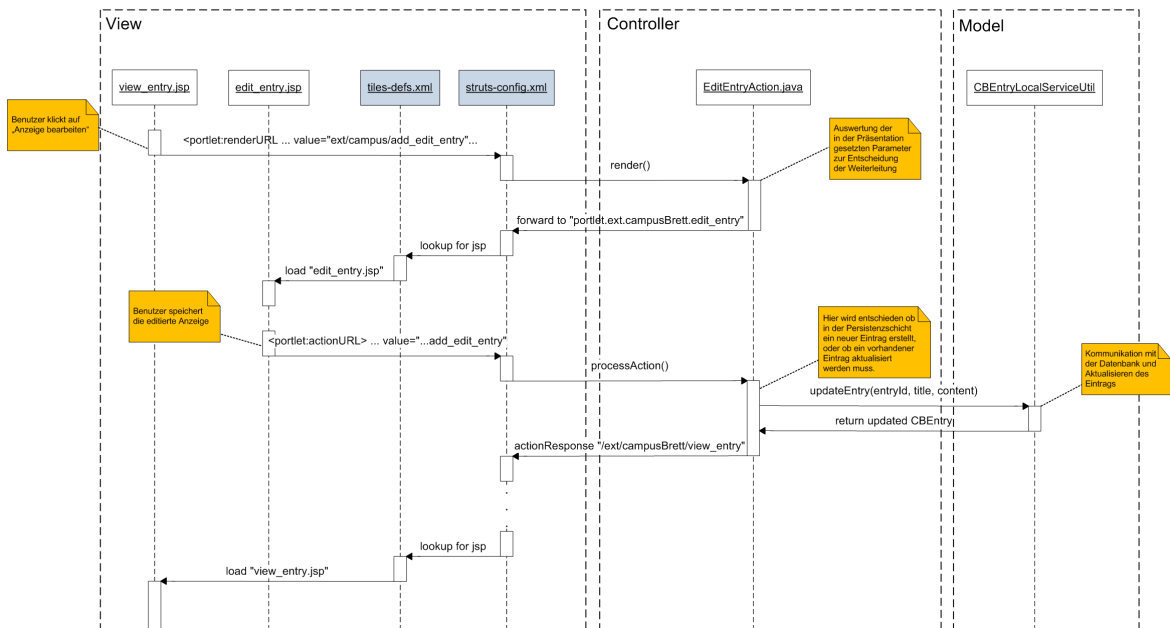


Abbildung 5.6.: Sequenzdiagramm für das Bearbeiten und Speichern einer Anzeige

und Sortieren von verknüpften Anzeigen. Gefundene Anzeigen werden in einer Liste von `CampusBrettEntryLinkerHit`-Objekten gespeichert. Diese Objekte enthalten die Anzeige inklusive ihrer Gewichtung / ihrem Sortierwert (siehe Abschnitt 5.2.1). Das Finden von „ähnlichen“ Anzeigen wird in 3 Schritten, einer Suche in den Tags, einer Suche in den Kategorien und einer Suche in den Autoren, durchgeführt. Die Funktionsweise von Tag-Suchen und Kategorie-Suchen sollen im Folgenden anhand von kurzen Code-Ausschnitten verdeutlicht werden.

### Die Tag-Suche

Zunächst müssen alle Tags aus der gegebenen Anzeige geholt werden. Diese Tags sind über die Serviceklasse `TagsAssetLocalServiceUtil` erreichbar, welche von Liferay bereitgestellt wird.

```

1 TagsAsset tagsAsset = TagsAssetLocalServiceUtil.getAsset(
2     CBEEntry.class.getName(), entry.getEntryId());

```

Die Tag-Objekte der Anzeige werden in einen String geschrieben und der Methode `searchInTags` übergeben. In dieser Methode wird der Suchvorgang über die in Liferay

integrierte Search Engine Apache Lucene gestartet. Der Einsatz dieser Search Engine ist im folgenden Listing zu sehen.

Listing 5.5: Suche nach Tags in Anzeigen

```
1 private List<CBEntry> searchInTags (String tagKeywords)
2                                     throws PortalException , SystemException {
3     BooleanQuery contextQuery = BooleanQueryFactoryUtil.create ();
4     contextQuery.addRequiredTerm (Field.PORTLET_ID, Indexer.PORTLET_ID);
5
6     BooleanQuery searchQuery = BooleanQueryFactoryUtil.create ();
7     searchQuery.addTerm (Field.TAGS_ENTRIES, tagKeywords);
8
9     BooleanQuery fullQuery = BooleanQueryFactoryUtil.create ();
10    fullQuery.add (contextQuery, BooleanClauseOccur.MUST);
11    fullQuery.add (searchQuery, BooleanClauseOccur.MUST);
12
13    Hits tagHits = SearchEngineUtil.search (entry.getCompanyId (), fullQuery,
14    QueryUtil.ALL_POS, QueryUtil.ALL_POS);
15
16    List<CBEntry> entryListTags = CBEntryLocalServiceUtil
17        .getCBEntriesWithHits (tagHits);
18    ...
19 }
```

Das Suchen in den Tags der Anzeigen wird in der Zeile

```
1 searchQuery.addTerm (Field.TAGS_ENTRIES, tagKeywords);
```

aktiviert. Das Suchen in den Tags der Anzeigen wird ermöglicht durch die Verbindung der Anzeigen zu den Tag-Funktionalitäten von Liferay, welche über die `service.xml` hergestell wurde (siehe „References“ in Listing A.1).

### Die Kategorien-Suche

Für die Suche in den Kategorien muss nicht jede Anzeige einzeln aus der Datenbank geholt und abgefragt werden. Diese Suche geschieht über eine Mapping-Tabelle die über den ServiceBuilder erstellt wurde (siehe „Relationships“ in Listing A.1). In dieser Mapping-Tabelle sind die Beziehungen zwischen Anzeigen und Kategorien definiert. Entgegen des Entwurfs für das Datenmodell aus Abschnitt 5.2 wurde hier eine many-to-many Beziehung bevorzugt, um die performante Suche auf allen Kategorieebenen zu ermöglichen. Eine Anzeige, z.B. mit der id '17418', für die Suche eines spanischen Sprachpartners sähe in dieser Mapping-Tabelle wie folgt aus:

categoryld	entryld
2	17418
6	17418
21	17418

**Kategorien:**

2 = Gesuch

6 = Sprachpartner

21 = Spanisch

Tabelle 5.1.: Beispielhafter Ausschnitt aus Mapping-Tabelle CBCategories\_CBEntries

Über die vom ServiceBuilder generierte Serviceklasse CBCategoryLocalServiceUtil können nun die Anzeigen einer Kategorie abgerufen werden. Listing 5.6 zeigt die Kategorien-Suche.

Listing 5.6: Suche nach ähnlichen Anzeigen über die Kategorien

```

1 private void searchInCategories(long categoryld, double severity)
2     throws PortalException, SystemException {
3     List<CBEntry> categoryMatches =
4         CBCategoryLocalServiceUtil.getCBEntries(categoryld);
5
6     // Jede gefundene Anzeige in die globale Hit-liste hinzufuegen
7     for (CBEntry cbEntryHit : categoryMatches) {
8         // Pruefung auf bereits vorhandenen Treffer:
9         // erneutes eintragen verhindern oder update mit hoeherer Severity??
10        if (!cbEntryList.contains(cbEntryHit)) {
11            cbEntryList.add(cbEntryHit);
12            // Hier wird Gewichtung fuer Random-Sortierung generiert
13            double randomSeverity = severity * Math.random();
14            CampusBrettEntryLinkerHit IHit =
15                new CampusBrettEntryLinkerHit(randomSeverity, cbEntryHit);
16            linkerHits.add(IHit);
17        }
18    }
19
20    // In Parent-Kategorien suchen (rekursiv)
21    CBCategory category =
22        CBCategoryLocalServiceUtil.getCBCategory(categoryld);
23    if (Validator.isNotNull(category.getParentCategoryld())) {
24        long parentCategoryld = category.getParentCategoryld();
25        // in Parent Kategorie suchen und severity um die Haelfte reduzieren
26        searchInCategories(parentCategoryld, severity * 0.5);
27    }

```

Nachdem die Kategorie-Treffer in einer Liste gespeichert wurden, wird die Methode rekursiv aufgerufen, um weitere Treffer in den Oberkategorien hinzuzufügen. Die Gewichtung der Treffer wird bei jeder höheren Stufe abgeschwächt. Somit sind direkte Kategorie-Treffer, z.B. Gesuche nach spanischen Sprachpartnern, bessere Treffer als indirekte Kategorie-Treffer, in diesem Fall Gesuche nach Sprachpartnern.

## 5.4. Fazit

Für die Realisierung von Social Networking Software existieren eine Reihe von erweiterbaren Produkten. Beispielhaft seien hier neben Liferay Portal „drupal<sup>4</sup>“, „Cyn.in<sup>5</sup>“, „Sprout Social<sup>6</sup>“, „Elgg<sup>7</sup>“ und „Mahara<sup>8</sup>“ genannt. Liferay Portal bietet eine gute Unterstützung bei der Interaktion zwischen Benutzern durch vorhandene Messaging-Funktionalitäten und integrierten Anwendungen wie Forum, Blog und Wiki. Zudem werden wichtige SLATES-Konzepte (siehe Abschnitt 4.3), wie z.B. dem Taggen von Inhalten, sowie das Erstellen von Benutzerinhalten mithilfe von konfigurierbaren WYSIWYG-Editoren, gut unterstützt.

Die von Liferay vorgegebene Model View Controller Architektur ermöglicht einen flexiblen Systementwurf, wodurch Änderungen bzw. Erweiterungen am Campus-Brett erleichtert werden. Einzelne Komponenten können wiederverwendet werden, ohne dass der Entwickler Kenntniss über andere Code-Abschnitte zu haben braucht. So könnten z.B. verschiedene Oberflächen, etwa für iPhone, Android oder Windows, implementiert werden, welche auf das selbe Datenmodell zugreifen.

In der Implementierungsphase ist sehr viel Zeit bei der Einarbeitung in die Konzepte von Liferay, z.B. dem Einsatz des ServiceBuilder's, sowie beim Gebrauch der Skript- und Programmiersprachen JSP, JavaScript, HTML und CSS, in Anspruch genommen worden. Trotz der Unterstützung die Liferay für die Präsentation z.B. über Templates und vorgefertigte Benutzeroberflächen bietet, ist die individuelle Layoutgestaltung über HTML und CSS mit sehr viel Aufwand verbunden. Hier könnte der Einsatz eines Frameworks, wie beispielsweise das Google Web Toolkit, eine bessere Unterstützung leisten. Der hohe Zeitaufwand bei der Einarbeitung in Liferay ist vor allem durch die, zum Zeitpunkt der Erstellung dieser Arbeit,

<sup>4</sup><http://www.drupal.de/>; Abrufdatum: 10.11.2010

<sup>5</sup><http://www.cynapse.com/cynin>; Abrufdatum: 10.11.2010

<sup>6</sup><http://sproutsocial.com/>; Abrufdatum: 10.11.2010

<sup>7</sup><http://elgg.org/>; Abrufdatum: 10.11.2010

<sup>8</sup><http://mahara.org/>; Abrufdatum: 10.11.2010

mangelnde Qualität und zum Teil auch dem Fehlen der Dokumentation begründet. Anleitungen für den Einstieg in Liferay sind auf der offiziellen Homepage<sup>9</sup> verfügbar, während weiterführende Dokumentationen im offiziellen Liferay-Wiki noch sehr lückenhaft präsent sind<sup>10</sup>. Hilfestellung bei der Entwicklung von Portlets in Liferay geben die in großer Anzahl integrierten Portlets (siehe Abschnitt 2.3), an welchen sich eigens implementierte Portlets orientieren können. Negativ zu bewerten ist hierbei aber der fast durchgehend unkommentierte Zustand dieser Portlets.

Das entworfene Datenmodell bietet eine gute Grundlage für das Herstellen einer erhöhten Awareness von interessanten Informationen und Personen auf dem Campus. Algorithmen können anhand dieses Modells Verbindungen zwischen Inhalten und zwischen Personen herstellen, wodurch, unterstützt durch die vorgesehene Platzierung des Campus-Bretts an halböffentlichen Orten, die Chance besteht soziale Interaktionen von Studenten auf dem Campus zu verstärken.

Ein Punkt, der im Entwurf eine genauere Betrachtung sinnvoll macht, ist die Erzielung des aktiven und gemeinschaftlichen Austauschs vor dem Bildschirm durch einen kollaborativen Einsatz. Auch wenn mehrere Personen die Informationen am halböffentlichen Wandbildschirm gleichzeitig betrachten können, ist die Interaktion am Campus-Brett momentan auf einen Single-User Einsatz beschränkt. Eine verstärkte Integration von CSCW-Konzepten zur gemeinsamen Interaktion am Bildschirm könnte das gemeinsame „Erleben“ von Inhalten verstärken.

Leider hat das in Liferay entwickelte Campus-Brett während der Erstellung dieser Arbeit, aufgrund von vielen fehlenden Komponenten, z.B. bei der Interaktion mit den Inhalten (bewerten, kommentieren, etc.) und den Personen (benachrichtigen etc.), keinen testbaren Entwicklungsstand erreicht. Aus diesem Grund bleibt eine genauere Bewertung, über den Nutzen und die Erzielung der gewollten Effekte des Campus-Bretts, an dieser Stelle offen.

---

<sup>9</sup><http://docs.liferay.com/>; Abrufdatum: 10.11.2010

<sup>10</sup>Stand: 10.11.2010

## 6. Zusammenfassung und Ausblick

In der vorliegenden Arbeit werden Einsatzmöglichkeiten von Social Computing-Konzepten im Hochschulumfeld untersucht. Dabei wird das Grundkonzept des traditionellen Schwarzen Bretts aufgegriffen und dieses mit den Erkenntnissen aus der Enterprise 2.0- und Social Computing-Forschung verknüpft.

Zu Beginn werden einige Grundlagen geklärt, u.a. die Konzepte aus CSCW und Enterprise 2.0, auf welche sich die weiteren Untersuchungen in dieser Arbeit beziehen.

Im Kapitel 3 wird das Umfeld der Hochschule und ihre kommunikative Umgebung beschrieben, und in einer umfangreichen Analyse der Gebrauch von Kommunikationsmedien auf dem Campus, mit Fokus auf dem Schwarzen Brett, untersucht. Anschließend werden Erkenntnisse aus Enterprise 2.0 und verwandten Themen analysiert und in einen Hochschulkontext übertragen. Wichtige Punkte des Social Computing, wie z.B. der Vernetzung von Inhalten und Personen, dem Ausbau von sozialen Beziehungen und der Bedeutung von losen Kontakten, werden hervorgehoben und an Beispielen erläutert. Social Networking Plattformen wie z.B. Facebook haben gezeigt, wie durch den gezielten Einsatz von sozialen Netzwerken neue Wege der Kollaboration, der Innovation und des Knowledge Sharing entstehen können.

Eine weiterführende Betrachtung von aktuellen Forschungsarbeiten in diesem Umfeld konkretisiert die Ideen und Ziele vom kollaborativen Arbeiten und dem Nutzen von sozialen Netzwerken. Einige Ideen und Konzepte aus den vorgestellten Projekten, wie z.B. der Integration von Wandbildschirmen in den sozialen Kontext zur Kommunikationsförderung (siehe Abschnitt 3.3.1) und dem Einsatz eines Belohnungssystems zur Motivationsförderung (siehe Abschnitt 3.3.3) fließen in der anschließenden Systemskizze des Campus-Bretts mit ein. Nach Aufstellung aller Anforderungen an die Plattform zur Unterstützung sozialer Interaktionen an Hochschulen, wird, nach einer kurzen Evaluierung, die Open-Source Portallösung Liferay Portal als Basis für die Realisierung ausgewählt.

Anhand der entwickelten Systemskizze werden im letzten Kapitel die Komponenten eines in Liferay Portal integrierten Campus-Bretts entworfen, und einige wichtige Schritte bei der Realisierung konkreter erläutert. Dabei wird ein Einblick in die Konzepte und die

Architektur von Liferay Portal gegeben. Das entworfene Campus-Brett dient den Studenten einer Hochschule als gemeinsame Austauschplattform auf dem Campus. Durch eingebaute Social Computing Konzepte wird die Awareness von interessanten Inhalten und Personen sowie die soziale Interaktion auf dem Campus gefördert. Ein wichtiger Erfolgsfaktor des Campus-Bretts, gegenüber anderen Social Networking Plattformen wie z.B. Facebook und StudiVZ, ist die Integration der Plattform in die reale Welt und damit ins direkte Sichtfeld der Studenten. Dadurch wird die Sichtbarkeit und die periphere Wahrnehmbarkeit der Inhalte gesteigert.

So wie die beobachteten Erfolge der Web 2.0-Entwicklung im Internet, und der erfolgreichen Übertragung der Konzepte für einen Einsatz im Unternehmensumfeld (Enterprise 2.0), birgt der Einsatz von Social Computing Konzepten ebenso große Chancen bei der Unterstützung des Hochschulalltags. Social Software kann den Aufbau sozialer Beziehungen auf dem Campus stärken, welcher nicht als ein Aufbau eines reinen Freundennetzwerks verstanden werden soll (wie viele soziale Netzwerke im Internet, allen voran Facebook), sondern als ein soziales Netzwerk von Personen die am selben Campus studieren und mit denen sie potenziell kollaborierend arbeiten können. Ein erhöhter Austausch und eine engere Zusammenarbeit zwischen Studenten einer Hochschule kann erzielt werden und wird sich nicht nur in privater Hinsicht positiv bemerkbar machen, wie beim privaten Austausch am traditionellen Schwarzen Brett, sondern auch bei der wissenschaftlichen Arbeit innovationsfördernd auswirken.

Eine gemeinsam genutzte Plattform, wie das Campus-Brett, wird nur dann von einem breiten Publikum regelmäßig genutzt, wenn ein hoher Bedienkomfort und eine gewisse Intuitivität gegeben ist. Nach der Implementierung eines fertigen Prototyps sollte deswegen im nächsten Schritt eine Usability-Studie durchgeführt werden, und eventuell Bedienkonzepte überarbeitet werden. Da die Inhalte am Campus-Brett von den Benutzern kommen, kann der tatsächliche Erfolg dieser Plattform erst ab Erreichen einer kritischen Masse identifiziert werden. Aus diesem Grund ist es sehr wichtig möglichst allen potenziellen Nutzern einen einfachen Zugriff auf die Plattform zu ermöglichen. Eine Möglichkeit zur Unterstützung dabei wäre die automatische Erstellung eines Zugangs mit vordefiniertem Profil direkt nach der Immatrikulation, sowie direkter Zugangsmöglichkeit per Studentenausweis. Weitere Möglichkeiten zum schnelleren Erreichen der kritischen Masse ist der Einsatz eines Belohnungssystems zur Steigerung der Motivation, welche in dieser Arbeit bereits angerissen wurde, jedoch weiter ausgearbeitet werden müsste.

Handelt es sich bei den ausgetauschten Inhalten am Campus-Brett um Wissensinformationen, wird eine Betrachtung des hochschulübergreifenden Informationsaustauschs interessant. Durch hochschulinterne Plattformen welche sich mit Plattformen anderer Hochschulen vernetzen lassen, könnten vielversprechende Synergieeffekte, die sonst nicht



sichtbar waren, entstehen. Bei solch einer hochschulübergreifenden Vernetzung könnten z.B. ähnliche Forschungsprojekte an anderen Hochschulen identifiziert und eine Zusammenarbeit angefragt werden.

# A. Quellcodes

Listing A.1: `service.xml` mit Beschreibung des Datenmodells für den ServiceBuilder

```
1 <!DOCTYPE service-builder
2     PUBLIC "-//Liferay//DTD Service Builder 5.2.0//EN"
3         "http://www.liferay.com/dtd/liferay-service-builder_5_2_0.dtd">
4 <service-builder package-path="com.ext.portlet.campusBrett">
5     <!-- eindeutiger identifizier fuer die Klassennamen -->
6     <namespace>CB</namespace>
7     <!-- Entity entspricht einer Tabelle in der Datenbank -->
8     <entity name="CBEntry" uuid="true" local-service="true"
9         remote-service="true">
10        <!-- PK fields -->
11        <column name="entryId" type="long" primary="true" />
12        <!-- Group instance -->
13        <column name="groupId" type="long" />
14        <!-- Audit fields -->
15        <column name="companyId" type="long" />
16        <column name="userId" type="long" />
17        <column name="userName" type="String" />
18        <column name="createDate" type="Date" />
19        <column name="modifiedDate" type="Date" />
20        <!-- Other fields -->
21        <column name="categoryId" type="long" />
22        <column name="title" type="String" />
23        <column name="content" type="String" />
24        <column name="viewCount" type="int" />
25        <column name="displayDate" type="Date" />
26
27        <!-- Relationships -->
28        <column name="categories" type="Collection"
29            entity="CBCategory" mapping-table="CBCategories_CBEntries" />
30
31        <!-- Order -->
32        <order by="desc">
33            <order-column name="createDate" />
34        </order>
```

```
35     <!-- Finder methods -->
36     <finder name="CategoryId" return-type="Collection">
37         <finder-column name="categoryId" />
38     </finder>
39
40     <!-- References -->
41     <reference package-path="com.liferay.counter" entity="Counter" />
42     <reference package-path="com.liferay.portal" entity="User" />
43     <reference package-path="com.liferay.portlet.tags"
44         entity="TagsAsset" />
45     <reference package-path="com.liferay.portlet.tags"
46         entity="TagsEntry" />
47 </entity>
48 </service-builder>
```

Listing A.2: Layoutdefinition in der `tiles-defs.xml`

```
1 <definition name="portlet.ext.campusBrett" extends="portlet">
2 </definition >
3
4 <definition name="portlet.ext.campusBrett.view"
5     extends="portlet.ext.campusBrett">
6     <put name="portlet_content" value="/portlet/ext/campusBrett/view.jsp" />
7 </definition >
8
9 <definition name="portlet.ext.campusBrett.add_edit_entry"
10     extends="portlet.ext.campusBrett">
11     <put name="portlet_content"
12         value="/portlet/ext/campusBrett/add_edit_entry.jsp" />
13 </definition >
14
15 <definition name="portlet.ext.campusBrett.edit_entry"
16     extends="portlet.ext.campusBrett">
17     <put name="portlet_content"
18         value="/portlet/ext/campusBrett/edit_entry.jsp" />
19 </definition >
20
21 <definition name="portlet.ext.campusBrett.error"
22     extends="portlet.ext.campusBrett">
23     <put name="portlet_content"
24         value="/portlet/ext/campusBrett/error.jsp" />
25 </definition >
26
27 <definition name="portlet.ext.campusBrett.view_entry"
28     extends="portlet.ext.campusBrett">
29     <put name="portlet_content"
30         value="/portlet/ext/campusBrett/view_entry.jsp" />
31 </definition >
32
33 <definition name="portlet.ext.campusBrett.search"
34     extends="portlet.ext.campusBrett">
35     <put name="portlet_content"
36         value="/portlet/ext/campusBrett/search.jsp" />
37 </definition >
```

Listing A.3: Aufbereitung der Inhaltsvorschläge in der `view_entry.jsp`

```

1 <table style="border:groove thick; border-color:#FFF; width:100%;">
2 <%
3     // ähnliche Anzeigen suchen
4     List entries = CampusBrettLocalServiceUtil.getSimilarEntries(entry);
5 %>
6 <!-- Header -->
7 <tr>
8     <th style="background-color:#CCC" colspan="2">
9     <!-- padding: oben, rechts, unten, links -->
10    <div valign="top" style="padding:2px 0px 2px 2px;">
11        
13        Ähnliche Anzeigen
14    </div>
15    </th>
16 </tr>
17 <!-- für jede gefunden Anzeige eine neue Zeile generieren. -->
18 <%
19     if(Validator.isNotNull(entries)) {
20     for (int i = 0; i < entries.size(); i++) {
21         CEntry similarEntry = (CEntry) entries.get(i);
22         long similarEntryUserId = similarEntry.getUserId();
23
24         User similarEntryAuthor =
25             UserLocalServiceUtil.getUser(similarEntryUserId);
26     %>
27 <tr style="border:#FFF 2pt solid; ">
28     <td>
29     <!-- padding: oben, rechts, unten, links -->
30     <div style="margin:10px 0px 10px 10px">
31         <!-- Foto des Autors -->
32         <c:if test="<%= author != null %>">
33             <img height="70"
34                 alt="<%=similarEntryAuthor.getFullName()%>"
35                 class="avatar"
36                 src="<%=themeDisplay.getPathImage()%>/user_<%=
37                 similarEntryAuthor.isFemale() ? "female" : "male"
38                 %>_portrait?img_id=<%=similarEntryAuthor
39                 .getPortraitId()%>&t=<%=ImageServletTokenUtil
40                 .getToken(similarEntryAuthor.getPortraitId())%>" />
41             </c:if>
42         </div>
43     </td>

```

```
44 <td valign="top" class="similar_entry_out"
45     onmouseover="this.className='similar_entry_over' "
46     onmouseout="this.className='similar_entry_out' "
47     onClick="location.href =
48         '<portlet:renderURL>
49             <portlet:param name="struts_action"
50                 value="/ext/campusBrett/view_entry" />
51             <portlet:param name="redirect"
52                 value="<%= currentURL %>" />
53             <portlet:param name="entryId"
54                 value="<%= String.valueOf(
55                     similarEntry.getEntryId() %>" />
56             </portlet:renderURL>' ;">
57 <div style="margin:10px 10px 10px 5px" >
58     <!-- Titel -->
59     <strong>
60         <%=similarEntry.getTitle() %>
61     </strong><br/>
62     <!-- Content -->
63     <%=similarEntry.getContentText() %>
64 </div>
65 </td>
66 </tr>
67 <% }} %>
68 </table>
```

## B. Abbildungen



Abbildung B.1.: Screenshot der Campus-Brett Hauptseite mit Aufteilung der JSP-Seiten



Abbildung B.2.: Screenshot der Campus-Brett Detailansichtsseite mit Aufteilung der JSP-Seiten

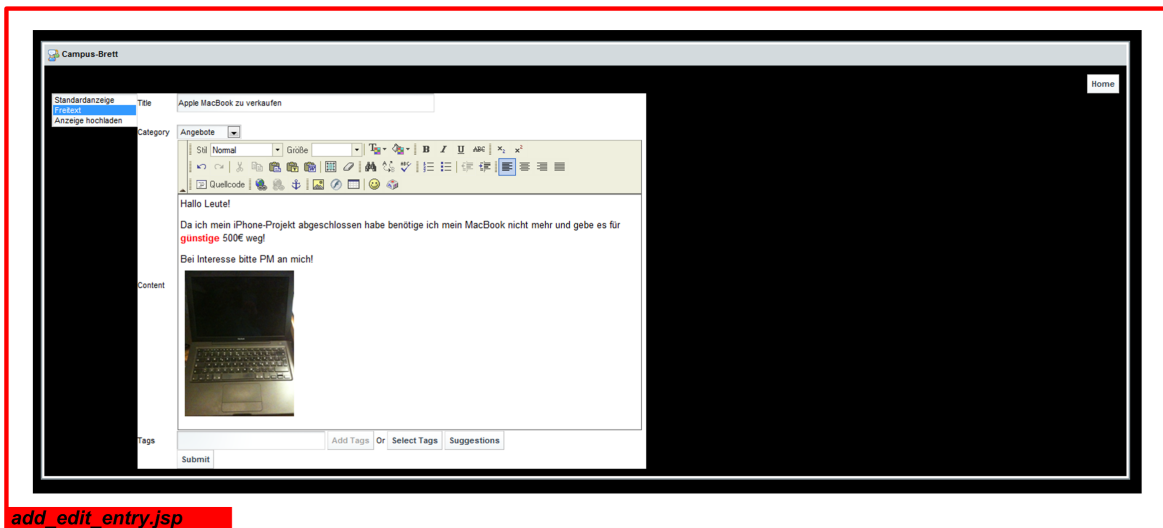


Abbildung B.3.: Screenshot der Campus-Brett Anzeigenerstellungsseite mit Aufteilung der JSP-Seiten



# Literaturverzeichnis

- [Alpar u. a. 2007] ALPAR, Paul ; BLASCHKE, Steffen ; KESSLER, Steffen: *Web 2.0 - Neue erfolgreiche Kommunikationsstrategien für kleine und mittlere Unternehmen*. Hessen Agentur, 2007. – ISBN 978-3-939358-57-2
- [Baacke 1985] BAACKE, Dieter: *Heimliche Kommunikation auf dem Hochschulcampus*. In: *Siegfried Müller: Graffiti: tätowierte Wände*. Bielefeld : AJZ, 1985. – 39–50 S
- [Back u. a. 2008] BACK, A. ; GRONAU, N. ; TOCHTERMANN, K.: *Web 2.0 in der Unternehmenspraxis: Grundlagen, Fallstudien und Trends zum Einsatz von Social Software*. Oldenbourg Verlag, 2008. – ISBN 978-3-486-58579-7
- [Barabási 2002] BARABÁSI, Albert-László: *Linked: The New Science of Networks*. Perseus Publishing, 2002
- [Borcea u. a. 2007] BORCEA, Cristian ; GUPTA, Ankur ; KALRA, Achir ; JONES, Quentin ; IFTODE, Liviu: The MobiSoC middleware for mobile social computing: challenges, design, and early experiences. (2007), S. 1–8. ISBN 978-1-59593-984-5
- [Bowers und Benford 1991] BOWERS, John M. ; BENFORD, Steven D.: *Studies in computer supported cooperative work: theory, practice and design*. Amsterdam, The Netherlands, The Netherlands : North-Holland Publishing Co., 1991. – ISBN 0-444-88811-X
- [Bry u. a. 2010] BRY, Francois ; CAP, Clemens ; DAHM, Ingo ; MAINTZ, Julia ; SCHAFFERT, Sebastian: *Digital Social Media Dagstuhl Manifesto*. (2010), March
- [Buhse und Stamer 2008] BUHSE, Willms ; STAMER, Sören: *Enterprise 2.0 - Die Kunst, loszulassen*. Rhombos Verlag, 2008
- [Carter u. a. 2004] CARTER, Scott ; CHURCHILL, Elizabeth ; DENOUE, Laurent ; HELFMAN, Jonathan ; NELSON, Les: Digital graffiti: public annotation of multimedia content. In: *CHI '04 extended abstracts on Human factors in computing systems*. New York, NY, USA : ACM, 2004 (CHI '04), S. 1207–1210. – URL <http://doi.acm.org/10.1145/985921.986025>. – ISBN 1-58113-703-6

- [Churchill u. a. 2004] CHURCHILL, Elizabeth F. ; NELSON, Les ; DENOUE, Laurent ; HELFMAN, Jonathan ; MURPHY, Paul: Sharing multimedia content with interactive public displays: a case study. In: *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*. New York, NY, USA : ACM, 2004 (DIS '04), S. 7–16. – URL <http://doi.acm.org/10.1145/1013115.1013119>. – ISBN 1-58113-787-7
- [Clement und Albers 2005] CLEMENT, M. ; ALBERS, S.: Netzeffekte und Lebenszyklus von Musik. In: *Ökonomie der Musikindustrie, Dt. Univ. Verl., Wiesbaden* (2005)
- [comundus 2010] COMUNDUS: *Liferay Portal Technik*. Webseite. November 2010. – URL <http://www.comundus.com/produkte/liferay/technik/>
- [Duma und Hecht 2006] DUMA, Ulrike ; HECHT, Heiko: Die Benutzerfreundlichkeit der Homepages deutscher Universitäten: Ein Internet Usability Vergleich. In: *Mainz Experimental Psychology Reports (Universität Mainz)* (2006). – URL [http://www.psych.uni-mainz.de/abteil/aep/download/forschung/Duma\\_Hecht\\_2006.pdf](http://www.psych.uni-mainz.de/abteil/aep/download/forschung/Duma_Hecht_2006.pdf)
- [Eagle und Pentland 2005] EAGLE, Nathan ; PENTLAND, Alex: Social Serendipity: Mobilizing Social Software. In: *IEEE Pervasive Computing* 4 (2005), April, S. 28–34. – URL <http://portal.acm.org/citation.cfm?id=1070601.1070625>. – ISSN 1536-1268
- [Espinoza u. a. 2001] ESPINOZA, Fredrik ; PERSSON, Per ; S, Anna ; NYSTRÖM, Hanna ; CACCIATORE, Elenor ; BYLUND, Markus: *GeoNotes: Social and Navigational Aspects of Location-Based Information Systems*, Springer, 2001, S. 2–17
- [Fischer 2009] FISCHER, Katrin: *Laute Wände an stillen Orten: Klo-Graffiti als Kommunikationsphänomen*. Deutscher Wissenschafts-Verlag, 2009
- [Foundation 2010] FOUNDATION, The Apache S.: *Apache Struts*. 2010. – URL <http://struts.apache.org/>
- [García 2007] GARCÍA, Jürgen S.: *Enterprise 2.0 - Web 2.0 in Unternehmen*. VDM Verlag Dr. Müller, 2007. – ISBN 978-3-8364-0399-3
- [Google 2010] GOOGLE: *Official Google Blog*. Weblog. 2010. – URL <http://googleblog.blogspot.com>
- [Granovetter 1973] GRANOVETTER, Mark: The Strength of Weak Ties. In: *American Journal of Sociology* 78 (1973), S. 1360–1380
- [Griswold u. a. 2004] GRISWOLD, William G. ; SHANAHAN, Patricia ; BROWN, Steven W. ; BOYER, Robert ; RATTO, Matt ; SHAPIRO, R. B. ; TRUONG, Tan M.: ActiveCampus: Experiments in Community-Oriented Ubiquitous Computing. In: *Computer* 37 (2004), October,

- S. 73–81. – URL <http://portal.acm.org/citation.cfm?id=1032275.1032351>. – ISSN 0018-9162
- [Katz 2008] KATZ, Richard N.: *The Tower And The Cloud*. 2008. – URL <http://www.educause.edu/thetowerandthecloud>
- [Köckritz 2006] KÖCKRITZ, Oliver: Geschichte und Konzept von Collaborative Workspaces. (2006). – URL <http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master2006/koeckritz/abstract.pdf>
- [Koch 2010] KOCH, Michael: *Community Mirrors - Using public shared displays to move information out of the box*. Präsentation. Januar 2010
- [Koch und Gross 2006] KOCH, Michael ; GROSS, T.: Computer-Supported Cooperative Work - Concepts and Trends. In: *Proc. Conf. of the Association Information And Management (AIM)* P-92 (2006), S. 165–172. – URL <http://www.kooperationssysteme.de/wordpress/wp-content/uploads/Koch2006-aim.pdf>
- [Koch 2008] KOCH, Prof. Dr. M.: *Enterprise 2.0 - Social Software in Unternehmen*. (2008)
- [Kopp 2006] KOPP, Olaf: *Viral- und Buzzmarketing - Neue Wege in der Musikvermarktung*, Fachhochschule Hildesheim / Holzminen / Göttingen, Diplomarbeit, 2006. – URL <http://www.online-marketing-deutschland.de/social-media-marketing/social-media-marketing-viral-und-buzzmarketing-neue-wege-in-der-musikvermarktung/>
- [Kreßner 2010] KRESSNER, Tino: *we and x - Social Communities*. Webseite. 2010. – URL <http://www.weandx.de/social-communities>
- [Kurbel u. a. 2009] KURBEL, Karl ; BECKER, Jörg ; GRONAU, Norbert ; SINZ, Elmar ; SUHL, Leena: *Enzyklopädie der Wirtschaftsinformatik, CSCW*. 2009. – URL <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/wi-enzyklopaedie/lexikon/daten-wissen/Informationsmanagement/IT-Infrastruktur/Informations--und-Kommunikationstechnologien/computer-supported-cooperative-work-cscw>
- [Liferay 2010] LIFERAY: *Liferay Developer Zone*. Webseite. 2010. – URL <http://www.blokefood.com/liferay-portal-4.2.1/web/guest/devzone>
- [LiferayInc 2010] LIFERAYINC: *Enterprise open source portal and collaboration software - Liferay.com*. 2010. – URL <http://www.liferay.com>
- [McAfee 2006a] MCAFEE, Andrew: *Andrew McAfee's Blog*. Blog. May 2006. – URL [http://andrewmcafee.org/2006/05/enterprise\\_20\\_version\\_20/](http://andrewmcafee.org/2006/05/enterprise_20_version_20/)

- [McAfee 2006b] MCAFEE, Andrew P.: Enterprise 2.0: The Dawn of Emergent Collaboration. In: *MIT Sloan Management Review* 47 (2006), Nr. 3, S. 21–28. – URL <http://sloanreview.mit.edu/the-magazine/articles/2006/spring/47306/enterprise-the-dawn-of-emergent-collaboration/>
- [Microsystems September 2008] MICROSYSTEMS, Sun: The Estuary Effect - Sun Takes Web 2.0 into the Workplace. (September 2008). – URL <http://www.mouncey3.com/Assets/Sun%20Enterprise%202.0%20CS.pdf>
- [Milgram 1967] MILGRAM, Stanley: The Small World Problem. In: *Psychology Today* (1967), S. 60–67
- [Nastansky u. a. 2002] NASTANSKY, L. ; BRUSE, T. ; HABERSTOCK, P. ; HUTH, C. ; SMOLNIK, S.: *Büroinformations- und Kommunikationssysteme: Groupware, Workflow Management, Organisationsmodellierung und Messaging-Systeme*. Erich Schmidt Verlag, 2002. – 235–327 S
- [Ott u. a. 2009] OTT, Florian ; RICHTER, Alexander ; KOCH, Michael: Einsatz großer Wandbildschirme als Fenster in Kooperationssysteme. In: *Mensch & Computer 2009: Grenzenlos frei?* (2009), S. 103–112
- [Ott u. a. 2010a] OTT, Florian ; RICHTER, Alexander ; KOCH, Michael: *CommunityMirrors: Touchscreens als ubiquitäre Natural User Interfaces für Kooperationssysteme*. 2010. – URL <http://www.communitymirrors.net/>
- [Ott u. a. 2010b] OTT, Florian ; RICHTER, Alexander ; KOCH, Michael: Einsatz halböffentlicher Touchscreens als ubiquitäre Benutzerschnittstellen für Social Networking Services. (2010). – URL <http://www.scribd.com/doc/27647978/SocialNetworkingMirror%E2%84%A2-Einsatz-halboffentlicher-Touchscreens-als-ubiquitare-Benutzerschnittstellen-fur-Social-Networking-Services>
- [Reiser 2009a] REISER, Peter: *Community Equity Overview*. November 2009. – URL [http://blogs.sun.com/peterreiser/entry/community\\_equity\\_overview\\_paper\\_published](http://blogs.sun.com/peterreiser/entry/community_equity_overview_paper_published)
- [Reiser 2009b] REISER, Peter: *Toward a social reputation economy*. Präsentation. 2009
- [Reiser und Arconati 2009] REISER, Peter ; ARCONATI, Bill: *Voice of the Customer webinar with Peter Reiser of Sun Microsystems*. Video. Februar 2009. – URL [http://blogs.atlassian.com/news/2009/02/sun\\_microsystem.html](http://blogs.atlassian.com/news/2009/02/sun_microsystem.html)
- [Rogers 1995] ROGERS, Everett M.: *Diffusion of Innovations*. Free Press, 1995

- [Schütt 2006] SCHÜTT, Peter: Social Computing im Web 2.0. (2006). – URL [http://www.wissensmanagement.net/insight/peter\\_schuett/2006\\_06\\_005.pdf](http://www.wissensmanagement.net/insight/peter_schuett/2006_06_005.pdf)
- [Schwarzer 2010] SCHWARZER, Jan: Collaborative Programming with Google Wave. (2010). – URL <http://users.informatik.haw-hamburg.de/~ubicomp/projekte/master09-10-aw1/schwarzer/bericht.pdf>
- [Siegler 2009-05-28] SIEGLER, MG: Google Wave Drips With Ambition. A New Communication Platform For A New Web. (2009-05-28). – URL <http://techcrunch.com/2009/05/28/google-wave-drips-with-ambition-can-it-fulfill-googles-grand-web-vision/>
- [Sledgianowski und Kulviwat 2008] SLEDGIANOWSKI, Deb ; KULVIWAT, Songpol: Social Network Sites: Antecedents of user adoption and usage. In: *Proceedings of 14th Americas Conference on Information Systems* (2008)
- [Snibbe und Raffle 2009] SNIBBE, Scott S. ; RAFFLE, Hayes S.: Social immersive media: pursuing best practices for multi-user interactive camera/projector exhibits. In: *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA : ACM, 2009, S. 1447–1456. – ISBN 978-1-60558-246-7
- [Surowiecki 2004] SUROWIECKI, James: *The Wisdom Of Crowds: Why The Many Are Smarter Than The Few And How Collective Wisdom Shapes Business, Economies, Societies And Nations*. Little, Brown, 2004
- [Unsworth 2008] UNSWORTH, John: University 2.0. In: *The Tower and The Cloud* (2008), S. 227–237. – URL <http://www.educause.edu/thetowerandthecloud>
- [Watts 2004] WATTS, Duncan J.: The New Science of Networks. In: *Annual Review of Sociology* 30 (2004), S. 243–270. – URL <http://www.annualreviews.org/doi/pdf/10.1146/annurev.soc.30.020404.104342>
- [Wellman 2009] WELLMAN, Barry: *The Networked Individual: A Profile of Barry Wellman*. Webseite. Januar 2009. – URL <http://www.semioticon.com/semiotix/semiotix14/sem-14-05.html>
- [Wikipedia 2010] WIKIPEDIA: *Wikipedia*. Webseite. 2010. – URL <http://www.wikipedia.org>
- [Yuan 2009] YUAN, Jonas X.: *Liferay Portal 5.2 Systems Development*. Packt Publishing, 2009. – ISBN 978-1-847194-70-1

Alle Verweise auf Quellen im Internet wurden am 21.12.2010 auf ihre Aktualität überprüft.

# Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §24(5) ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 23. Dezember 2010

Ort, Datum

Unterschrift