

# BEWERTUNG VON NUTZERGENERIERTEN INHALTEN

Zur Erlangung des Grades

**Bachelor of Science**

im Studiengang Software System Entwicklung

an der Fakultät Informatik der Universität Hamburg

vorgelegt von

Simon Kostede

---

Erstbetreuer:

Dr. Guido Gryczan

Zweitbetreuer:

Dr. Axel Schmolitzky

Eingereicht am:

24.09.2012

# EIDESSTATTLICHE ERKLÄRUNG

Ich versichere, dass ich die Arbeit eigenständig verfasst und keine anderen als die angegebenen Hilfsmittel - insbesondere keine im Quellenverzeichnis nicht benannten Internetquellen - benutzt habe, die Arbeit vorher nicht in einem anderen Prüfungsverfahren eingereicht habe und die eingereichte schriftliche Fassung der auf dem elektronischen Speichermedium entspricht. Ich stimme der Veröffentlichung in nicht digitaler Form in der Bibliothek des Fachbereichs Informatik zu.

**Datum:** 24.09.2012

Simon Kostede

# INHALTSVERZEICHNIS

---

<b>INHALTSVERZEICHNIS</b> .....	<b>3</b>
<b>DANKSAGUNG</b> .....	<b>4</b>
<b>1 EINLEITUNG &amp; PROBLEMSTELLUNG</b> .....	<b>5</b>
1.1 DELEGS-Projekt.....	5
<b>2 GRUNDLAGEN &amp; BEWERTUNGSSYSTEME</b> .....	<b>7</b>
2.1 Definition Bewertungssysteme .....	7
2.2 Typen von Bewertungssystemen.....	7
2.3 Modelle der Bewertungssysteme .....	11
2.4 Bewertung Ermitteln .....	13
2.5 Engagement der Nutzer .....	16
2.6 Signale .....	20
2.7 Probleme bei Bewertungssystemen.....	21
2.8 Beispiel: Reddit.....	23
<b>3 KRITERIENKATALOG FÜR BEWERTUNGSSYSTEME</b> .....	<b>26</b>
3.1 Aufwand & Infrastruktur .....	26
3.2 Grundlagen des Bewertungssystems.....	28
3.3 Modelle.....	31
3.4 Risiken.....	33
3.5 Ziele & Inhalte .....	35
3.6 Nutzerbindung.....	37
3.7 Gamification .....	38
3.8 Indirekte Bewertungssysteme.....	39
3.9 Auswertung.....	41
<b>4 BEISPIELANWENDUNG</b> .....	<b>42</b>
4.1 Motivation & Umsetzung .....	42
4.2 Architektur .....	43
4.3 Client-Server Kommunikation.....	46
4.4 Benutzerkonten & Authentifikation.....	49
4.5 Persistenz .....	50
4.6 Eingesetzte Technologien .....	51
<b>5 FAZIT &amp; AUSBLICK</b> .....	<b>53</b>
5.1 Zusammenfassung.....	53
5.2 Ausblick.....	53
5.3 Fazit.....	53
<b>LITERATURVERZEICHNIS</b> .....	<b>54</b>
<b>ABBILDUNGSVERZEICHNIS</b> .....	<b>56</b>
<b>TABELLENVERZEICHNIS</b> .....	<b>57</b>
<b>ANHANG</b> .....	<b>58</b>

# DANKSAGUNG

---

*Ich bedanke mich bei Dr. Guido Gryczan, dass er sich bereit erklärt hat, der Erstbetreuer dieser Arbeit zu sein. Ebenso danke ich Dr. Axel Schmolitzky für die Zweitbetreuung.*

*Weiterhin bedanke ich mich bei den Mitarbeitern des Delegs-Projekts für die wertvolle Kritik und Anmerkungen.*

*Bei Stefan Wöhrmann bedanke ich mich, dass er sich die Zeit genommen hat, die Beispielanwendung auszuprobieren und mir Fragen zu beantworten.*

*Insbesondere bedanken möchte ich mich bei meinen Eltern. Ganz besonders für die Unterstützung beim Erstellen dieser Arbeit.*

# 1 EINLEITUNG & PROBLEMSTELLUNG

---

Die vorliegende Arbeit beschäftigt sich mit Bewertungssystemen für nutzergenerierte Inhalte. Diese Thematik hat durch die Entwicklung des „Social Webs“ und des Web 2.0 eine hohe Aktualität. Viele der neuen aufstrebenden Internetfirmen operieren im Bereich der sozialen Netzwerke und der nutzergenerierten Inhalte. Durch die Entwicklungen im mobilen Bereich entstanden neue Webseiten, bereits existierenden Webseiten wurden größer und der Anteil der durch Nutzer erstellten Inhalte nahm zu.

Die Inhalte in Form von Links, Kommentaren oder Fotos, sollen von anderen Nutzern gesehen werden. Die Nutzer wollen mit den Inhalten interagieren. Allerdings wollen die meisten Nutzer nur mit denen für sie relevanten Inhalten interagieren. So bekommen Nutzer von Facebook beispielsweise viele Interaktionen ihrer Freunde nicht angezeigt, da Facebook diese Inhalte für nicht relevant für sie hält. Um diese Relevanz festzustellen muss Facebook die Inhalte anhand der Interessen der Nutzer bewerten.

Die Arbeit soll mögliche Bewertungssysteme aufzeigen und eine Lösungshilfe bei der Auswahl des richtigen Bewertungssystems bieten.

## 1.1 DELEGS-PROJEKT

Das Thema entwickelte sich aus den Problemen des DELEGS-Projekts und der SignWriting-Community. Das DELEGS-Projekt versucht den Unterricht der deutschen Sprache für Gehörlose mithilfe eines Web-Editors für GebärdenSchrift zu unterstützen. Der Anwendungsfall für ein Bewertungssystem ist dabei die Bewertung der vielen nutzergenerierten GebärdenSchrift-Symbole. Diese können von jedem Nutzer erstellt werden.

Im Detail importiert das DELEGS-Projekt für seinen Web-Editor das Wörterbuch aus dem Signpuddle Service. Das Wörterbuch enthält, für jede Sprache einzeln, alle GebärdenSchrift-Symbole, die von Nutzern für diese Sprache erstellt wurden. Beispielsweise enthält das Wörterbuch im Falle der deutschen GebärdenSprache über 15.000 GebärdenSchrift-Symbole.

### 1.1.1 GEBÄRDENSCHRIFT UND GEBÄRDENSPRACHE

Die Gebärdensprache ist keine direkte Übertragung der deutschen Sprache, oder irgendeiner Sprache in die Gestalt von Gebärden. Stattdessen ist die Gebärdensprache eine eigene Sprache, die sich allerdings regional stark unterscheidet. Unter anderem durch die regionalen unterschiedlichen Ausprägungen gibt es viele Varianten einzelner Gebärden.

Die vielen unterschiedlichen Gebärden werden durch die Nutzer in die Gebärdenschrift überführt. Die Gebärdenschrift ist eine zweidimensionale Abbildung der Gebärden. Diese Abbildungsvariante wurde aus einer Notation für Tanzbewegungen entwickelt. Sie drückt Gebärden durch intuitiv verständliche Symbole aus. So werden beispielsweise Mundbewegungen durch Mundformen auf simplifizierten Köpfen oder Handbewegungen durch Hände und Pfeile dargestellt.

### 1.1.2 DELEGS-EDITOR

Der DELEGS-Editor ist ein Web-2.0-Werkzeug zum Erstellen von Dokumenten, insbesondere Lehrmaterialien, in Gebärdenschrift. Der Editor funktioniert so, dass der Nutzer ein Wort, einen Buchstaben oder eine Phrase in einer Lautsprache, beispielsweise Deutsch, eingibt und daraufhin das entsprechende Gebärdenschrift-Symbol dargestellt wird. Da es für eine Eingabe in Lautsprache oft mehrere alternative Ausdrucksweisen gibt, werden diese auch angeboten. Dieses indiziert einen Bedarf für ein Bewertungssystem, da beispielweise für das Wort „Hallo“ 17 verschiedene Varianten angeboten werden. Diese Varianten werden zurzeit in der Reihenfolge angezeigt wie sie in die Datenbank von Signpuddle eingegeben wurden. Das ist meist keine sinnvolle Reihenfolge. Aber da es keine einfache Methode gibt mit der die Gebärdenschrift-Symbole bewertet oder geordnet werden könnte, besteht erst mal keine Möglichkeit eine sinnvolle Ordnung herzustellen.

Hier soll der praktische Teil der Arbeit anknüpfen in dem die möglichen Bewertungsmethoden aufgezeigt werden und ausgearbeitet wird, wie ein Bewertungssystem für das Projekt aussehen könnte.

## 2 GRUNDLAGEN & BEWERTUNGSSYSTEME

---

Das folgende Kapitel beschäftigt sich mit den Grundlagen von Bewertungssystemen. Dazu zählen neben der Definition von Bewertungssystemen (2.1) auch die verschiedenen Typen von Bewertungssystemen (2.2).

Ein weiterer Abschnitt beschäftigt sich mit der Motivation und dem Engagement (2.5) der Nutzer von Bewertungsplattformen und dem Design von Communities um diese Bewertungssysteme herum (2.5.1). Im Kontext der Motivation der Nutzer wird auch der Ansatz Gamification und seine Nähe zu nutzerbasierten Bewertungssystemen dargestellt (2.5.3). Abschließend geht es dann um Probleme von Bewertungssystemen (2.7.2).

### 2.1 DEFINITION BEWERTUNGSSYSTEME

Ein Bewertungssystem ist ein System das über Inhalte eine direkt oder indirekt wertende Aussage trifft. Dafür nutzt das System eine definierte Menge an Signalen, um den Wert der einzelnen Inhalte zu bilden. Dieser Wert ist nur innerhalb eines Kontextes gültig, wenn keine speziellen Vorkehrungen getroffen wurden. (vgl. [FAGL10, p.6])

### 2.2 TYPEN VON BEWERTUNGSSYSTEMEN

Die Bewertungssysteme können grob in zwei Gruppen aufgeteilt werden: direkte und indirekte Bewertungssysteme. Bei direkten Systemen werden unmittelbar die Inhalte bewertet oder ausgewertet. (vgl. [FAGL10, p.4])

Bei indirekten Systemen werden die Inhalte über einen Proxy bewertet. Dabei wird eine Bewertung für etwas erzeugt, von dem angenommen wird, dass es in starker Relation zu der Qualität der Inhalte steht. Ein Beispiel für einen Proxy ist ein Nutzer, der einen Inhalt erstellt hat. (vgl. [FAGL10, p.4])

Die Aufteilung der Typen ist nicht absolut. Einzelne Bewertungssysteme können Methoden aus diesen Typen kombinieren, um ein Gesamtsystem zu bilden. Eine besonders häufige Ausprägung ist dabei die Kombination von computergestützter und Nutzerbasierter Bewertung. (siehe auch 2.2.3)

## 2.2.1 ANALYSEBASIERTE BEWERTUNGEN

Eines der grundlegenden Bewertungssysteme ist die Analyse von Daten. Diese Systeme erfassen bestimmte Signale in der Datenbasis und bilden aus den Signalen dann eine Bewertung über den Inhalt. Je nach Art der zu bewertenden Inhalte kann man verschiedene sinnvolle Signale auswählen. Beispiele für Signale im Bereich Webseiten sind die Rate der Aktualisierung, die Anzahl der Besucher oder die Anzahl an qualitativ hochwertigen Webseiten, die auf eine Webseite verlinken. (vgl. [LIFS08])

Diese Signale sind extern zum Bewertungssystem und es gibt keine expliziten Bewertungsaktionen, die der Nutzer im Bewertungssystem durchführen kann. Bewertungssysteme solche sind stark von ihren Auswertungssystemen abhängig. Da solche Auswertungssysteme die Bewertung nur aus einer bestehenden Datenlage erzeugen, ist ein Nachteil die oft sehr hohe Komplexität der Systeme. Ein Vorteil jedoch ist, dass Bewertungssysteme dieser Art auf sehr große Mengen von Inhalten skalieren können. (vgl. [FAGL10, p.23])

Je nach Quelle und Umfang der Inhalte bieten sich eher Techniken aus dem Data-Mining oder Techniken aus dem Bereich des maschinellen Lernens an. Data-Mining Techniken sind bei strukturierteren Daten anwendbar. Bei unstrukturierten oder unbekanntem Daten sind Techniken des maschinellen Lernens anwendbar. Diese beiden Techniken nutzen statistisch und mathematisch begründete Methoden um aus Daten Wissen und Muster zu extrahieren. Im Fall eines Bewertungssystems ist dieses Wissen das Wissen um die Qualität des Inhaltes. (vgl. [LIFS08])

Ein Beispiel für den Einsatz solcher Systeme sind Suchmaschinen. Sie bewerten nutzergenerierte Webseiten aufgrund von verschiedenen Signalen. Zusätzlich zu den Signalen versuchen Suchmaschinen Muster zu ermitteln, um von diesen aus die Autorität der verlinkten Seiten zu bewerten. (vgl. [LIFS08])

## 2.2.2 NUTZERBASIERTE BEWERTUNG

Hierbei werden die Signale des Bewertungssystems durch indirekte oder explizite Nutzeraktionen erzeugt. Das heißt, dass die Nutzer direkt eine Bewertung eines Inhaltes vornehmen, sei es, dass sie den Inhalt als Favorit markieren oder indem sie dem Inhalt eine hohe Bewertung zuteilen. Auch können die Nutzer durch indirekte



Aktionen Daten für das Bewertungssystem erzeugen. Ein Beispiel für einen solchen Indikator ist, wie lange die Nutzer einen Inhalt betrachteten. (vgl. [FAGL10, p.6])

Bei der nutzerbasierten Bewertung gibt es viele verschiedene konkrete Implementationen. Einfache Systeme erlauben es dem Nutzer, nur eine positive Aussage über den Inhalt zu treffen. Andere Systeme erlauben auch negative Bewertungen für Inhalte, wobei Farmer und Glass (vgl. [FAGL10]) zeigen, dass die negative Bewertung seltener genutzt wird. Wie an Abbildung 2 zu sehen ist, wird diese geringere Nutzung auch in Benutzerinterfaces berücksichtigt. (siehe auch 2.3.1)

### 2.2.3 KOMBINATION VON COMPUTERGESTÜTZTER UND NUTZERBASIERTER BEWERTUNG

Bei diesem Bewertungstyp wird die computergestützte Auswertung mit der nutzerbasierten kombiniert. Beide Bewertungssysteme laufen parallel und dann werden in einer Kombinationsphase die Ergebnisse zu einem Gesamtergebnis zusammengefügt. (vgl. [FAGL10, p.78])

In einer Variante wird durch einen der beiden Prozesse der andere mit Initialwerten bedient. Dies hat den Vorteil, dass entweder die computergestützten Bewertungsprozesse damit optimiert werden können, was die Nutzer für wichtig erachten. Alternativ kann auch eine Vorauswahl an Bewertungsgegenständen getroffen werden, um die Bewertungen der Nutzer fokussierter durchführen zu können. (vgl. [FAGL10, p.83])

Ein Beispiel für diese fokussierte Bewertung durch den Nutzer sind Spamfilter. Sie bewerten Teile des Inhaltes vollautomatisch als für die Nutzer uninteressant und erlauben es so den Nutzern, sich auf relevante Inhalte zu konzentrieren. (vgl. [FAGL10, p.134])

### 2.2.4 EXPERTENBEWERTUNG

Bei der Expertenbewertung werden die Inhalte von Fachleuten bewertet. Die Experten wurden für die Bewertungstätigkeit eingestellt oder mit ihr beauftragt. Der Vorteil dieser Bewertungsmethode ist, dass sehr gute Ergebnisse zu erwarten sind.

Da Experten ein großes Fachwissen im Bereich ihrer Expertise haben, können sie Inhalte mit hoher Genauigkeit bewerten. Auch können sie Inhalte mit hoher inhaltlicher Komplexität bewerten. (vgl. [FAGL10, p.248])

Nachteil der Expertenbewertung sind die hohen Kosten für die Experten und die Einschränkungen in der Bewertungsrate. Da nur eine limitierte Anzahl von Experten vorhanden ist, können diese auch nur eingeschränkte Inhaltsmengen verarbeiten. Dies wird des Weiteren durch die Komplexität des Inhalts eingeschränkt. (vgl. [FAGL10, p.273])

Die Expertenbewertung ist von Vorteil, wenn die hohe Qualität der Bewertung benötigt wird und wenn die Inhaltsmenge nicht übermäßig groß ist. Dazu zählt auch, dass die Expertenbewertung für sehr komplexe Inhalte eingesetzt werden kann.

Ein Beispiel für diese Art der Bewertung sind die frühen Suchmaschinen bzw. Linksammlungen. Dabei wird jede Webseite per Expertenbewertung in den Index eingeordnet. Das Wachstum des Internets machte dieses Vorgehen irgendwann unmöglich. (vgl. [FAGL10, p.108])

## 2.2.5 INDIREKTE BEWERTUNGSSYSTEME

Bei indirekten Systemen wird die Qualität von Inhalten daran gemessen wie gut die Bewertung des Proxys ist. Eine häufige Ausprägung ist das Ermitteln einer Bewertung oder Reputation des Nutzers oder von Nutzergruppen. Die indirekten Bewertungssysteme sind hier beispielhaft an der Reputation für Nutzer aufgezeigt.

Die Reputation in diesen Systemen ist die Summe aller „guten“ und „schlechten“ Interaktionen eines Nutzers mit einer Community. Was „gut“ oder „schlecht“ ist, entsteht aus der Community heraus. Dabei ist das Reputationssystem dafür zuständig, die Aktionen eines Nutzers und die Reaktionen der restlichen Community auf diese Aktionen aufzubereiten. Hierbei nutzt das System die Reputation intern, um Rückschlüsse auf die Inhalte der Nutzer zu ziehen. (vgl. [GLAS08])

Oft wird die Reputation eines Nutzers diesem oder der gesamten Community angezeigt. (siehe auch 2.3) Wobei es nicht notwendigerweise darum geht, ein Belohnungssystem zu haben. (siehe auch 2.5.3) Es gibt in der Praxis viele Beispiele

für solche Systeme, unter anderem das System von eBay, in dem die Käufer die Verkäufer und umgekehrt bewerten können, wobei auch die Anzahl der Transaktionen in die Reputation einfließt. Ein weiteres Beispiel sind die Kundenrezensionen bei Amazon, wo die Anzahl der Reviews und deren Bewertung durch die anderen Nutzer die Reputation des Nutzers bildet. (vgl. [FAGL10, p.25])

Bei genauerer Betrachtung der Beispiele eBay und Amazon zeigt sich, dass es auf beiden Seiten auch noch weitere Indikatoren für die anderen Kunden gibt, ob sie es gerade mit einer besonders angesehenen Person zu tun haben. Um den Status eines Nutzers in der Community zu verdeutlichen, haben diese Seiten bestimmte Awards oder Level eingeführt. (vgl. [FAGL10, p.193])

Diese zusätzlichen Nutzereigenschaften können verschiedenartig sein und lassen sich auch kombinieren. Wie erwähnt, ist eine dieser Eigenschaften ein Level für den Nutzer. Dieser Level wird entweder als Zahl oder als Titel z.B. „Neuling“, „Enthusiast“, „Profi“ umgesetzt. Wobei es numerische Level einfacher machen, Nutzer miteinander zu vergleichen. Titel involvieren den Nutzer stärker und können im Kontext der Community spezielle Bedeutung haben. (vgl. [FAGL10, p.188])

Ein zu dem Level mittels Bezeichnung ähnliches Pattern ist die Verwendung von Bezeichnungen oder Typenschildern, welche bestimmte Eigenschaften eines Nutzers hervorheben. So könnten Nutzer beispielsweise in „Hilfreiche“, „Offizielle“ oder „Elite“ eingeordnet werden, je nachdem, was im Kontext sinnvoll ist. (vgl. [FAGL10, p.190])

## 2.3 MODELLE DER BEWERTUNGSSYSTEME

Das Modell eines Bewertungssystems beschreibt wie dieses System die Signale für den Bewertungsprozess erhält. Dabei handelt es sich einerseits um Benutzerschnittstellen und andererseits um das Sammeln von indirekten Indikatoren der Qualität.

### 2.3.1 EINFACHE MODELLE

Einfache Modelle sind Grundbausteine für ein Bewertungssystem. Sie sind entweder allein oder in Kombination einzusetzen. (vgl. [FAGL10, p.67]). Dabei können die

Modelle in *Favoriten*, *Flags*, *Abstimmen*, *This-or-That Bewertungen*, *Ratings*, *Rezensionen*, *Punkte* oder *Karma* aufgeteilt werden.

Zu diesen einfachen Modellen zählen Farmer und Glass (vgl. [FAGL10, p.68]) *Favoriten* und *Flags*, ein Modell in dem der Nutzer den Inhalt markiert und entweder damit direkt eine wertenden Aussage macht oder sich diese Wertung indirekt ergibt. Die Markierungen können nach Farmer und Glass (vgl. [FAGL10, p.69]) auch dafür genutzt werden, um unerwünschte Inhalte auszusortieren.

Eine Variante dieses Modells ist das *Abstimmen* um so einen Inhalt hervorzuheben, etwa indem dieser Inhalt dann prominenter für alle Nutzer angezeigt wird.

Als weiteres einfaches Modell nennen Farmer und Glass *This-or-That Bewertungen* (vgl. [FAGL10, p.69]). Bei diesem Modell soll der Nutzer eine enger gefasste Aussage treffen als nur eine positive oder negative Bewertung. Ein Beispiel für so eine Aussage ist die Antwort auf die Frage, ob etwas hilfreich war.

Ein weiteres Modell sind *Ratings*. Die können entweder Gut-Schlecht-Bewertungen sein oder nuancierte Bewertungen wie Abbildung 3 an dem Sternerating zeigt. Bei diesen nuancierten Bewertungen kann der Nutzer deutlicher darstellen, wie er einen Inhalt einschätzt. (vgl. [FAGL10, p.70])

Außerdem gibt es noch die *Rezensionen*, welche es den Nutzern erlauben, eine Vielzahl von verschiedenen Reaktionen auf einen Inhalt zu haben. Dabei verteilt der Nutzer mehrere Ratings auf ein Objekt, wie an Abbildung 4 zu sehen. Zusätzlich zu den Ratings kann der Nutzer auch freien Text als Teil der Rezension ausfüllen. (vgl. [FAGL10, p.70])

Ein weiteres Modell ist das Vergeben von *Punkten* für die Aktionen von Nutzern an die Nutzer (siehe auch 2.2.5 & 2.5.3). Nach Farmer und Glass (vgl. [FAGL10, p.71]) ist dies ein kniffliges Modell, da es Risiken birgt. Es besteht die Gefahr, dass es eine erhöhte Transparenz der Regeln gibt, da die Nutzer wissen müssen, für was die Punkte vergeben werden. Diese Transparenz erhöht die Chance eines erfolgreichen Angriffs auf das Bewertungssystem (siehe auch 2.7.2). Ein Teil der Nutzer wird auch durch das Punktesystem übermäßig in ihrem Verhalten beeinflusst. Sie machen bestimmte Handlungen übermäßig von den erreichbaren Punkten abhängig. Diese

Punkte sind nicht als Währung gedacht. In vielen Systemen werden Punkte genutzt um Dinge zu erwerben. Das sollten aber nicht dieselben Punkte sein, welche die Qualität eines Nutzers repräsentieren. Sollten die gleichen Punkte für beide Funktionen genutzt werden, so muss der Nutzer seine Qualität aufgeben um Dinge zu erhalten. Gleichzeitig ist für das Bewertungssystem nicht mehr erkennbar wie welcher Nutzer bewertet war. (vgl. [GLAS08])

Daneben gibt es noch das Modell *Karma*. Dieses Modell wird oft in Kombination mit andern Bewertungsmodellen genutzt. Es gibt zwei Ausprägungen des Karma-Modells, die eine trifft Aussagen über die Aktivität der Nutzer. Die zweite misst die Qualität der Inhalte. Wenn diese beiden Ausprägungen kombiniert werden, so kann das Modell als Robust angesehen werden (vgl. [FAGL10, p.72]). Bei der Kombination dieser beiden Karma-Modelle werden diejenigen Nutzer, wie an Abbildung 5 zu sehen, am besten bewertet, die sich häufig beteiligen und gute Inhalte erzeugen. Es ist vorteilhaft, das Ergebnis des robusten Modells nicht dem Nutzer anzuzeigen, sondern es nur intern zu verwenden, um die Schwierigkeit eines Angriffs zu erhöhen.

### 2.3.2 KOMBINIERTER MODELLE

Die kombinierten Modelle fügen mehrere der einfachen Bewertungsmodelle zusammen, um damit mehr Facetten der Inhalte zu erfassen. Eine häufige Variante davon ist die Bewertung von Bewertungen, um gute Bewertungen besser zu gewichten. Dabei ist insbesondere das Modell Karma gut zum Kombinieren geeignet (vgl. [FAGL10, p.72]). Ein Beispiel hierfür ist das Bewerten von Rezensionen auf sozialen Webseiten, um die guten *Rezensionen* anzuzeigen.

## 2.4 BEWERTUNG ERMITTELN

In einem Bewertungssystem müssen die Signale verarbeitet werden, um ein Ergebnis des Bewertungsprozesses ausgeben zu können. Dazu müssen die Signale je nach Bewertungsmodell unterschiedlich verarbeitet werden. (vgl. [FAGL10, p.46])

### 2.4.1 ROLL-UPS

Roll-Ups sind ein grundlegendes Element der Auswertung in einem Bewertungssystem. Sie verarbeiten die Eingabewerte zu Zwischenwerten oder

Ausgabewerten. Um eine Auswertung zu erstellen werden mehrere Roll-Ups kombiniert, unter Umständen auch mit zusätzlichen externen Verarbeitungsprozessen. Sie lassen sich in *Zähler*, *Akkumulator*, *Mittelwerte*, *Mixer* und *Ratios* einteilen. (vgl. [FAGL10, p.46])

#### 2.4.1.1 ZÄHLER

Dabei kann ein einfacher Zähler genutzt werden, um einen numerischen Wert über die Anzahl der Eingaben in dieses Signal aufzuaddieren. Der Vorteil dieses Zählers ist die einfache Implementation, welche auch für hohe Lasten geeignet ist. Nachteil dabei ist, dass der Zähler im Fall von Missbrauch (siehe auch 2.7.2) nicht rückgängig gemacht werden kann, so Farmer und Glass (vgl. [FAGL10, p.47]). Zudem sinkt die Relevanz der Bewertung mit abnehmender Aktualität. Das kann ein einfacher Zähler nicht abbilden.

Bei reversiblen Zählern sind diese Probleme gelöst, indem sie speichern, welcher Nutzer wann die Eingabe getätigt hat. Dies hat den Vorteil, dass den einzelnen Nutzern angezeigt werden kann, was sie bewertet haben. Nachteil ist, dass diese Zähler hohen Aufwand im Persistenz System erzeugen, was zu Problemen bei hoher Last führt. (vgl. [FAGL10, p.48])

Allgemein sind diese Zähler insbesondere für Systeme geeignet, die die Modelle *Favoriten*, *Flags* und *Abstimmen* einsetzen.

#### 2.4.1.2 AKKUMULATOREN

Akkumulatoren sind den Zählern sehr ähnlich, nur dass sie statt eines einfachen hochzählenden Wertes jeweils einen spezifischen Wert auf den Stand des Akkumulators addieren. Sie lassen sich wie die Zähler in einfache und reversible Akkumulatoren aufteilen, wobei die reversiblen Akkumulatoren wie bei den reversiblen Zählern zu jedem Nutzer die Aktion speichern. (vgl. [FAGL10, p.49])

#### 2.4.1.3 MITTELWERTE

Ein Mittelwert kann genutzt werden, um dem Nutzer eine verständliche Statistik anzuzeigen. Bei einfachen Mittelwerten wird der Wert der gesammelten Bewertungen durch die Anzahl der Bewertungen geteilt. Bei reversiblen Mittelwerten

wird stattdessen bei jeder neuen Bewertung oder bei jedem Löschen einer Bewertung der Mittelwert neu berechnet. (vgl. [FAGL10, p.50f])

#### 2.4.1.4 MIXER

Ein Mixer kombiniert mehrere Signale mithilfe eines Gewichtungsschlüssels in einen einzigen Wert. Er eignet sich gut, um das Bewertungssystem zu optimieren, indem der Mixer bereits existierende Signale und Bewertungen im System neu kombiniert und gewichtet. (vgl. [FAGL10, p.51])

#### 2.4.1.5 RATIOS

Die einfachen und reversiblen Ratios zählen die Anzahl der Eingaben unabhängig von dem Wert der Eingabe. Mit ihnen kann dann beispielsweise angezeigt werden, wie viele Nutzer aus einer Gesamtanzahl an Nutzern einen Inhalt positiv bewertet haben. Die reversiblen Ratios erlauben es wiederum, einzelne Einträge aus dem Ratio zu entfernen und jedem Nutzer seinen Eintrag zuzuteilen. (vgl. [FAGL10, p.53])

#### 2.4.2 DATEN NORMALISIEREN

Um eine große Anzahl an Eingabewerten sinnvoll auf eine limitierte Anzahl an Ausgabewerten abzubilden, wird der Prozess der Datennormalisierung genutzt. Dabei bildet der Prozess die Eingabewerte nach einer Gewichtung auf einen Wert zwischen 0 und 1 ab. Alternativ bildet der Prozess die Eingabewerte auf verschiedene reguläre Werte auf einer Skala ab, beispielsweise ob etwas Spam ist oder nicht. (vgl. [FAGL10, p.54])

#### 2.4.3 NEURONALE NETZE & DATA-MINING

Die Ermittlung der Bewertung kann auch durch Systeme aus der künstlichen Intelligenz oder der Mustererkennung erfolgen. Diese Systeme sind vor allem dafür geeignet, Inhalte gewissen Flags zuzuordnen. Ein Beispiel ist das Finden von Spam in den Inhalten. Dabei kann das System mit den Daten von altem Spam trainiert werden. Auch kann es Eingaben aus den Meldungen der Nutzer in seinen Bewertungsprozess mit einbeziehen. (vgl. [FAGL10, p.246])

## 2.5 ENGAGEMENT DER NUTZER

Ein Problem bei der Verwendung von Bewertungssystemen, die Nutzer die Inhalte bewerten lassen, ist, dass auch genügend Nutzer die Inhalte bewerten. Dabei gibt es zwei Probleme. Erstens sind die wenigsten Nutzer bereit, Zeit zu investieren, wenn ihnen kein klarer Nutzen aus dem Aufwenden dieser Zeit entsteht. Zweitens sollte auch bei einem klaren Nutzen die Anwendung des Bewertungssystems kurzweilig sein. Vorteilhaft ist es auch, wenn das System so designt ist, dass es Nutzer, die sich viel und gut beteiligen, belohnt. Diese Belohnungen sollten keine monetären Gegenwerte haben, sondern sollten einen rein ideellen Wert haben. (vgl. [GLAS08])

Eine gute Möglichkeit bietet die Erweiterung der erwähnten Reputationslevels (siehe auch 2.2.5 & 2.3.1). So können für Interaktionen Punkte an die Nutzer verteilt werden, welche den Nutzern angezeigt werden. Dieses Vorgehen lässt sich gut mit den Levels verbinden, indem beispielsweise eine gewisse Anzahl an Punkten den Aufstieg ins nächste Level ermöglicht. Diese Punkte können auch genutzt werden, um den Nutzer auf einer Bestenliste einzuordnen. Auf ihr werden die Nutzer positiv hervorgehoben, die besonders aktiv waren oder besonders wertvolle Bewertungen abgegeben haben. (vgl. [GLAS08])

### 2.5.1 GEMEINSCHAFTSMODELLE & GEMEINSCHAFTSGEIST

Bei diesen Vorgehensweisen sollte bedacht werden, welcher Gemeinschaftsgeist in der Community herrschen soll, insbesondere wie die Nutzer in Konkurrenz zueinander treten (vgl. [GLAS08]). Dabei gibt es verschiedene Grade der Konkurrenz:

**Kümmern:** Die Nutzer werden durch das Kümmern um andere Nutzer motiviert. (vgl. [GLAS08])

**Zusammenarbeit:** Die Nutzer teilen ein größtenteils gemeinsames Ziel und arbeiten zusammen, um dieses zu erreichen. (vgl. [GLAS08])

**Freundschaftlich:** Die Nutzer haben unterschiedliche Ziele, diese sind aber nicht gegensätzlich. (vgl. [GLAS08])



**Konkurrierend:** Die Nutzer haben das gleiche Ziel, müssen aber gegeneinander antreten, um es zu erreichen. (vgl. [GLAS08])

**Kämpferisch:** Die Nutzer haben gegensätzliche Ziele und nur einer bzw. wenige können sie erreichen. (vgl. [GLAS08])

Das der gewählte Gemeinschaftsgeist zum Kontext des Bewertungssystems passt, ist sehr wichtig, da ein Konflikt zwischen dem Kontext und dem Gemeinschaftsgeist bei den Nutzern zu Frustration führt. Ein Beispiel ist ein System zum Ermitteln von niedlichen Katzenbildern, das ein Gemeinschaftsmodell im „Kämpferischen“ Spektrum einsetzt. Hierbei würde das Gemeinschaftsmodell dafür sorgen, dass die Nutzer alles versuchen, um anderen Nutzer der Plattform zu sabotieren, ungeachtet von welcher Qualität deren Inhalte sind.

## 2.5.2 MOTIVATION DER NUTZER

Aus den in Abschnitt 2.5.1 vorgestellten Gemeinschaftsmodellen ergibt sich auch die Frage, was die Nutzer der Plattform motiviert. Wie schon in 2.5 erwähnt, werden kaum Nutzer nur aus Altruismus heraus oder für ein virtuelles Belohnungssystem Inhalte bewerten. Das Mitmachen bei dem Bewertungssystem sollte entweder eine schon bestehende Nachfrage befriedigen oder einen direkten Mehrwert für den Nutzer bieten. (vgl. [GLAS08])

Es gibt auch Nutzer, die sich allein durch das Erlangen von Reputation motivieren lassen. Diese sind aber häufig nicht oder nur gering an dem eigentliche Ziel des Bewertungssystem interessiert.

Im Bereich der Kundenrezensionen für Geschäfte stellten sich bei der Befragung durch Glass der Nutzer die Motivationen in Tabelle 1 dar (vgl. [GLAS08]).

## 2.5.3 GAMIFICATION

*„Gamification ist die Verwendung von spieltypischen Mechaniken außerhalb reiner Spiele mit dem Ziel, das Verhalten von Menschen zu beeinflussen.“ [BREU11A]*

Die Verhaltensänderung, welche im Kontext Motivation der Nutzer erreicht werden soll, ist, dass die Nutzer sich länger und intensiver mit den Inhalten und dem

Bewerten dieser auseinandersetzen. Dazu werden wie bei Breuer erwähnt spieltypische Mechaniken genutzt (vgl. [BREU11A]).

Beispiele für diese Mechaniken sind unter anderem Punkte, die als Belohnung vergeben werden, wenn der Nutzer bestimmte Aktionen vollführt. Das Punktesystem ist ähnlich zu den in 2.2.5 und 2.5 vorgestellten Systemen, aber nicht gleich. Beim Punktesystem im Kontext Gamification geht es direkt um die Belohnung der Nutzer für Aktionen, nicht um Reputation der Nutzer oder die Qualität der Inhalte. Weiterhin kann man auch Ranglisten einsetzen, um über die Konkurrenz die Nutzer zum Benutzen des Bewertungssystems zu bringen. Eine weitere Belohnungsmechanik ist das Vergeben von Auszeichnungen an die Nutzer, wenn sie bestimmte Ziele erreicht haben. Auszeichnungen können dabei verschiedene Formen annehmen, sie können rein virtuell sein, entweder von nur ideellem Wert, oder sie nehmen die Form von echten Preisen ein. (vgl. [GLAS08])

#### 2.5.3.1 SPIELERMOTIVATION

Die grundlegende Motivation der Nutzer diese Spielmechanik zu nutzen, folgt aus dem Spaß, den der Nutzer hat. Das bedeutet, wenn die Spielmechaniken keinen Spaß bringen, ist der Einsatz von Gamification gescheitert.

Was den Spielern Spaß macht, hängt nach Breuer stark von deren Charakter ab (vgl. [BREU11B]). Die Spieler teilt Breuer in *Achiever*, *Explorer*, *Socializer* und *Killer* auf. Dabei sucht der *Achiever* nach klaren granularen Zielen, die ihnen Statussymbole und Status in der Gemeinschaft bringen. Der *Explorer* versucht, mehr gesehen und verstanden zu haben, als andere Spieler. Auch möchte er mehr wissen als andere und möchte sein Wissen weitergeben. Der *Socializer* ist primär an der Interaktion mit anderen Spielern interessiert. Die eigentlichen Spielmechaniken interessieren in nur beiläufig. Der *Killer* hat einen etwas destruktiven Charakter, er will nicht nur gewinnen, er will auch, dass seine Gegner scheitern.

Eine Modifikation des Charaktermodelles ist die Motivationsmatrix von Radoff (vgl. [RADO11]). In Abbildung 6 werden die Spielermotivationen auf zwei Achsen aufgetragen. Erstens die Anzahl der Spieler, vom Einzelspieler bis zu sehr vielen Spielern. Zweitens die Art wie die Spieler „gewinnen“. Dabei wird oben die

qualitativen Belohnungen wie beispielsweise Emotionen und Geschichten und unten die quantitativen Belohnungen wie etwa Punkte und Bestenlisten aufgetragen.

Zu diesen Charaktermodellen und Motivationsmatrizen kommt laut Breuer noch die Motivation, dass sich die Nutzer kreativ ausdrücken wollen (vgl. [BREU11A]). Beispiele hierfür sind das Personalisieren eines Profils oder das Ändern des Designs der Webseite.

#### 2.5.3.2 RISIKEN

Beim Einsatz von Gamification sollte bedacht werden, wie Gamification zum in 2.5.1 erwähnten Gemeinschaftsmodell kompatibel ist. Viele der Spielmechaniken sind von der Mechanik her eher der Konkurrenz der Nutzer untereinander förderlich.

Auch besteht die Gefahr, dass Verhalten gefördert wird, das nicht mehr oder nicht in dem Ausmaße positiv ist. (vgl. [MORG12]) Weiterhin besteht die Möglichkeit, dass der eigentliche Sinn des Bewertungssystems in den Hintergrund gedrängt wird und sich die Nutzer primär mit den Gamification-Elementen auseinandersetzen. Zudem wird ein Teil der Nutzer der Meinung sein, sie würden manipuliert und wie Kinder behandelt. (vgl. [BREU11B] & [FEIN12])

#### 2.5.4 BEWERTUNGSGEGENSTAND & BEWERTUNGSART

In Bezug zur Motivation und zu den Gefühlen, die ein Nutzer mit dem System verbindet, ist es auch wichtig zu bedenken was und wie bewertet werden soll. Wie dieses Bewerten aussieht, beeinflusst stark, wo sich das Bewertungssystem oder die Plattform auf der bei 2.5.1 dargestellten Gemeinschaftsmodellskala wiederfindet.

Insbesondere Reputationssysteme (siehe 2.2.5) stehen oft diametral zum Gemeinschaftsmodell, wenn der Nutzer direkt bewertet werden kann. Diese direkte Nutzerbewertung führt dazu, dass sich Gruppen von Nutzern gegen andere Nutzer verbünden können, um sie zu ärgern. Desweiteren ist es auch für den Rezensenten ein unangenehmes Gefühl, wenn er jemand anderes bewerten soll. Auch das kann zu verfälschten Bewertungen führen. (vgl. [GLAS08])

Um solche Probleme zu vermeiden, kann es auch bei dem Reputationssystem sinnvoll sein, die Nutzer die erzeugten Inhalte der anderen Nutzer bewerten zu lassen, und dann aus diesen Bewertungen die Reputation der erstellenden Nutzer zu ermitteln (siehe auch 2.3.1). Sollten mehrere Aktionsmöglichkeiten auf der Plattform gegeben sein, so ist es sinnvoll, diese mit unterschiedlicher Gewichtung in die Reputation zu übernehmen. Beispielsweise ist es bei Punktereputationen (siehe 2.5 oder 2.5.3) gut vermittelbar, diese verschiedenen Interaktionsweisen zu gewichten, dass der Nutzer für manche Aktionen mehr Punkte erhält als für andere.

## 2.6 SIGNALE

Die Signale, die ein Bewertungssystem nutzt, sind entscheidend dafür, wie gut dessen Ergebnisse sind. Allerdings hängen die Signale, die für ein Bewertungssystem genutzt werden können, stark von dem Kontext, in dem sie eingesetzt werden sollen, ab. Das hat zur Folge, dass es nur sehr grobe Vorgaben geben kann, was ein gutes Signal ist. Ein Signal kann in einem Kontext gut Ergebnisse liefern und in einem anderen Kontext unzureichend sein. (vgl. [FAGL10])

Daher sollte die Plattform des Bewertungssystems analysiert werden. Insbesondere sollte beachtet werden welche Aktionen ein Nutzer ausführen kann. Bei diesen Nutzer-Aktionen müssen besonders die berücksichtigt werden, welche eine wertige Aussage über den Inhalt geben; aber auch indirekte Signale können gute Hinweise auf die Qualität des Inhalts geben. Nach Glass gibt es beispielsweise die folgenden Interaktionsmöglichkeiten in einem Forum (vgl. [GLAS08]). An diesen wird deutlich, dass viele der Signale nicht aus explizierten Bewertungsaktionen der Nutzer stammen, sondern implizit sind.

Auch sollte ein Signal, welches stark in die Bewertung eingeht, nicht beliebig wiederholbar sein, um zu verhindern, dass die Nutzer das System manipulieren (vgl. [GLAS08]). Wenn beispielsweise ein Nutzer Reputation für das Hochladen eines Bildes bekommt und dann beginnt qualitativ minderwertige Bilder hochzuladen, um seine Reputation zu steigern (siehe auch 2.7.2).

Bei Signalen, die der Nutzer explizit im Nutzerinterface nutzt, ist es notwendig, dass dem Nutzer auch die semantische Bedeutung der Aktion klar ist, da sonst das Signal

nicht zuverlässig auszuwerten ist. Ein Beispiel ist die Möglichkeit, dass der Nutzer einen Stern bei einem Inhalt setzt. Ohne zusätzliche Erläuterung ist für den Nutzer nicht ersichtlich, ob es sich bei dem Stern um eine Qualitätsaussage oder um z.B. ein Lesezeichen oder ähnliches handelt. Diese Mehrdeutigkeit findet sich dann auch bei der Auswertung dieses Signals. Bei der Auswertung ist nicht eindeutig, welche semantische Bedeutung der Nutzer der Aktion beigemessen hat. Es ist vorteilhaft, wenn mögliche mehrdeutige Handlungen jeweils als einzelne Handlungen im Benutzerinterface zu finden sind. Dann ist es möglich, diese konsistent auszuwerten. (vgl. [GLAS08])

## 2.7 PROBLEME BEI BEWERTUNGSSYSTEMEN

Es ist problematisch, wenn bei indirekten oder bei nutzerbasierten Bewertungssystemen entweder die Qualität des Inhalts oder die des Proxys nicht angemessen abgestuft werden. Dies hat zur Folge, dass frischer Inhalt es deutlich schwerer hat, alten Inhalt abzulösen. Das kann verhindert werden, indem die Gewichtung eines Inhaltes mit der Zeit nachlässt oder die Bewertungen neuerer Inhalte mit der Zeit stärker werden. Das gleiche Prinzip lässt sich auch auf die Proxys bei indirekten Bewertungssystemen anwenden, um beispielsweise die Reputation eines Nutzers relativ sinken zu lassen, wenn er lange Zeit nicht mehr aktiv war. (vgl. [FAGL10, p.93])

Bei der Implementation des Bewertungssystems ist ein Spielraum zum Anpassen des Systems und zum Einführen neuer Signale, Modelle und Signalgewichtungen hilfreich, so Farmer und Glass (vgl. [FAGL10, p.93]).

### 2.7.1 VERSAGEN DER SYSTEME

Bewertungssysteme versagen, wenn sie die Inhalte nicht mehr sinnvoll einordnen können. Diese Einordnung muss aus Sicht der Nutzer und des Betreibers des Bewertungssystems erfolgen.

Das Versagen kann verschiedene Ursachen haben. Eine ist der direkte Angriff auf das Bewertungssystem (siehe auch 2.7.2). Außerdem können die Annahmen auf denen das Bewertungssystem beruht, nicht in allen Fällen zutreffend sind. Beispielsweise Transaktionen, an denen mehr Personen beteiligt sind als modelliert.

## 2.7.2 ANGRIFFE AUF BEWERTUNGSSYSTEME

Ein Angriff auf ein Bewertungssystem ist die aktive, kontraproduktive Beeinflussung der Ergebnisse des Systems durch Personen. Die Erzeuger von Inhalten haben ein, oft monetäres, Interesse daran, dass ihre Inhalte von vielen Personen gesehen werden. Ein Grund dafür ist, dass im Internet viele Angebote über Werbung finanziert werden. Allerdings versuchen manche Nutzer, das Bewertungssystem aus nicht monetären Gründen zu manipulieren. Beispielsweise um die Möglichkeit der Manipulation zu zeigen. Andern Nutzer zu trollen<sup>1</sup> oder versuchen dem Betreiber des Bewertungssystems eins auszuwischen. (vgl. [FAGL10, p.89])

Um diese Szenarien zu vermeiden, ist es von Vorteil, wenn Nutzern des Systems nicht bewusst ist, wie das System technisch funktioniert. Dabei sollten vor allem die Details der genutzten Algorithmen, Signale und Gewichtungen möglichst geheim gehalten werden. Ein Beispiel für diese Geheimhaltung sind Suchmaschinen und unethisches SEO<sup>2</sup>. Die Suchmaschinen halten ihre Bewertungssysteme und Signale größtenteils geheim und geben nur grobe Details heraus. Das unethische SEO versucht die Funktionsweise der Suchmaschine zu durchschauen, um ein übermäßig hohes Ranking zu erhalten. (vgl. [FAGL10, pp.89–90])

## 2.7.3 BOTS UND SOCKENPUPPEN

Ein weiteres Problem für Bewertungssysteme, insbesondere bei nutzerbasierten oder indirekten Bewertungssystemen, ist das Entdecken von Bots und Sockenpuppen. Bots sind automatisierte Programme, welche versuchen, die Bewertungen zu manipulieren oder Spam zu erstellen. Sockenpuppen sind Online Identitäten, die zusätzlich zu einer originalen Identität erstellt werden, um zu täuschen (vgl. [WIKI12]). Dabei gibt die Sockenpuppe vor nicht die originale Identität zu sein.

---

<sup>1</sup> Troll [http://de.wikipedia.org/wiki/Troll\\_\(Netzkultur\)](http://de.wikipedia.org/wiki/Troll_(Netzkultur))

<sup>2</sup> Suchmaschinenoptimierung <http://de.wikipedia.org/wiki/Suchmaschinenoptimierung>

Bei Bewertungssystemen können Bots genutzt werden, um menschliche Nutzer zu imitieren, um das Bewertungssystem zu stören oder aber auch, um die Agenda des Bot-Besitzers zu stärken.

Sockenpuppen werden dazu verwendet, die eigene Position in einem Bewertungssystem zu stärken. Bei Bewertungssystemen, die Kommentare beinhalten, kann die Sockenpuppe auch dazu eingesetzt werden, die fremde Position zu unterminieren. (vgl. [FAGL10, p.269])

Die Zahl der Bots kann beschränkt werden, indem es aufwändiger gemacht wird, Accounts für das Bewertungssystem zu erstellen. Beispielsweise mithilfe von CAPTCHAs<sup>3</sup> oder durch das Nutzen eines Karma-modells bzw. Reputationssystems, das menschliche Aktionen von maschinellen unterscheiden kann.

Das Erkennen von Sockenpuppen ist schwierig. Da hierbei ein echter Mensch die Aktionen vornimmt, ist er nicht durch Bot-Erkennungsmaßnahmen zu entdecken. Indikatoren für Sockenpuppen sind Anmeldevorgänge aus demselben Browser bzw. von der gleichen IP-Adresse.

## 2.8 BEISPIEL: REDDIT

Reddit ist eine soziale Plattform, auf der Webinhalte oder Reddit eigene Inhalte verlinkt und bewertet werden. Diese Inhalte können auch auf Reddit kommentiert werden. Reddit ist ein Open-Source-Projekt unter der "Common Public Attribution License Version 1.0 (CPAL)", diese Lizenz schließt kommerzielle Nutzung aus. (vgl. [SPLA12])

Zu jedem Eintrag, sei es nun ein Link oder ein Kommentar, gibt es ein Bewertungsinterface, in welchem der Nutzer eine positive oder negative Bewertung zu diesem Inhalt abgeben kann. Dabei wird jeder Eintrag in einem Sub-Reddit erstellt. Diese Sub-Reddits werden durch die Nutzer geschaffen und sind jeweils verschiedenen Themen zugeordnet. Aus den populären Themen der einzelnen Sub-

---

<sup>3</sup> Completely Automated Public Turing test to tell Computers and Humans Apart

Reddits wird dann eine Seite mit aggregierten Inhalten gebildet, „Frontpage“ genannt.

### 2.8.1 BEWERTUNGSSYSTEM UND NUTZERBINDUNG

Mit diesen Interaktionsformen bewegt sich Reddit im Bereich eines nutzerbasierten Bewertungssystems. Alle wichtigen Bewertungssignale, welche die Einordnung auf der Webseite betreffen, werden von den Nutzern erzeugt. Als Bewertungs-Modell nutzt Reddit dafür das Modell *Punkte* (siehe auch 2.3.1). Zusätzlich nutzt Reddit *Flags* um regelwidrige Inhalte zu melden. Dieses *Flag* hat aber keinen direkten Einfluss auf die Bewertung des Inhalts, außer der Inhalt wird entfernt. Zusätzlich werden Experten eingesetzt, um die Inhalte mit *Flag* endgültig zu bewerten und zu entfernen, falls sie regelwidrig sind.

Ein weiteres Bewertungssystem lernt aus den Spam-Flags und aus dem, was die Moderatoren als Spam markieren. Dieses System versucht dann automatisch anhand des Gelernten, neuen Spam zu entdecken. Damit gehört es in die analysebasierten Bewertungssysteme. (vgl. [KETR08])

Um die Nutzerbindung aufrecht zu erhalten, nutzt Reddit ein minimales Gamification-Element, nämlich Punkte (siehe auch 2.5.3). Jeder Nutzer hat zwei Punktestände. Erstens einen Punktestand für die bei Reddit eingereichten Links und zweitens einen Punktestand für die bei Reddit abgegeben Kommentare. In beiden Fällen funktioniert es so, dass die Summe (Positive Bewertungen – Negative Bewertungen) den Punktestand ergibt. Dieser Wert kann auch ins Negative fallen. Trotz dieser Einfachheit entwickelt sich eine starke Dynamik um diesen als „Karma“ bezeichneten Punktestand.

Das Gemeinschaftsmodell von Reddit ist dabei *konkurrierend*. Die negativen Bewertungen ermöglichen es, die Inhalte von konkurrierenden Inhaltserstellern abzuwerten. Die Nutzer wetteifern miteinander, da nur Wenige die Top-Positionen in der Auflistung erreichen können. Insbesondere das Erreichen der „Frontpage“ ist schwierig. Das gewählte Gemeinschaftsmodell und das Bewertungsmodell passen durch die Möglichkeit, Inhalte auch negativ zu bewerten, gut zusammen. Die



Motivation der Nutzer (siehe auch 2.5.2), die Links einstellen und nicht nur kommentieren, beruht auf Eigeninteresse.

## 2.8.2 AUSWERTUNG

Am Beispiel Reddit lässt sich aufzeigen, wie verschieden Auswertungsansätze genutzt werden, um für die Nutzer relevante Inhalte anzuzeigen. Reddit bietet dem Nutzer dann vier verschiedene Auswertungs-Systeme im Benutzerinterface an: „Beliebt“, „Neu“, „Kontrovers“, „Top“ und zusätzlich für Kommentare „Best“.

Das „Beliebt“ genannte Auswertungsmuster bildet dabei eine Auswertung über die positiven und negativen Bewertungen und den Einreichungszeitpunkt. Dabei kommt die Formel in Abbildung 8 zum Einsatz:

Diese Formel hat den Vorteil, dass sie aggressiv den Einreichungszeitpunkt beachtet, was verhindert, dass Inhalte übermäßig lange Zeit angezeigt werden. Um dies zu erreichen, nimmt der Wert der positiven Bewertungen ab, je mehr positive Bewertungen vorhanden sind.

Das Auswertungsmuster „Neu“ beachtet nur den Einreichungszeitpunkt und bewertet die Einträge hoch, die am neuesten sind.

„Kontrovers“ hebt Inhalte hervor, welche eine große, aber ausgewogene Anzahl an positiven und negativen Bewertungen haben und die häufig kommentiert wurden.

Das „Top“ erlaubt es dem Nutzer einen Zeitraum anzugeben, und bewertet dann in diesem diejenigen Inhalte besonders hoch, welche die meisten absoluten positiven Bewertungen erhalten haben.

Die Kommentarauswertung „Best“ wurde von Reddit eingeführt um zu verhindern, dass die zuerst erstellten Kommentare dauerhaft als beste angezeigt werden. Das System versucht aus den positiven und negativen Bewertungen einen Konfidenzwert zu ermitteln, um die verschiedenen Kommentare einzuordnen. Dazu nutzt es die untere Grenze des Wilson Intervalls. (vgl. [MUNR09]) Abbildung 9 zeigt das Intervall wie es von Reddit genutzt wird. Die untere Grenze wird errechnet, wenn Minus ausgewählt wird, die obere bei Plus.

# 3 KRITERIENKATALOG FÜR BEWERTUNGSSYSTEME

---

Der hier bereitgestellte Kriterienkatalog soll die Entscheidungsfindung unterstützen. Hierzu ermöglicht er die Auswahl eines Bewertungssystems und die dazugehörigen Modelle. Außerdem lässt er den Anwender des Kriterienkatalogs über weitere Teilbereiche eines Bewertungssystems reflektieren. Dazu gehören die Risiken eines Bewertungssystems sowie die genaue Ausprägung. Auch die Nutzung von Methoden zur Steigerung der Nutzermotivation im Kontext des Bewertungssystems wird erörtert. Der Kriterienkatalog benötigt bei Anwendung Wissen über den Kontext des Bewertungssystems und Grundkenntnisse in der Softwareentwicklung.

## 3.1 AUFWAND & INFRASTRUKTUR

<b>Aufwand &amp; Infrastruktur</b>		
<b>1) Wie groß ist der Implementationsaufwand der Software?</b>		
Hoch	Mittel	Gering
<b>2) Wird neue Infrastruktur benötigt?</b>		
Ja	Nein	
<b>3) Muss das System laufend gepflegt werden?</b>		
Ja	Nein	
<b>4) Ist das System von Teilen der bestehenden Software abhängig?</b>		
Ja	Nein	

<b>5) Kann das System auf bestehende Infrastruktur aufbauen?</b>	
Ja	Nein
<b>6) Kann das System auf bestehende Software aufbauen?</b>	
<b>Ja</b>	Nein

Der folgende Abschnitt des Kriterienkatalogs beschäftigt sich mit dem Aufwand der zur Erstellung des vorgesehenen Bewertungssystems erforderlich ist. Dieser Aufwand sollte vom Anwender des Kriterienkatalogs bedacht werden, da ein zu hoher Aufwand im schlimmsten Fall das Erstellen des Bewertungssystems scheitern lässt. Allerdings ist es auch möglich, dass das Bewertungssystem nur limitiert implementiert wird und es bei diesem Implementierungsgrad keine guten Ergebnisse erzeugt.

**Zu 1):**

Das Kriterium zielt darauf ab, dass sich der Anwender Gedanken über den Implementationsaufwand macht. Sollte dieser unterschätzt werden, kann das Projekt zum Erstellen des Bewertungssystems scheitern. Wie erwähnt, kann es auch die Folge haben, dass nur ein Sub-Set der gewünschten Fähigkeiten des Bewertungssystems implementiert werden und daher die Ergebnisse qualitativ nachlassen. In den meisten Fällen ist ein „machine-learning“ System eher in der Kategorie „Hoch“ einzuordnen und ein System mit Nutzerbewertungen mit nur wenigen Modellen eher in der Kategorie „Gering“ einzuordnen. Im Bereich „Mittel“ sind Systeme mit vielen Modellen oder mit komplexen Auswertungsfunktionen einzuordnen.

**Zu 2):**

Bei diesem Kriterium soll der Anwender überlegen, ob er die benötigte Infrastruktur für sein Projekt hat. Dabei sollte vor allem darauf geachtet werden, ob die Infrastruktur die zu erwartenden Datenmengen speichern und die zu erwartende Last aushalten kann.

Zu 3)

Dieses Kriterium soll Bewusstsein darüber schaffen, ob für den Betrieb des Systems ein dauerhaftes Monitoring notwendig und ob dafür das nötige Personal vorhanden ist.

Zu 4)

Es soll Bewusstsein darüber geschaffen werden, ob das Bewertungssystem von bestehender Software abhängig ist, da eventuelle Änderungen in diesen Abhängigkeiten neuen Entwicklungsaufwand auslösen können.

Zu 5)

Wenn das System auf bestehende Infrastruktur aufbauen kann, können die Kosten und der Aufwand sinken.

Zu 6)

Sollte das System auf bereits bestehende Software aufsetzen können, so sinken die Kosten und der Aufwand.

## 3.2 GRUNDLAGEN DES BEWERTUNGSSYSTEMS

<b>Grundlegendes</b>	
<b>7) Soll das Bewertungssystem in einen existierenden Kontext eingebunden werden?</b>	
Ja	Nein
<b>8) Sollen Nutzer den Inhalt bewerten?</b>	
Ja	Nein
<b>a. Bei Ja: Ist die Motivation der Nutzer ein Problem?</b>	
Ja	Nein

<b>9) Wie wichtig ist die korrekte Bewertung aller Inhalte?</b>				
Hoch	Mittel			Gering
<b>10) Wie viele Inhalte sollen bewertet werden?</b>				
Viele	Einige			Wenige
<b>11) Wie strukturiert sind die Inhalte?</b>				
Sehr	Mittel			Gering
<b>12) Wie inhaltlich komplex sind die Inhalte?</b>				
Hoch	Mittel			Gering
<b>13) Gibt es Proxys, mit deren Eigenschaften die Qualität des Inhaltes korreliert?</b>				
Ja			Nein	
<b>14) Welche Typen von Bewertungssystemen sollen genutzt werden?</b>				
Analyse	Nutzer	Kombination	Nutzer	Experten
		und Analyse		Indirekte

Dieser Abschnitt beschäftigt sich mit den grundlegenden Anforderungen, die an das Bewertungssystem gestellt werden. Die Grundlagen beeinflussen die genaue Ausprägung des Bewertungssystems.

Zu 7)

Oft werden Bewertungssysteme nachträglich eingeführt oder erweitert (vgl. [FAGL10, p.223]. Beispiel hierfür sind Shop-Seiten, die ein Bewertungssystem für die Nutzer-Reviews einführen.

Zu 8)

Die Bewertung durch Nutzer ist eine Vorgabe, da über das Bewertungssystem auch die Nutzerloyalität erhöht werden soll.

### Zu 9)

Die Qualität der Ergebnisse eines Bewertungssystems ist je nach Typ unterschiedlich (siehe auch 2.2). Expertenbewertung liefert die besten Ergebnisse, während bei den anderen Bewertungssystemen eine hohe Varianz in der Ergebnisqualität besteht. Bei nutzerbasierter Bewertung werden beispielsweise nicht alle Inhalte der gleichen Aufmerksamkeit zuteil (siehe auch 2.2.2).

### Zu 10)

Die Menge der Inhalte bedingt das Bewertungssystem insofern, dass nicht alle Systeme unter realistischen Bedingungen sehr große Datenmengen verarbeiten können. Die Expertenbewertung kann nur geringe Inhaltsmengen verarbeiten, bedingt durch die hohen Kosten. Aber auch die nutzerbasierte Bewertung ist durch die Menge der Nutzer limitiert.

### Zu 11)

Wenn die Inhalte stärker strukturiert sind, haben es Analysebewertungssysteme einfacher, diese einzulesen und zu verarbeiten. Beispiele für unstrukturierte Daten sind Bilder oder Videos, da diese nicht einfach von einem Algorithmus analysiert werden können.

### Zu 12)

Wenn die zu bewertenden Inhalte eine hohe Komplexität aufweisen, ist es schwierig, sie mit Algorithmen zu analysieren. Auch kann es für Nutzer schwer sein, gewisse Eigenschaften des Inhaltes richtig zu bewerten, wenn ihnen Wissen über den Inhalt oder Kontext fehlt. Sollte beispielsweise ein Auto durch Crowdsourcing<sup>4</sup> designt werden, ist es den Nutzern nicht möglich zu bewerten, ob dieses Auto auch auf einer bestimmten Produktionsstraße gebaut werden kann. (z.B. vorgehensweise bei Fiat)

---

<sup>4</sup> Crowdsourcing: Bestimmte Aufgaben an Freizeitarbeiter auslagern - <http://en.wikipedia.org/wiki/Crowdsourcing>

Zu 13)

Wenn ein solcher Proxy existiert, ist dies ein starker Indikator für die (Mit)Nutzung eines indirekten Bewertungssystems. Oft ist ein solcher Proxy der Nutzer, der den Inhalt erstellt hat. (siehe auch 2.2.5)

### 3.3 MODELLE

<b>Modelle</b>					
<b>15) Sollten die Nutzer die Inhalte bewerten?</b>					
Ja	Nein				
<b>16) Bei Ja: Ist die Qualität des Inhaltes ein einfaches oder komplexes Merkmal?</b>					
Einfach	Komplex				
<b>17) Bei Einfach: Soll jeder Nutzer eine differenzierte Aussagen treffen?</b>					
Ja	Nein				
<b>18) Bei Komplex: Können die Qualitätsmerkmale konkret ausgedrückt werden?</b>					
Ja	Teilweise	Nein			
<b>19) Welche Art von Bewertungsmodell bietet sich an?</b>					
Einfach	Kombiniert				
<b>20) Welche Modelle sollen genutzt werden?</b>					
Favoriten und Abstimmen Flags	This-or-That	Ratings	Rezensionen	Punkte	Karma

Die Modelle eines Bewertungssystems sind die grundlegenden Bausteine eines Systems. Aus diesen stammen die expliziten Signale, die die Bewertung ermöglichen.

#### Zu 15)

Gewisse Bewertungsmodelle sind nur im Kontext von Nutzerbewertung sinnvoll einsetzbar.

#### Zu 16)

Einfache Merkmale lassen sich leicht auf Punkte oder Ratings reduzieren. (siehe auch 2.3.1) Beispielsweise bei Inhalten, die sich in Gut/Schlecht einordnen lassen. Bei komplexen Merkmalen hat die Qualität des Inhaltes viele Facetten, ein Beispiel hierfür ist die Rezension einer Geschichte, welche neben Gut/Schlecht auch andere Bewertungsaspekte hat.

#### Zu 17)

Die differenzierte Aussage ist eine Abstufung des Qualitätsgrades. Beispielsweise Sternbewertungen bei Amazon. Undifferenzierte Aussagen bedeutet, dass binäre Positiv-Negativ Bewertungen oder reine Positiv-Bewertungen genutzt werden können. (siehe auch 2.3.1)

#### Zu 18)

Wenn die Qualitätsmerkmale konkret ausdrückbar sind, können Reviews mit den verschiedenen Merkmalen eingesetzt werden. Sollten die Merkmale nur teilweise oder gar nicht konkret ausgedrückt werden, so sollte dem Bewerter beispielsweise mithilfe von Freitext-Eingabe ermöglicht werden, den Inhalt zu bewerten. Ein Beispiel sind dabei die Nutzerreviews in Webshops, welche ein Rating und Freitext nutzen, um etwas zu bewerten. (siehe auch 2.3.1)

#### Zu 19)

Die kombinierten Bewertungsmodelle bieten sich an, wenn die Genauigkeit der Bewertung höher sein soll, als diese durch die Verwendung von einfachen Bewertungssystemen erreicht werden kann. Dazu bieten sich vor allem Modelle wie Karma an. Weiterhin ermöglicht es die Kombination von Bewertungsmodellen, dem Nutzer die Möglichkeit zu geben, seine Aussage zu einem Inhalt zu konkretisieren.



### 3.4 RISIKEN

<b>Risiken</b>	
<b>21) Wird das System nachträglich eingeführt?</b>	
Ja	Nein
<b>a. Bei Ja: Müssen die Nutzer/Kunden vom System überzeugt werden?</b>	
Ja	Nein
<b>22) Kann das System von böartigen Nutzern beeinflusst werden?</b>	
Ja	Nein
<b>a. Bei Ja: Wie kann das System von böartigen Nutzern beeinflusst werden?</b>	
<b>b. Kann das System Angriffe erkennen?</b>	
Ja	Nein
<b>c. Kann das System Angriffe revidieren?</b>	
Ja	Nein
<b>d. Kann das System Angriffe minimieren?</b>	
Ja	Nein
<b>23) Ist das System gegen Bots geschützt?</b>	
Ja	Nein
<b>24) Kann das System unerwünschte Inhalte filtern?</b>	
Ja	Nein

Die Risiken eines Bewertungssystems beeinflussen die Adaption des Systems bei den Nutzern und die Probleme, die der Einsatz eines Bewertungssystems bewirken kann.

#### Zu 21) und 21a)

Wenn das Bewertungssystem nachträglich in einen Kontext eingeführt wird, besteht die Gefahr, dass die Nutzer die Verwendung ablehnen oder versuchen, das System zu manipulieren.

#### Zu 22) a-d

Die meisten Systeme können von bösartigen Benutzern beeinflusst werden. Es sollte sichergestellt werden, dass diese bösartigen Nutzer erkannt werden und die Beeinflussung revidiert oder minimiert werden kann. Dabei bedeutet revidieren, dass das System die Bewertung auf den Stand bringen kann, auf dem sie ohne den Angriff gewesen wäre. Minimieren eines Angriffs bedeutet, dass dieser die Bewertung nicht übermäßig beeinflussen kann. Weiterhin ist das System in der Lage Angriffe selbstständig zu erkennen, um die erwähnten Gegenmaßnahmen einzuleiten.

#### Zu 23)

Wenn das System gegen Bots geschützt ist, so ist die Chance auf erfolgreiche Angriffe auf das Bewertungssystem reduziert, da viele Angriffe auf dem massiven Bewerten bestimmter Inhalte mittels Bots basieren.

#### Zu 24)

Das Bewertungssystem sollte in der Lage sein, Inhalte zu filtern, die gegen die Regeln der Plattform oder gegen das Gesetz verstoßen.

### 3.5 ZIELE & INHALTE

<b>Ziele &amp; Inhalte</b>			
<b>25) Was sind die Ziele des Bewertungssystems?</b>			
Inhaltsqualität erhöhen	Inhalte einstufen	Inhalte filtern	Anderes
<b>26) Wird eine hohe Nutzerloyalität angestrebt?</b>			
Ja		Nein	
<b>27) Wer moderiert die Inhalte?</b>			
Nutzer	Betreiber	Beide	
<b>28) Verlieren die Inhalte mit der Zeit an Relevanz?</b>			
Ja		Nein	
<b>29) Soll mit den Inhalten Wert geschaffen werden?</b>			
Ja		Nein	
<b>30) Müssen die Nutzer moderiert werden?</b>			
Ja		Nein	
<b>31) Wer soll die Nutzer moderieren?</b>			
Nutzer	Betreiber	Beide	

Die Ziele und Inhalte eines Bewertungssystems beeinflussen die genaue Ausprägung des Systems, da sie bestimmen, wie die Nutzer das System wahrnehmen.

#### Zu 25)

Die Ziele eines Bewertungssystems sind vielfältig (vgl. [FAGL10, p.98]). Die Qualität der Inhalte kann durch Bewertung in dem Maße werden wie die als hochwertig betrachteten Inhalte prominent dargestellt werden und die Nutzer bei der Erzeugung von Inhalten sich an diesen orientieren. Ein Bewertungssystem eignet sich auch dafür, die Inhalte in eine Ordnung zu bringen, wie bei Reddit (siehe auch 2.8). Das Filtern von unerwünschten, regelwidrigen oder verbotenen Inhalten ist ebenfalls ein Ziel der Bewertung.

#### Zu 26)

Eine erhöhte Loyalität der Nutzern führt zu einer höheren Aktivität und somit zu besseren Daten im Bewertungssystem, aber auch beispielsweise zu einem höheren Umsatz durch Werbung.

#### Zu 27)

Die Moderation der Inhalte kann anstatt durch den Betreiber auch über das Bewertungssystem erfolgen. Dabei können Angestellte des Betreibers entweder nur als letzte Instanz involviert oder gar nicht.

#### Zu 28)

Inhalte können mit der Zeit an Relevanz verlieren (Aktualität von Nachrichten) und das Bewertungssystem muss diesen Relevanz-Nachlass abbilden können. (siehe auch 2.8)

#### Zu 29)

Diese Wertschöpfung hat Auswirkungen auf das Bewertungssystem. Es ist wichtig, dass hierbei das Bewertungssystem dafür sorgt, dass diese Inhalte möglichst regelkonform zu den Vorgaben der Plattform sind. Wird Werbung zusammen mit dem Inhalt angezeigt, sollte der Inhalt dem Auftraggeber zusagen, damit dieser weiterhin Anzeigen schaltet.

### Zu 30)

Insbesondere in sozialen Anwendungen kann es dazu kommen, dass die Nutzer miteinander in Konflikt geraten. Beispielsweise mit Freitext-Reviews welche, genutzt werden können, um andere Nutzer verbal anzugreifen.

### Zu 31)

Das Bewertungssystem kann die Moderation der Nutzer unterstützen, indem es Funktionen anbietet mit denen Nutzer, welche sich nicht an die Regeln der Plattform halten, gemeldet werden. Diese Meldung kann durch andere Nutzer der Plattform erfolgen oder durch Angestellte des Betreibers. Wie auch bei 27) kann das Moderieren gänzlich innerhalb der Community erfolgen oder Angestellte des Betreibers involvieren.

## 3.6 NUTZERBINDUNG

<b>Nutzerbindung (bei Nutzerbewertung)</b>		
<b>32) Ziehen die Nutzer aus dem Bewerten einen Nutzen?</b>		
Ja	Nein	
<b>33) Ist das Bewertungssystem kurzweilig?</b>		
Ja	Nein	
<b>34) Welche Motivation haben die Nutzer?</b>		
Eigeninteresse	Interesse an Anderen	Interesse an dem Bewertungsobjekt

### Zu 32)

Wenn die Nutzer keinen Nutzen aus dem Bewertungssystem ziehen, sind sie nicht langfristig motiviert und hören auf, das Bewertungssystem zu verwenden (siehe auch 2.5).

Zu 33)

Das Bewertungssystem sollte, angemessen zum Kontext, kurzweilig sein, um die Nutzer zu animieren, oft und viel zu bewerten. Viele Bewertungen führen zu besseren Ergebnissen.

Zu 34)

Wenn ein Bewertungssystem designt werden soll, muss die Motivation der Nutzer bedacht werden. Die Nutzer müssen die Möglichkeit haben, *Eigeninteresse* und *Interesse in Andere* im Rahmen des Bewertungssystems auszuleben. Dazu zählt beispielsweise, dass die Nutzer miteinander kommunizieren und interagieren können. Es ist wichtig das zu Bewertende gut zu präsentieren, damit *Interesse am Bewertungsobjekt* geweckt und erhalten wird. Der Nutzer ist somit bereit, das Objekt umfassend zu bewerten.

### 3.7 GAMIFICATION

<b>Gamification</b>			
<b>35) Soll Gamification eingesetzt werden?</b>			
Ja		Nein	
<b>36) Welche Spielmechanik soll genutzt werden?</b>			
Punkte	Level	Anderes:	
<b>37) Bei Gamification: Welche Spielertypen werden erwartet?</b>			
Achiever	Explorer	Socializer	Killer
<b>38) Wurden die Risiken von Gamification bedacht?</b>			
Ja		Nein	

Zu 35)

Gamification kann die Nutzerbindung stark erhöhen, Daten für das Bewertungssystem liefern und somit die Bewertung genauer machen. Viele Nutzer lassen sich stark durch Spielmechaniken motivieren.

Zu 36)

Die gewählte Spielmechanik muss oft nicht besonders komplex sein, um eine starke Beteiligung und Motivation der Nutzer auszulösen. Simple Punkte- oder Level-Systeme reichen aus, um die Nutzerbindung stark zu steigern.

Zu 37)

Spielertyp und eingesetzte Spielmechaniken bedingen einander. Beispielsweise sollten für den Spielertyp *Socializer* soziale Elemente wie Freundeslisten oder das „Folgen“ von Personen (z.B. Follower bei Twitter) eingesetzt werden.

Zu 38)

Die Risiken von Gamification sind im Bereich der Nutzerakzeptanz und der Verhaltensweisen, die durch die Spielmechaniken verstärkt werden sollen zu finden. (siehe auch 2.5.3.2)

### 3.8 INDIREKTE BEWERTUNGSSYSTEME

<b>Indirekte Bewertungssysteme</b>		
<b>39) Sollen indirekte Bewertungssysteme eingesetzt werden?</b>		
Ja		Nein
<b>40) Welcher Proxy soll verwendet werden?</b>		
Nutzer	Nutzer-Gruppen	Anderes
<b>41) Soll den Nutzern eine Reputation angezeigt werden?</b>		
Ja		Nein

<b>a. Bei Ja: Wie soll die Reputation angezeigt werden?</b>				
Punkte	Level		Anderes	
<b>42) Bei Nutzer/Nutzergruppen: Welches Gemeinschaftsmodell soll bestehen?</b>				
Kümmern	Zusammenarbeit	Freundschaftlich	Konkurrierend	Kämpferisch

Zu 39)

Indirekte Bewertungssysteme können die Ergebnisse von anderen Bewertungssystemen unterstützen, indem sie deren Resultate mit permanenteren Bewertungswerten korrelieren.

Zu 40) & 41)

Insbesondere Nutzer und Nutzer-Gruppen werden von indirekten Bewertungssystemen verwendet. Dabei kann die Reputation entweder direkt dem Nutzer angezeigt und damit eine Art Spielmechanik eingeführt werden. Es ist auch möglich die Reputation nur systemintern zu nutzen. Beispielsweise kann die Reputation verwendet werden, um die Relevanz einer Meldung über einen Regelverstoß einzuschätzen. Nutzer, die oft korrekt Regelverstöße melden, werden vom Bewertungssystem als relevanter eingestuft müssen aber selbst nicht wissen, dass sie besonders gute Melder sind. (siehe auch 2.5.3 & 2.3.1)

Zu 42)

Wenn eine Community entstehen soll, muss das gewählte Gemeinschaftsmodell an das Interaktionssystem und Bewertungssystem angepasst werden. Dabei zählt vor allem, dass die Systeme und das Ziel des Bewertungssystems kohärent sind. Es vermieden werden, dass ein Teil des System sagt „macht euch gegenseitig fertig“ und ein anderes eher die gegenseitige Hilfe anstrebt. (siehe auch 2.5)



### 3.9 AUSWERTUNG

<b>Ergebnisse</b>		
<b>43) Wie entwickelt sich die Qualität der Auswertung von neuen Inhalten im Zeitverlauf?</b>		
Steigt	Bleibt gleich	Abnehmend
<b>44) Wie hoch wird die Fehlerrate des Systems geschätzt?</b>		
Hoch	Mittel	Gering
<b>a. Bei Hoch oder Mittel: Ist die Fehlerrate akzeptabel?</b>		
Ja	Nein	

#### Zu 43)

Die einzelnen Bewertungssysteme haben einen unterschiedlichen Ergebnisverlauf. Bei nutzerbasierter Bewertung muss die Nutzer erst mal etwas bewerten, was eine gewisse Zeit dauert. Wenn ein Inhalt nur durch einen Nutzer bewertet wurde, ist dies nicht aussagekräftig. Je mehr Nutzer den Inhalt jedoch bewerten, desto aussagekräftiger wird die Bewertung. Analysebasierte Bewertungssysteme können hingegen sofort nach Eingang eines neuen Inhalts den Inhalt bewerten. Die Aussagekraft der Bewertung ändert sich aber nicht, wenn man den gleichen Analyse-Prozess mehrmals den selben Inhalt verarbeiten lässt.

#### Zu 44)

Die Fehlerrate des Systems sollte dem Kontext des Bewertungssystems angemessen sein. Beispielsweise sollte ein Bewertungssystem zur Spamerkennung nicht zu viele false-positives oder false-negatives erzeugen, da diese dann manuell freigeschaltet bzw. gelöscht werden müssten. Allerdings macht die Bewertung durch Nutzer diesen Wert schwer einschätzbar, da die Nutzer eine eigene Agenda haben können, welche den Zielen des Bewertungssystems widerspricht. (siehe auch 2.7)

# 4 BEISPIELANWENDUNG

---

Dieses Kapitel beschreibt die Beispielanwendung, welche aus den Erkenntnissen der vorherigen Kapitel entstanden ist. Dabei geht das Kapitel auf die Motivation und Umsetzung der Anwendung ein. Es werden die eingesetzte Architektur und die eingesetzten Technologien dargestellt.

## 4.1 MOTIVATION & UMSETZUNG

Aus der Frage heraus, ob dem Nutzer im Delegs-Editor das GebärdenSchrift-Symbol angezeigt wird, welches gerade relevant für ihn ist, entstand die Idee, eine Anwendung zur Bewertung des Symbols – genannt Delegs-Voting – zu entwickeln. Dabei werden die oben entwickelten Kriterien der Bewertung berücksichtigt. Idealerweise zeigt das System immer das GebärdenSchrift-Symbol an, welches der Nutzer gerade verwenden möchte. Die vorliegende, prototypische Anwendung ist der erste Schritt in diese Richtung.

Unter Einbeziehung der bisherigen Erkenntnisse dieser Arbeit und unter Anwendung des erarbeiteten Kriterienkatalogs stellt sich die Frage, ob die Nutzerbewertung angesichts der Gegebenheiten das optimale Bewertungssystem ist.

Bei Anwendung des ersten Teils des Kriterienkatalogs wird deutlich, dass wegen der geringen Entwicklungsressourcen keine komplexen Systeme entwickelt werden können. Ein Gespräch mit einem GebärdenSchrift Experten ergab zudem, dass die Bewertung der GebärdenSchrift eine mittlere Komplexität besitzt. Einige Aspekte der Bewertung der Symbole sind für die Nutzer einfach zu erfassen, andere benötigen Hintergrundwissen. Seiner Meinung nach sind *Reviews* mit einigen Auswahlmöglichkeiten eine gute Möglichkeit, um die GebärdenSchrift-Symbole zu bewerten. Laut Aussage des Experten ermöglichen passend gewählte Kriterien für diese *Reviews* gute Aussagen über die Qualität und Verbreitung eines GebärdenSchrift-Symbols (allgemeine Gültigkeit oder lokale Variante) zu treffen. Als Gemeinschaftsmodell wird „Zusammenarbeit“ genutzt werden, da die Anwender das gemeinsame Ziel haben gute Gebärden-Schrift-Symbole zu nutzen.

Eine der Anforderungen an die Anwendung ist die Integrierbarkeit dieser in das Delegs-Projekt. Weiterhin sollen auch die Hürden bis zur Nutzung möglichst niedrig

sein. Diese beiden Anforderungen machen die Entwicklung als Webanwendung naheliegend. Die Ausführung der Anwendung als Webanwendung bzw. RIA<sup>5</sup> erfüllen diese beiden Anforderungen. Das Delegs-Projekt implementiert sein Hauptwerkzeug, den Delegs-Editor, als Webanwendung und HTML-Links machen die Integration einfach. Auch besteht die Möglichkeit die Bewertungsdaten zu exportieren, was die laufende Integration der von der Beispielanwendung erzeugten Daten mit dem Delegs-Editor vereinfacht. Zudem ist durch die Implementation als Webanwendung auch die Hürde bis zur Nutzung der Anwendung sehr gering. Es muss keine Software installiert werden und es sollte, einen modernen Browser vorausgesetzt, auch keine Kompatibilitätsprobleme geben. Der Nutzer muss sich nur anmelden bzw. registrieren und kann dann sofort GebärdenSchrift bewerten.

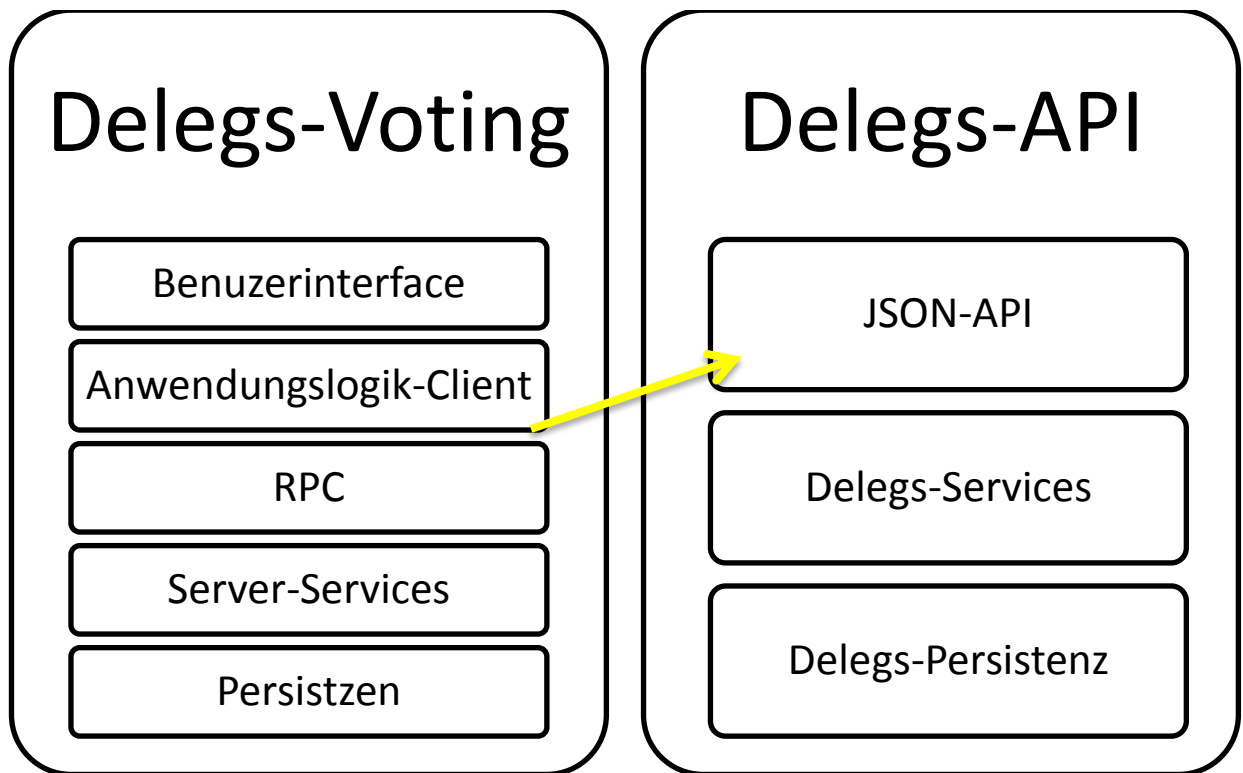
## 4.2 ARCHITEKTUR

Delegs-Voting nutzt eine Schichtenarchitektur, um einzelne Aufgabenteile voneinander zu trennen. Dies wird durch die Ausführung als Webanwendung unterstützt, weil dadurch eine grundlegende Trennung auf der Skala zwischen Persistenz und Benutzerinterface notwendig ist.

Delegs-Voting nutzt dabei ein vierteiliges Schichtenmodell mit den Schichten Persistenz, Server-Services, Anwendungslogik-Client und Benutzerinterface. Wobei die RPC-Schicht mit den Services von Delegs-Voting und mit der API des Delegs-Projektes kommuniziert.

---

<sup>5</sup> Rich Internet Application



Die erwähnte Trennung zwischen dem Benutzerinterface und der Client-Anwendungslogik wird durch den Einsatz des MVC-Patterns erreicht, was stark durch den GWT-UiBinder unterstützt wird. Da bei diesem das statische Layout in XML definiert wird, ist es nicht möglich Anwendungslogik in die UI-Schicht zu bauen.

In Abbildung 1 werden die Schnitte der Anwendung dargestellt. An der X-Achse sind die Schnitte aufgetragen und an der Y-Achse sind die Schichten angegeben. Dabei gibt es keine Verbindungen von Authentifikation zu Abstimmen und von Persistenz zu UI.

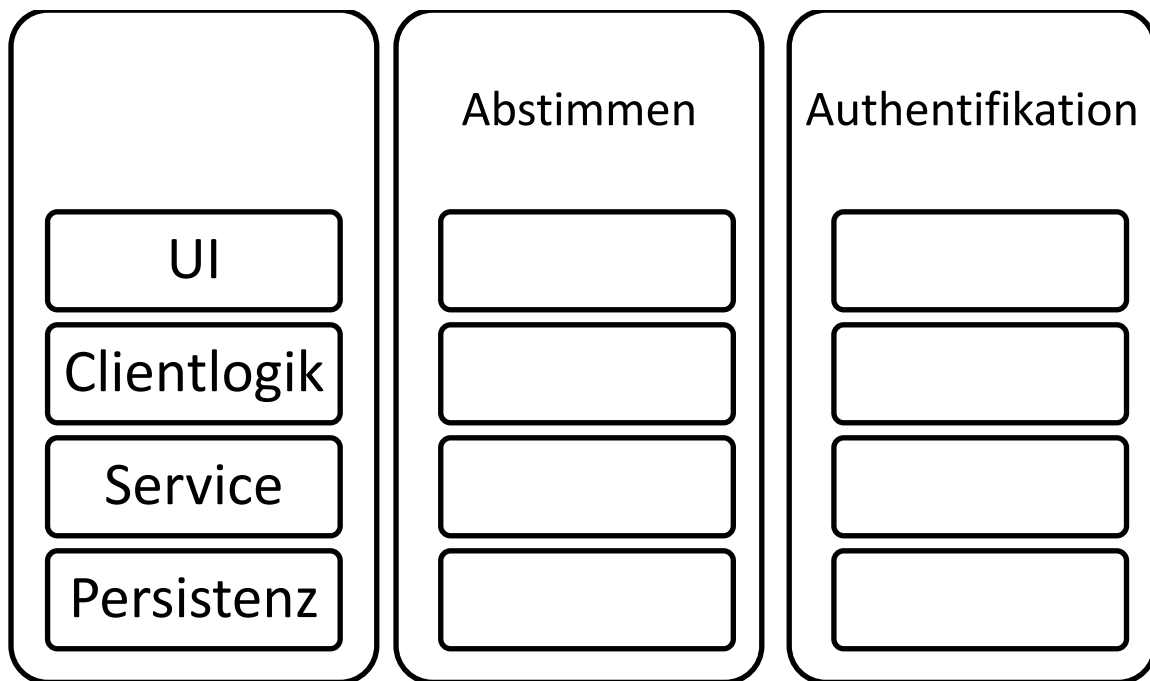


Abbildung 1 Schichten und Schritte bei Delegs-Voting

#### 4.2.1 UI-ENTWICKLUNG

Das Benutzerinterface (UI) soll einfach zu bedienen sein und der Nutzer soll sich auf das GebärdenSchrift-Symbol konzentrieren können. Durch die Reduzierung auf das Wesentliche wird versucht, die Hemmungen der Nutzer zu reduzieren und die Aufmerksamkeit des Nutzers voll auf die Bewertung der GebärdenSchrift zu lenken. Die erste Idee Abbildung 11 war daher recht simpel mit dem GebärdenSchrift-Symbol groß in der Mitte und der Bewertung darunter. Die Bewertung erfolgt eingeschränkt mit nur der positiven und negativen Bewertungsoption.

Nach der Vorstellung der ersten Variante wird in der weiterentwickelten Variante die Bewertung der GebärdenSchrift differenzierter betrachtet. So gibt es jetzt Abbildung 12 mehrere Kriterien, die verschiedene Aspekte eines GebärdenSchrift-Symbols bewerten sollten. Diese Bewertung wird hier mithilfe von Sternen als Qualitätsmaß umgesetzt.

In der endgültigen Version Abbildung 13 sind die Sterne durch Menüs ersetzt, da diese es besser erlauben, heterogene Antworten darzustellen. So kann eine Ja-Nein-Frage schlecht in Sternen ausgedrückt werden. Auch wird bei den Menüs deutlicher, was die einzelnen Bewertungsstufen bedeuten, da sie jetzt klar benannt

sind. Ein weiterer positiver Aspekt an den neuen Bewertungselementen ist, dass sie deutlich besser auf Touch-Geräten wie Smartphones oder Tablets bedient werden können. Dies ist der Fall, da solche Menüs speziell von diesen Geräten unterstützt werden und diese anstelle von Dropdown-Menüs wie auf dem Desktop spezielle modulare UI-Elemente anzeigen.

Die Anwendung funktioniert auch auf Mobilgeräten gut. Dies ist ein weiterer Vorteil der Implementierung in HTML und CSS, da es einfach ist, die Anwendung so zu erstellen, dass sie größtenteils ein relatives Layout benutzt und sich damit an die Größe des Geräts anpasst.

### 4.3 CLIENT-SERVER KOMMUNIKATION

Delegs-Voting nutzt zwei verschiedene Methoden zur Client-Server Kommunikation. Eine JSON-API, um Daten aus dem Delegs-Projekt zu bekommen, und eine mit GWT-RPC realisierte Kommunikation, um die anwendungsinternen Nachrichten zu übermitteln.

#### 4.3.1 DELEGS-API

Die Delegs-API ist eine auf JSON<sup>6</sup> basierende REST-API<sup>7</sup>. An dieser API können drei verschiedene Anfragen gestellt werden:

- I. Präfix-Suche
- II. GebärdenSchrift Bilder-Service
- III. GebärdenSchrift-Suche

Über die Präfix-Suche kann anhand eines Wortpräfixes eine Liste mit möglichen kompletten Wörtern ausgegeben werden, zu welchen GebärdenSchrift existiert. Die untenstehende URL ist ein Beispielaufruf mit dem Präfix „ba“

<http://www.delegs.com/delegseditor/signwritingeditor/signitems?prefix=bad&callback=test>

---

<sup>6</sup> JavaScript Object Notation

<sup>7</sup> Representational State Transfer

Die Wort-Suche gibt dem Anfrager anhand eines Wortes alle GebärdenSchrift-IDs zurück, welche dieses Wort repräsentieren. Die untenstehende URL ist ein Beispielaufruf mit dem Wort „Bad“

```
http://www.delegs.com/delegseditor/signwritingeditor/signitems?word=BAD&call  
back=maybe
```

Diese GebärdenSchrift-IDs können dann an den GebärdenSchrift-Bilder-Service übergeben werden, der zu einer ID das GebärdenSchrift-Bild zurückgibt. Die untenstehende URL und das Bild sind der Aufruf nach einem GebärdenSchrift-Bild für „Bad“ wie in Abbildung 10 zu sehen.

```
http://www.delegs.com/delegseditor/signwritingeditor/signimages?upperId=1602&lowerId=Bad
```

#### 4.3.1.1 JSONP & SoP

Um die JSON API zuverlässig in einer Webanwendung zu nutzen, welche nicht von der gleichen Domain kommt wie die API, ist es notwendig die Same Origin Policy (SoP) modernen Browser zu beachten. Die SoP verhindert, dass man aus JavaScript neues JavaScript nachlädt, welches nicht von der exakt gleichen Domain wie der derzeitige Webkontext kommt.

Um trotzdem eine API zu benutzen, die nicht den SoP Kriterien entspricht, muss die API callbacks unterstützen und der Aufruf selber erfolgt über ein HTML-Script Element, welches per JavaScript in die Webseite eingefügt wird. In GWT gibt es für diese Vorgehensweise den `JsonpRequestBuilder`. Dieser abstrahiert das Vorgehen so, dass es eine klare Schnittstelle gibt, um diese API Anfragen durchzuführen. Wie am JSONp-Quellcode zu sehen, ist es nur notwendig, dem `JsonpRequestBuilder` eine URL und einen Callback zu übergeben, und die API wird aufgerufen.

```

public void setContent(String word) {
String url = "http://www.delegs.com/delegseditor/signwritingeditor/signitems?word=";
url = url + word;
JsonpRequestBuilder jsonp = new JsonpRequestBuilder();
jsonp.requestObject(url, new AsyncCallback<JSSign>() {
    @Override
    public void onFailure(Throwable caught) {
Window.alert(caught.getMessage()); }
    @Override
    public void onSuccess(JSSign result) {
position = 0;
JsArray<JavaScriptObject> signs = result.getSigns();
List<Sign> newContent = new LinkedList<Sign>();
for (int i = 0; i < signs.length(); i++) {
    Sign contentSign = new Sign(signs.get(i));
    newContent.add(contentSign);}
content = newContent;
setInitialContent(newContent);
}});}

```

Der beim JSONp-Quellcode am Callback zu sehende Typ JSSign ist ein JavaScript Overlay. Dieser macht das Verarbeiten der JavaScript-Objekte einfacher, da die Objekte vom JsonpRequestBuilder gleich in den Overlay-Typ überführt werden. Daher können die Overlay-Objekte direkt im Callback-Code genutzt werden. Wie am Overlay-Quellcode zu sehen, müssen die Daten mit nativem JavaScript aus dem JSON geladen werden.

```

class JSSign extends JavaScriptObject {
protected JSSign() {
}
public final native JsArray<JavaScriptObject> getSigns() /*-
return this;
*/-; }}

```

#### 4.3.2 GWT-RPC

Die Gwt-RPC ist die Methode, welche Delegs-Voting nutzt, um mit dem Server zu kommunizieren (vgl. [GOOG12A]). Diese RPC Methode wird von GWT bereitgestellt und benötigt einen Java-Servlet-Server auf der Serverseite. Die RPC Aufrufe sind komplett asynchron und basieren intern auf XMLHttpRequests (vgl. [GOOG12A]). Durch die Asynchronität ist die Benutzung des RPC anders als etwa bei klassischen RPC Verfahren wie RMI. Auf Clientseite übergibt man jedem RPC Aufruf ein Callback-Objekt, welches zwei Methoden implementiert: *onSucess* und *onFailure*



(vgl. [GOOG12A]). Wie am Beispiel zu sehen, wird ein vollwertiges Java-Objekt in diesen Methoden zurückgegeben.

```
new AsyncCallback<AuthTokenCreate>() {  
    @Override  
    public void onSuccess(AuthTokenCreate result) {  
        [...]  
    }  
}
```

GWT serialisiert und deserialisiert automatisch Java-Objekte bei der Client-Server-Kommunikation (vgl. [GOOG12A]).

Für jeden RPC-Service existieren zwei Interfaces und eine Implementation. Es gibt einen Service und ein Interface, in dem normale Java-Methoden definiert werden, mit Rückgabewerten, und ein Interface in dem die asynchronen Methoden definiert werden, wobei jeder Methode ein Callback-Objekt übergeben wird. GWT sorgt dann automatisch dafür, dass die Aufrufe, welche alle an die asynchronen Methoden gehen, an die normalen Methoden weitergeleitet werden und dass die Ergebnisse in den Callback-Objekten verpackt werden und zurück an den Client gelangen (vgl. [GOOG12A]).

#### 4.4 BENUTZERKONTEN & AUTHENTIFIKATION

Delegs-Voting nutzt Benutzerkonten um die einzelnen Bewertungen einem Nutzer zuzuordnen und um zu verhindern, dass jemand einfach mehrmals für die gleiche GebärdenSchrift stimmen kann. Die Nutzer können sich über die Webanwendung selber ein Benutzerkonto erstellen, indem sie einen Benutzernamen, ein Passwort und eine E-Mail Adresse angeben. Diese Daten werden per GWT-RPC an den Server übertragen, welcher, sollten die angegebenen Daten den Kriterien wie einzigartiger Benutzername, einzigartige E-Mail Adresse genügen, daraufhin einen AuthToken erstellt. Der AuthToken ist grundlegend ein kryptographischer Hash über den Benutzernamen und eine Zufallszahl. Beim Erstellen des AuthTokens wird dieser zusammen mit dem Erstellungszeitpunkt in die Datenbank geschrieben.

Bei jeder Anfrage über die RPC-Services, welche erfordert, dass der Nutzer eingeloggt ist, muss der Client diesen AuthToken mit übergeben werden, um zu beweisen, dass er authentifiziert ist.

#### 4.4.1 SICHERHEIT

Um sich anzumelden benötigt jeder Nutzer seinen Benutzernamen und sein Passwort. Delegs-Voting speichert dazu den Benutzernamen und die E-Mail-Adresse im Klartext ab, speichert allerdings vom Passwort nur den gesalzenen BCrypt-Hash, wie er von JBCrypt erzeugt wird ab. (vgl. [MILL10])

JBCrypt ist eine Implementation des BCrypt Algorithmus in Java. BCrypt basiert auf dem Key-erstellungsverfahren von Blowfish (vgl. [MILL10]). BCrypt wurde dabei so designt, dass es möglichst schwierig ist, den Algorithmus effizient auf Grafikkarten oder in FPGAs zu implementieren (vgl. [MILL10]). Grafikkarten die bei SHA-1 mehrere hundert Millionen Hashes pro Sekunde probieren können, kommen bei BCrypt oft nur auf einige hundert Hashes pro Sekunde. Da die BCrypt API es auch sehr schwer macht, Passwörter ohne Salt zu hashen, ist es auch nicht sinnvoll, Rainbowtables für BCrypt zu erstellen.

Durch BCrypt sollte es bei Verlust der Datenbank nicht möglich sein, selbst schwache Passwörter zu bruteforcen.

#### 4.5 PERSISTENZ

Delegs-Voting nutzt als persistenten Datenspeicher MySQL. Auf MySQL wird über jOOQ und JDBC zugegriffen wird (siehe auch 4.6.2 & 4.6.3). Dabei werden die Daten in einem relationalen Datenmodell gespeichert, welches auf OLTP ausgelegt ist. Das Datenbankschema besteht aus den Tabellen VOTES, LOGGED\_IN\_USER und USER, wie in Abbildung 7 zu sehen ist.

Die Tabelle VOTES dient dabei dem Speichern von Abstimmungsdaten der Anwendung. Jeder USER kann einmal pro GebärdenSchrift-Symbol abstimmen, was durch ein Unique-Constrain auf den Feldern SIGNID und USERID modelliert ist, wobei die USERID einen Nutzer referenziert. Auch wird bei jedem Abstimmen in den Feldern Krit1-4 gespeichert, wie der Nutzer dieses GebärdenSchrift-Symbol bewertet hat.

In der Tabelle LOGGED\_IN\_USER werden die AuthTokens (siehe 4.4) und das Datum des Anmeldens zu jedem Nutzer gespeichert.

In der Tabelle USER werden die Nutzerdaten gespeichert. Diese bestehen aus dem Nutzernamen, dem Hash des Nutzerpassworts und der E-Mailadresse des Nutzers. Dabei kann es jeden Nutzernamen und jede E-Mailadresse nur einmal geben, was durch ein Unique-Constrain jeweils auf diese Felder modelliert ist.

## 4.6 EINGESETZTE TECHNOLOGIEN

### 4.6.1 GOOGLE WEB TOOLKIT

Das Google Web Toolkit (GWT) ist ein Open-Source-Werkzeug zum Erstellen von Webanwendungen mit Java. GWT setzt dabei auf einen Java zu JavaScript Cross-Compiler. Dieser wandelt den Java-Code in JavaScript um, welches dann im Browser ausgeführt wird. (vgl. [GOOG12B])

Neben dem Java zu JavaScript Compiler enthält GWT noch eine mächtige Bibliothek, welche das Entwickeln für den Browser einfacher macht. So abstrahiert GWT Browserfeatures wie LocalStorage oder Canvas. Des Weiteren stellt GWT ClientBundles bereit, welche dazu dienen, effizient Ressourcen vom Server auf den Client zu übertragen (vgl. [GOOG12C]). GWT hat besondere ClientBundles unter anderem für Bilder und CSS. Bei Bildern fügt GWT beim Kompilieren alle Bilder in einem großen Bild zusammen und spart so Round-Trips und erreicht bei vielen ähnlichen Bildern auch eine bessere Komprimierung dieser. Ein weiterer Vorteil der Bundles ist, dass diese auch immer eine Java-Schnittstelle bereitstellen, so dass es einfach ist, die Ressourcen im Quellcode zu nutzen.

Ein weiteres Feature von GWT ist ein pseudo WYSIWYG-UI-Designer, das zum Erstellen von Benutzerinterfaces für eine GWT-Anwendung genutzt werden kann. Mit diesem UI-Binder genannten UI-Designer kann entweder im WYSIWYG-Editor das Benutzerinterface zusammengeklickt werden oder es kann in XML definiert werden (vgl. [GOOG12D]). Der XML-Code und der WYSIWYG-Editor werden synchron gehalten, so dass parallel in beiden gearbeitet werden kann, je nachdem, was angemessen ist (vgl. [GOOG12D]). Der UI-Binder unterstützt auch die ClientBundles für Ressourcen und erlaubt es auch direkt im XML das CSS der UI-Komponenten zu definieren (vgl. [GOOG12D]).

Zu jedem im UI-Binder definierten UI gibt es eine Begleitklasse, so dass es einfach ist, das MVC-Muster umzusetzen. Die UI-Elemente werden in dieser Begleitklasse automatisch instanziiert und können dann aus dem Controller heraus manipuliert werden (vgl. [GOOG12D]).

#### 4.6.2 JOOQ – JAVA OBJECT ORIENTED QUERY

jOOQ ist eine objektorientierte Schnittstelle zu einer der vielen unterstützten SQL – Datenbanken. jOOQ ist dabei kein Object-Relational-Mapper (vgl. [EDER12]). jOOQ ist eher mit einem Embedded-SQL zu vergleichen und ist also mehr eine DSL für Java, die das Arbeiten mit SQL-Datenbanken einfacher machen will (vgl. [EDER12]). Technisch baut jOOQ auf JDBC auf und hat also keine der Optimierungen, die mit Embedded-SQL in Verbindung gebracht werden.

Das Vorgehen bei jOOQ ist, dass ein Werkzeug automatisch das Datenbankschema analysiert und daraus Klassen erstellt, welche dann genutzt werden können, um auf die Datenbank zuzugreifen. jOOQ nutzt dabei das „fluid“ Design, um SQL-Anweisung natürlich in Java umzusetzen, was am jOOQ-Quellcode deutlich wird. Da bei diesen Methodenaufrufen Objekte zurückkommen, welche nur die Aufrufe implementieren, welche auch an dieser Stelle im SQL sinnvoll sind, ist auch die Code-Vervollständigung der genutzten IDE eine große Hilfe. (vgl. [EDER12])

```
dbVoting
.select(USER.ID, USER.USERNAME)
.from(USER, LOGGED_IN_USER)
.where(LOGGED_IN_USER.AUTHTOKEN.equal(authToken
.getAuthToken()))
.and(LOGGED_IN_USER.USERID.equal(USER.ID)).fetch();
```

#### 4.6.3 MYSQL

Delegs-Voting nutzt MySQL als persistenten Datenspeicher. MySQL ist ein Open-Source Datenbanksystem das von Oracle entwickelt wird. MySQL versucht den aktuellen SQL-Standard zu unterstützen. MySQL implementiert aber auch einige Erweiterungen zu diesem, wenn entsprechender Bedarf vorhanden ist. (vgl. [MYSQ10])

# 5 FAZIT & AUSBLICK

---

## 5.1 ZUSAMMENFASSUNG

Diese Arbeit stellt die verschiedenen Aspekte eines Bewertungssystems dar und führt Kriterien, die bei der Erstellung eines Bewertungssystems beachtet werden sollten, an. Die Möglichkeiten des Designs der grundlegenden Bewertungsmodelle und -systeme werden vorgestellt. Weiterhin wird erörtert, wie die Nutzer eines Bewertungssystems mit diesem interagieren und wie sie zur Nutzung des Systems motiviert werden können. Auf diesen Grundlagen wurde eine Beispielanwendung für die Bewertung von GebärdenSchrift-Symbolen entwickelt und erläutert.

## 5.2 FAZIT

Die praktische Erprobung der Anwendung Delegs-Voting wurde durch den GebärdenSchrift-Experten vorgenommen. Dabei bestätigte sich die ursprüngliche Idee, dass Bewertungssysteme für GebärdenSchrift im Kontext des Delegs-Projekts gut geeignet sind, um die Qualitätssicherung und Optimierung der Datenbasis des Projekts vorzunehmen. Dies ist ein Indikator für die Anwendbarkeit des erstellten Kriterienkatalogs.

## 5.3 AUSBLICK

Bewertungssysteme werden von vielen sozialen Internetseiten eingesetzt, um die Inhalte für die Nutzer aufzubereiten. Im Kontext des Delegs-Projekts ist der Einsatz von Bewertungssystemen auch für andere Teilbereiche neben der Bewertung von GebärdenSchrift-System sinnvoll. Insbesondere die Bewertung von Dokumenten oder Unterrichtseinheiten kann angedacht werden. Auch könnte das Bewertungssystem für die GebärdenSchrift erweitert werden, um bessere Ergebnisse zu erzeugen. Zum Beispiel, indem das System um soziale Features oder um ein analysebasiertes Bewertungssystem erweitert wird. Zusätzlich könnten die Bewertungsmechaniken noch direkter in den Delegs-Editor integriert werden, so dass die Nutzer bei ihren normalen Interaktionen mit dem Editor GebärdenSchrift bewerten könnten.

# LITERATURVERZEICHNIS

---

- [BREU11A] M. Breuer, "Was ist Gamification?," *intelligent gamification*, 11-May-2011. [Online]. Available: <http://intelligent-gamification.de/2011/05/11/was-ist-gamification/>. [Accessed: 24-Aug-2012].
- [BREU11B] M. Breuer, "Gamification ergibt schlechte Software?," *intelligent gamification*, 27-Dec-2011. [Online]. Available: <http://intelligent-gamification.de/2011/12/27/gamification-ergibt-schlechte-software/>. [Accessed: 30-Aug-2012].
- [DOVE08] D. Dover, "Reddit, Stumbleupon, Del.icio.us and Hacker News Algorithms Exposed!," *Seomoz*, 02-Jun-2008. [Online]. Available: <http://www.seomoz.org/blog/reddit-stumbleupon-delicious-and-hacker-news-algorithms-exposed>. [Accessed: 11-Sep-2012].
- [EDER12] L. Eder, "jOOQ : A peace treaty between SQL and Java," 2012. [Online]. Available: <http://www.jooq.org/>. [Accessed: 12-Sep-2012].
- [FAGL10] F. R. Farmer and B. Glass, *Building Web reputation systems*. Sebastopol, CA: O'Reilly, 2010.
- [FEIN12] C. Fein, "Gamification: Risiken und Chancen durch Spielelemente in Businessprozessen," *Boxedpages*, 27-Aug-2012. [Online]. Available: <http://blog.boxedpages.net/2012/08/27/gamification-risiken-und-chancen/>. [Accessed: 30-Aug-2012].
- [GLAS08] B. GLASS, "Designing your reputation system," *Presentation at IASummit, Miami FL*, 2008.
- [GOOG12A] Google, "Making Remote Procedure Calls," 27-Jun-2012. [Online]. Available: <https://developers.google.com/web-toolkit/doc/2.1/tutorial/RPC>. [Accessed: 18-Sep-2012].
- [GOOG12B] Google, "Google Web Toolkit Overview," 27-Jul-2012. [Online]. Available: <https://developers.google.com/web-toolkit/overview>. [Accessed: 12-Sep-2012].
- [GOOG12C] Google, "Developer's Guide - Client Bundle," 27-Jun-2012. [Online]. Available: <https://developers.google.com/web-toolkit/doc/latest/DevGuideClientBundle>. [Accessed: 18-Sep-2012].
- [GOOG12D] Google, "Declarative Layout with UiBinder," 27-Jun-2012. [Online]. Available: <https://developers.google.com/web-toolkit/doc/latest/DevGuideUiBinder>. [Accessed: 18-Sep-2012].

- [KETR08] D. Ketrainis, "per-reddit automated spam filters," *blog.reddit*, 10-Jun-2008. [Online]. Available: <http://blog.reddit.com/2008/06/per-reddit-automated-spam-filters.html>. [Accessed: 29-Aug-2012].
- [LIFS08] Y. Lifshits, "The Architecture of the Web," 2008.
- [MILL09] E. Miller, "How Not To Sort By Average Rating," *Evanmiller.org*, 09-Feb-2009. [Online]. Available: <http://www.evanmiller.org/how-not-to-sort-by-average-rating.html>. [Accessed: 10-Sep-2012].
- [MILL10] D. Miller, "jBCrypt," 01-Feb-2010. [Online]. Available: <http://www.mindrot.org/projects/jBCrypt/>. [Accessed: 11-Sep-2012].
- [MORG12] J. Morgan, "The Risks of 'Gamification' for the Enterprise," 29-Mar-2012. [Online]. Available: <http://www.jmorganmarketing.com/risks-gamification-enterprise/>. [Accessed: 30-Aug-2012].
- [MUNR09] R. Munroe, "reddit's new comment sorting system," *blog.reddit.com*, 15-Oct-2009. [Online]. Available: <http://blog.reddit.com/2009/10/reddits-new-comment-sorting-system.html>. [Accessed: 10-Sep-2012].
- [MYSQ10] MySQL-Doc, "1.8. MySQL Standards Compliance," 2010. [Online]. Available: <http://dev.mysql.com/doc/refman/5.5/en/compatibility.html>. [Accessed: 25-Aug-2012].
- [RADO11] J. Radoff, "Game Player Motivations Game On," 19-May-2011. [Online]. Available: <http://radoff.com/blog/2011/05/19/game-player-motivations/>. [Accessed: 25-Aug-2012].
- [SPLA12] spladug, "reddit LICENSE," 19-Jun-2012. [Online]. Available: <https://github.com/reddit/reddit/blob/master/LICENSE>. [Accessed: 01-Sep-2012].
- [WIKI12] Wikipedia, "Sockpuppet (Internet)," *Wikipedia*, 2012. [Online]. Available: [http://en.wikipedia.org/w/index.php?title=Sockpuppet\\_\(Internet\)&oldid=507432441](http://en.wikipedia.org/w/index.php?title=Sockpuppet_(Internet)&oldid=507432441). [Accessed: 28-Aug-2012].
- [WÖHR12] S. Wöhrmann, "Bad," 21-Sep-2012. [Online]. Available: <http://delegs.de/delegseditor/signwritingeditor/signimages?upperId=1602&lowerId=Bad>. [Accessed: 21-Sep-2012].

# ABBILDUNGSVERZEICHNIS

---

Abbildung 1 Schichten und Schnitte bei Delegs-Voting.....	45
Abbildung 2 Bewertung bei YouTube .....	58
Abbildung 3 Sterneratings [FAGL10, p.70] .....	58
Abbildung 4 Rezension eines Nutzers [FAGL10, p.71] .....	58
Abbildung 5 Robustes Karma-Modell [FAGL10, p.73].....	59
Abbildung 6 Motivationsmatrix nach Radoff [RADO11].....	59
Abbildung 7 Datenbankschema für Delegs-Voting.....	60
Abbildung 8 Beliebt Bewertung Reddit [DOVE08].....	61
Abbildung 9 Wilson Intervall [MILL09].....	61
Abbildung 10 GebärdenSchrift für Bad [WÖHR12] .....	62
Abbildung 11 Delegs-Voting Mockup .....	62
Abbildung 12 Delegs Voting Mock mit Reviews .....	62
Abbildung 13 Delegs-Voting fertige Webanwendung.....	63



# TABELLENVERZEICHNIS

---

Tabelle 1 Motivation in Foren [GLAS08] .....	64
Tabelle 2 Aktionen in Foren nach Bryce Glass [GLAS08] .....	65

# ANHANG

---

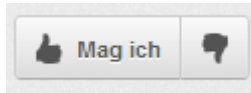


Abbildung 2 Bewertung bei YouTube



Abbildung 3 Sterneratings [FAGL10, p.70]

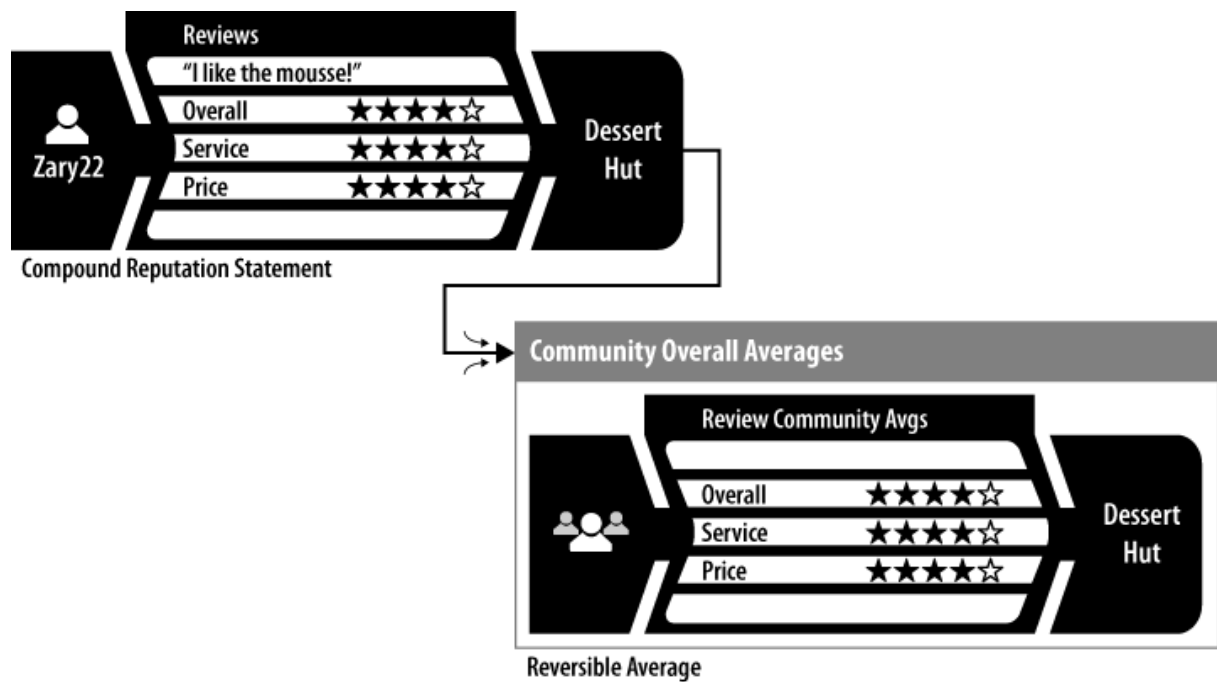


Abbildung 4 Rezension eines Nutzers [FAGL10, p.71]

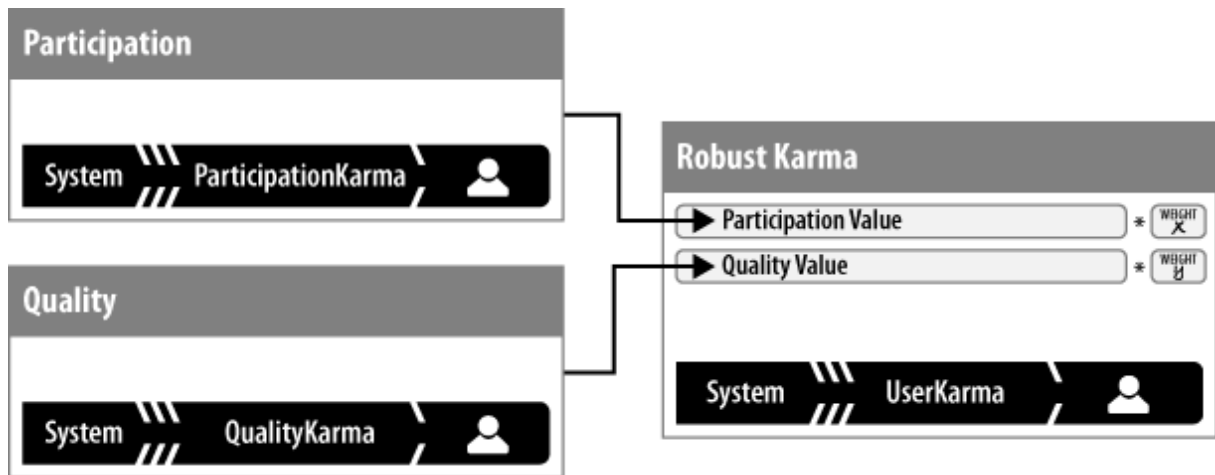


Abbildung 5 Robustes Karma-Modell [FAGL10, p.73]

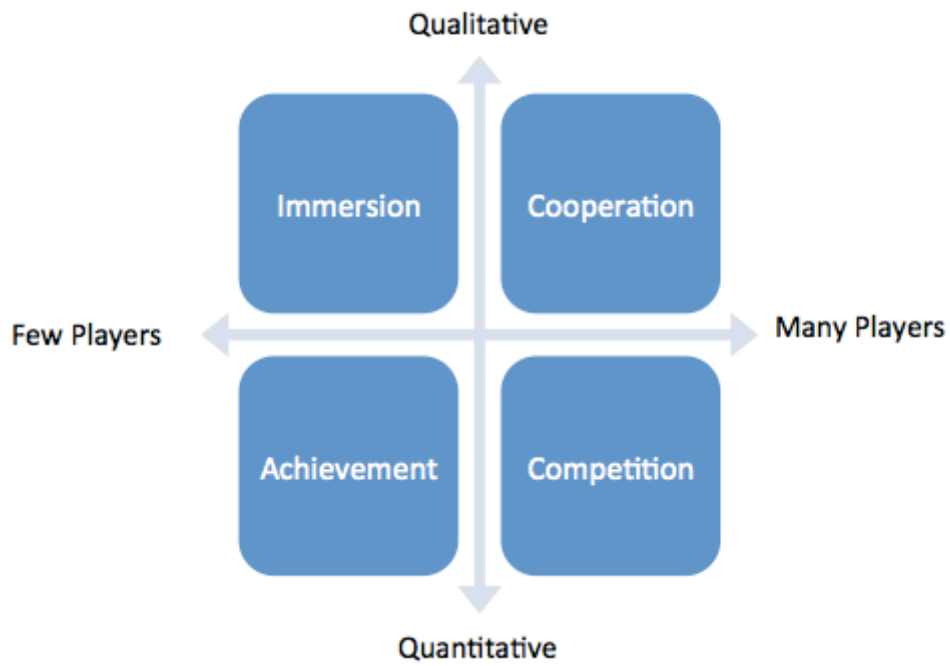


Abbildung 6 Motivationsmatrix nach Radoff [RADO11]

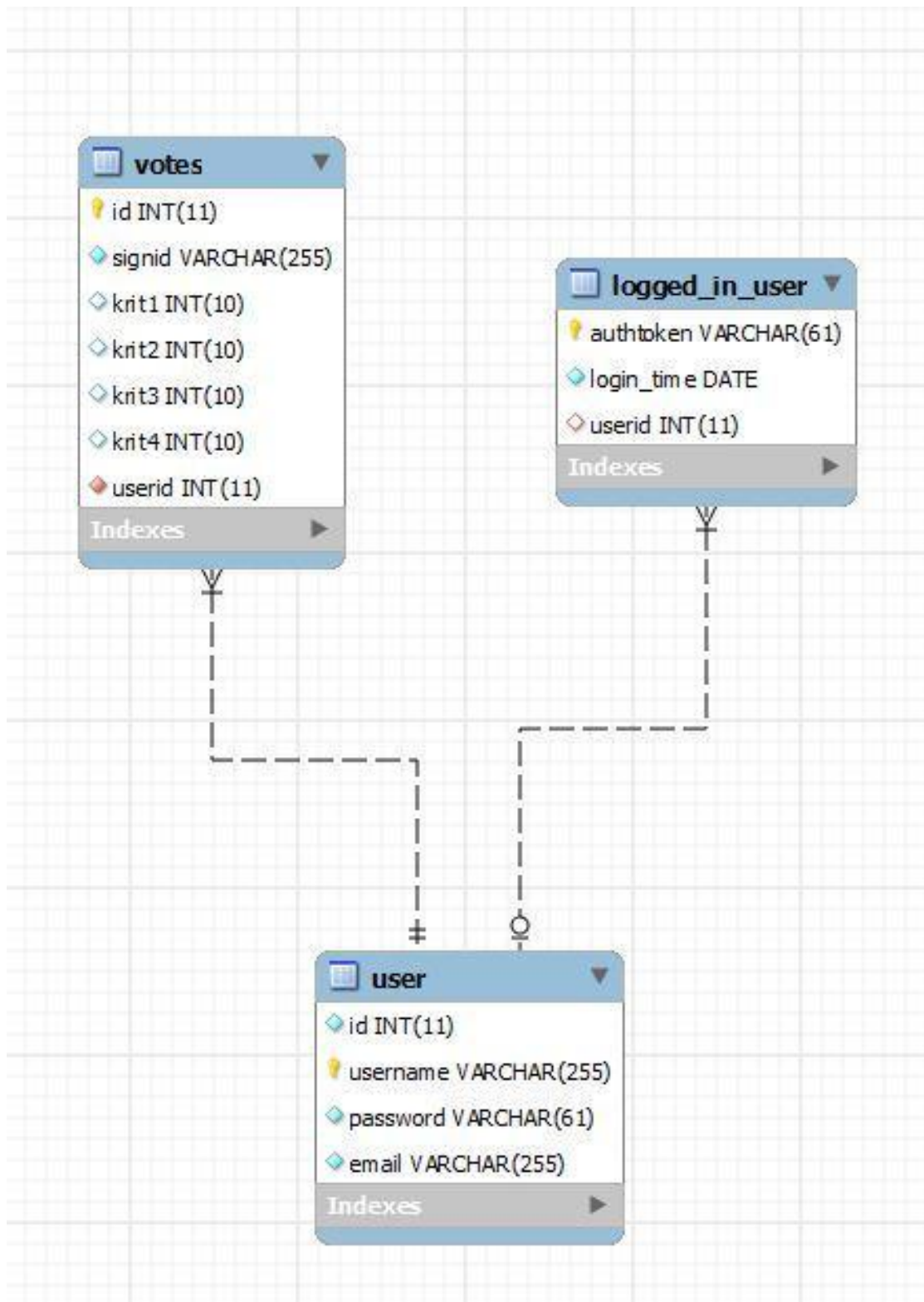


Abbildung 7 Datenbankschema für Delegs-Voting

Given the time the entry was posted  $A$  and the time of 7:46:43 a.m. December 8, 2005  $B$ , we have  $t_s$  as their difference in seconds

$$t_s = A - B$$

and  $x$  as the difference between the number of up votes  $U$  and the number of down votes  $D$

$$x = U - D$$

where  $y \in \{-1, 0, 1\}$

$$y = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}$$

and  $z$  as the maximal value, of the absolute value of  $x$  and 1

$$z = \begin{cases} |x| & \text{if } |x| \geq 1 \\ 1 & \text{if } |x| < 1 \end{cases}$$

we have the rating as a function  $f(t_s, y, z)$

$$f(t_s, y, z) = \log_{10} z + \frac{y t_s}{45000}$$

**Abbildung 8** Beliebt Bewertung Reddit [DOVE08]

$$\left( \hat{p} + \frac{z_{\alpha/2}^2}{2n} \pm z_{\alpha/2} \sqrt{[\hat{p}(1 - \hat{p}) + z_{\alpha/2}^2/4n]/n} \right) / (1 + z_{\alpha/2}^2/n).$$

**Abbildung 9** Wilson Intervall [MILL09]

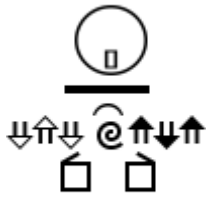


Abbildung 10 GebärdenSchrift für Bad [WÖHR12]

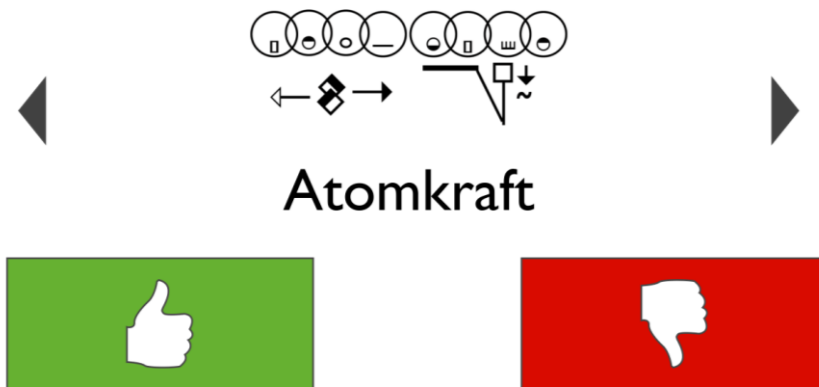


Abbildung 11 Delegs-Voting Mockup

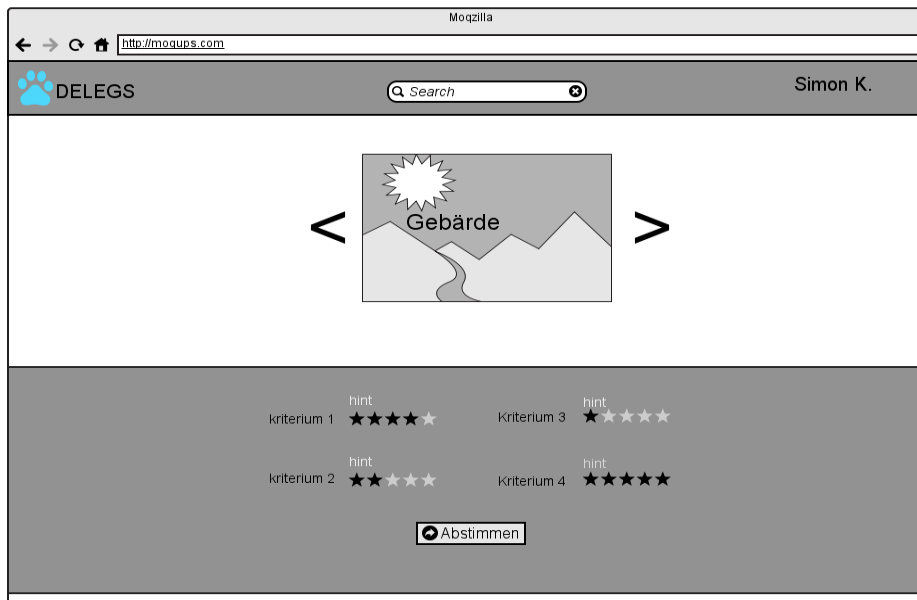


Abbildung 12 Delegs Voting Mock mit Reviews



Abbildung 13 Delegs-Voting fertige Webanwendung

Tabelle 1 Motivation in Foren [GLAS08]

<b>Eigeninteresse</b>	<b>Interesse in Andere</b>	<b>Interesse in das Bewertungsobjekt</b>
Eine Stimme haben	Anderen helfen	Eine Leere füllen
Die eigenen Fertigkeiten verbessern	Eine einzigartige Perspektive teilen	"Etwas Klarstellen" (Eine abweichende Meinung)
Die eigene Geschichte erzählen	Sich mit anderen verbunden fühlen	Die Firma belohnen
Sich selbst positiv darstellen	Anderer unterhalten	Die Firma bestrafen
Seine Meinung äußern		



Tabelle 2 Aktionen in Foren nach Bryce Glass [GLAS08]

<b>Aktion</b>	<b>Beeinflusst primär</b>	<b>Bedeutung</b>
Starte eine neue Diskussion	Diskussion	Aktivität
Beantworte einen Post	Post	Interesse (Implizit)
	Person	Aktivität
Bewerte einen Post	Post	Qualität (Explizit)
Einem Freund mitteilen	Post	Qualität (Implizit)
Lese einen Post	Post	Interesse? (Implizit)
Klick auf „Mehr“ um einen Post auszuklappen	Post	Interesse (Implizit)
Melden	Post	Qualität (Explizit)
Nutzer blockieren	Person	Qualität (Implizit)
Einen Post zitieren	Post	Interesse (Implizit)