

# Badges als Bestandteil des Reputationsmanagements im Universitätsstudium

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur/in**

im Rahmen des Studiums

**Medieninformatik**

eingereicht von

**Immanuel Bauer**

Matrikelnummer 0626117

an der  
Fakultät für Informatik der Technischen Universität Wien

Betreuung  
Betreuer: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Peter Purgathofer  
Mitwirkung: Dipl.-Ing. Naemi Luckner

Wien,

\_\_\_\_\_  
(Unterschrift Verfasser/in)

\_\_\_\_\_  
(Unterschrift Betreuer/in)

Immanuel Bauer  
Belvederegasse 6/3/71, 1040 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien,

## **Kurzfassung**

In dieser Arbeit wird exploriert, wie sich, im Internet etablierte Reputationsmechanismen, sowie Gamificationkonzepte, in einen Universitätskontext der TU Wien einbetten lassen. Um so etwa die Lehre zu verbessern, eine granulärere Beurteilung der Leistungen von Studierenden zu ermöglichen oder etwa einen zusätzlichen Anreiz zu außerordentlichen Leistungen für Studierende zu bieten. Ein Gamificationkonzept, das primär betrachtet wird, sind sogenannte Badges. Diese sind digitale Abzeichen, welche für die Erfüllung gewisser Aufgaben oder Erreichen bestimmter Ziele vergeben werden. Es wird auch diskutiert, inwieweit bereits vorhandene Reputation von externen Quellen eingebracht werden kann und wo Reputationsmechanismen in bestehenden Systemen wie z.B. TUWEL oder TISS eingesetzt werden können. Mit Hilfe von Literaturrecherche und Interviews wird der Status Quo in der Forschung und im Einsatz befindliche Methoden sowie Ansprüche an ein Reputationssystem erhoben. Darauf basierend entstanden ein Prototyp für ein Badgevergabesystem und ein konzeptuelles zentrales System für die TU Wien um Reputation zu erfassen und präsentieren.

## **Abstract**

This paper explores how reputation methods established in the internet as well as gamification concepts could be used in the university context of the TU Vienna in order to improve teaching, enable a more granular rating of student's performance or motivate students in general to show exceptional results. A gamification concept this work will focus on are badges. These are digital awards that can be earned by completing certain tasks. In this paper will be discussed as well how already existing reputation information could be aggregated and how already existing systems like TUWEL or TISS could be augmented with reputation methods. In order to define current reputation methods and requirements to a reputation system literature research and interviews were conducted. Based on the findings a prototype for a platform to award and present badges as well as a conceptual TU Vienna centric reputation aggregation system were designed.

# Inhalt

<b>Einleitung</b>	<b>5</b>
<b>Fragestellung</b>	<b>6</b>
<b>Definitionen</b>	<b>6</b>
<b>Motivation</b>	<b>10</b>
<b>Reputationsysteme</b>	<b>12</b>
Amazon	13
eBay	16
Stackoverflow	18
Facebook	21
Reddit	23
YouTube	25
Twitter	27
Klout	28
Coderwall	30
Behance	32
Newgrounds	34
Yahoo Design Patterns	36
Reputation Modelle	41
<b>Gamification</b>	<b>45</b>
<b>Interviews</b>	<b>48</b>
<b>Badges Prototyp</b>	<b>53</b>
Was ist ein Badge?	53
Ähnliche Projekte	54
Design	56
Badge Wert	61
Technische Implementation	62
Evaluation	77
Mögliche Erweiterungen	79
Zusammenfassend	82
<b>TU Wien Reputation</b>	<b>83</b>
Ziele & Konzepte	83
TURep	85
Umgang mit externen Plattformen	106
Anonymität & Privatsphäre	110
Opt-Out, Datenexport & Datenschutz	111
Über den Tellerrand	113
<b>Conclusio &amp; Ausblick</b>	<b>114</b>
<b>Referenzen</b>	<b>116</b>

## Einleitung

Im vergangenen 20. Jahrhundert wurde durch die immer stärkere Verbreitung und Kommerzialisierung des Internets, sowie durch die Bildung erster größerer virtueller Communities das Bedürfnis nach Lösungen immer größer, wie man in dieser virtuellen Welt Vertrauen zu anderen Teilnehmern aufbauen kann. Sei es nun, um sich über ein gewisses Thema auszutauschen oder um Handel mit Produkten und Dienstleistungen zu treiben. Systeme die als Lösung für diese Problemstellung entstanden sind, kann man umfassend als Reputationsysteme bezeichnen.

Da virtuelles Vertrauen („Virtual Trust“, u.a. [Laat, 2005]) ursprünglich verstärkt in kommerziellen Anwendungen von Bedeutung ist und sich Reputation in diesem Kontext auch relativ einfach mathematisch modellieren lässt, beschäftigt sich die wissenschaftliche Literatur vorzugsweise mit diesem Aspekt der virtuellen Reputation. Durch das Entstehen des sogenannten Web 2.0, mit seinen, verstärkt durch Benutzer generierten Inhalten und den in letzter Zeit immer wichtiger werdenden, Sozialen Netzwerken, gibt es aber auch zunehmend Arbeiten, die sich mit diesem etwas sozialeren Aspekt der Reputation näher beschäftigen.

Eingehend werde ich einige wichtige Begriffe im Bezug auf Reputation und Badges definieren und anschließend eine Auswahl von Beispielen für Reputationsysteme in Webseiten anführen, um so einige der Elemente zu erläutern, die bei solchen Systemen zum Einsatz kommen können. Diese Liste von Beispielen werde, ich mit einer Klassifikation von Methoden und Best Practices, die bereits von anderen Autoren ermittelt wurden, abschließen und so einen Einstieg für die weiteren vertiefenden Kapitel liefern.

Anschließend werde ich noch etwas genauer auf die Thematik der Gamification eingehen, um so einige Grundlagen und Motivationen zur Verwendung von Badges zu vermitteln.

Nach diesen einleitenden Kapiteln werde ich mit einer Zusammenfassung mehrerer Interviews, zum Grundthema „Förderung von außergewöhnlichen Leistungen“, versu-

chen einen Überblick zu den Anforderungen eines Reputationssystems innerhalb der TU Wien zu geben.

Anschließend werde ich die Gestaltung und Implementation eines Prototypen zur Vergabe von Badges beschreiben, welcher basierend auf den Erkenntnissen aus den Interviews und eines Designworkshops gestaltet und umgesetzt wurde.

Im letzten Abschnitt dieser Arbeit, werde ich darauf eingehen wie ein komplettes Reputationssystem für die TU Wien aussehen könnte und wie man dieses in bereits vorhanden Plattformen einbinden könnte. Dazu werde ich vor allem auf die Gestaltung eingehen und auch noch ein paar Vorschläge zu Implementationsdetails anführen.

## **Fragestellung**

Im Rahmen dieser Arbeit möchte ich die Frage behandeln, wie man im Internet gebräuchliche Reputationmechanismen in einem Universitätskontext zur Anwendung bringen könnte. Ein besonderes Augenmerk soll auch auf Badges, eine Form von Reputation generierender Gamification, gelegt werden.

Wie können mögliche Anwendungsszenarien aussehen und welche Gefahren und Spannungsfelder würde eine Implementation dieser bergen.

## **Definitionen**

Einleitend möchte ich auf einige Grundbegriffe aus dieser Arbeit eingehen, die sich in der Literatur teilweise in ihrer Semantik unterscheiden oder bei denen sich noch keine allgemeingültige Definition durchgesetzt hat.

## **Vertrauen**

Der Begriff „Vertrauen“ beziehungsweise zu Englisch „Trust“, ist stark verknüpft mit dem der Reputation. Das kommt daher, dass das Ziel von Reputation und Reputationssystemen meist das Schaffen von Vertrauen zwischen den Teilnehmern eines spezifischen Kontext ist.

Kwan und Ramachandran verstehen in [Kwan & Ramachandran, 2009] die Reputation, als einen von drei Aspekten, neben Beziehung (Relationship) und Verfahren (Process), aus denen sich Vertrauen zusammensetzt. Abhängig von der Art des Netzwerks oder der Plattform können diese drei

Teilaspekte unterschiedlich stark ausgeprägt sein. Facebook hat seinen Schwerpunkt beispielsweise auf der Beziehung zwischen den Benutzern. eBay andererseits setzt bei seinen Ver-/Käufer Bewertungen verstärkt auf Reputation. Ein Vertrauen auf Basis des Verfahrens kann man beispielsweise bei Wikipedia finden. Dort vertraut man Inhalten bzw. den Autoren, da diese von unzähligen Benutzern jederzeit auf Richtigkeit überprüft und korrigiert werden können.

Vertrauen setzt sich, laut Kwan und Ramachandra, vor allem auch aus zwei Erwartungen zusammen, die eine Person einer anderen gegenüber hat. Die Erwartung, dass die andere Person gute Absichten hat sowie die Erwartung, dass die andere Person die Fähigkeiten hat, diese Absichten umzusetzen.

Zwei weitere wichtige Feststellung, die sie zu Vertrauen machen sind, dass Vertrauen nicht transitiv ist. Sprich wenn Person A einer Person B vertraut und B einer Person C, bedeutet das nicht, dass A auch C vertraut. Vertrauen ist auch stark Kontextabhängig. So traut A beispielsweise B zu, eine Software zu programmieren, aber nicht einen Tisch zu zimmern.

## **Reputation**

Unter dem Wort Reputation versteht man im Deutschen so wie im Englischen weitestgehend den Ruf oder Status einer Person oder Organisation. Reputation entstammt laut dem Duden von dem lateinischen Begriff *reputatio*, welcher so viel bedeutet wie „Erwägung“ oder „Berechnung“. Kurioserweise trifft die Bedeutung des lateinischen Ursprungswortes eher auf die Reputation im Kontext virtueller Welten zu, als der im alltäglichen Leben. In letzterem setzt sich die Reputation meist intuitiv und teils unterbewusst aus Erfahrungen, Emotionen und vor allem Meinungen aus unserem Umfeld zusammen. Im Gegensatz dazu ist die Reputation in einer virtuellen Welt primär eine Ansammlung von Daten und Informationen die einem abstrakten „Objekt“ zugeordnet sind. Diese muss daher auch nicht gezwungenermaßen immer dazu dienen, dass sich eine Person mit Hilfe dieser Informationen eine Meinung bilden kann. Die Reputation kann auch von internen Prozessen eines EDV System genutzt werden um etwa, wie von [Xiong

& Liu, 2004] beschrieben, in einem Peer-to-Peer Netzwerk die Vertrauenswürdigkeit einer potentiellen Gegenstellen zu ermitteln und mit dieser Information darüber entscheiden, ob eine Verbindung oder ein Datentransfer eingeleitet wird oder nicht.

Eine Reputation Information entsteht im einfachsten Fall dadurch, dass A eine Aussage über B trifft. [Farmer & Glass] bezeichnen dies als das „Minimum Reputation Statement“ und generalisieren diese Aussage wie folgt: „*a source makes a claim about a target*“ sprich eine Quelle behauptet etwas zu einem Ziel.

Diese Behauptung oder Aussage kann entweder implizit oder explizit erfolgen. Bei einer expliziten Aussagen trifft A eine direkte Aussage zu B, z.B. „A sagt, dass B gut ist“. Eine implizite Aussage wiederum ergibt sich indirekt durch das Verhalten von A gegenüber B z.B. „A kauft Produkt B statt Produkt C“.

Dellarocas stellt in [Dellarocas, 2010] fest, dass die Definition der Reputation noch relativ weich ist und oft vom Kontext der Anwendung abhängt. In Bezug auf Personen versteht er darunter vor allem die Summe der vergangenen Aktionen, die jemand innerhalb eines Kontextes, z.B. einer Internetcommunity, getätigt hat.

## **Karma**

[Farmer & Glass] sehen Reputationinformation zu einer Person als einen Sonderfall und unterscheiden zwischen der Reputation eines beliebigen „Objekt“ und der einer Person beziehungsweise eines Benutzers einer Plattform. Diese Reputation einer Person betiteln Sie als „Karma“. Der Begriff kommt aus dem Hinduismus und Buddhismus und geht auf das Konzept der Wiedergeburt zurück, in dem das Handeln im aktuellen Leben das Leben nach der Wiedergeburt beeinflusst. Somit wird Karma vor allem auch als eine auf das Benutzerverhalten bezogene Reputation verstanden und ist abhängig vom Kontext oder der Applikation. Das Karma ist auch unterschiedlich stark mit der Reputation von Objekten, die der Benutzer in die Plattform eingebracht hat, verbunden. So kann sich etwa eine

schlechte Bewertung auf ein Kommentar oder etwa auch die Reputation eines Objekts innerhalb eines digitalen Portfolios, sich auf das Karma des Autors auswirken.

## **Gamification**

Das Forschungsfeld der Gamification ist noch ein relativ junges und der Begriff selbst ist noch in einem stetigen Wandel. Einen ersten Versuch, eine Definition zu finden, hat [Deterding et al, 2011] unternommen. In diesem Paper wird zwischen zwei unterschiedlichen Konzepten unterschieden. Einerseits die Sichtweise, dass sich Gamification mit der gesellschaftlichen Anerkennung und Institutionalisierung von Videospielelementen und den Einfluss von Spielen und Spielelementen auf das alltägliche Leben beschäftigt. Die zweite und mittlerweile verbreiterte Sichtweise ist, dass es sich bei Gamification um das Anwenden von Game Design Prinzipien für das Design von Artefakten in einem nicht unbedingt mit Spielen in Verbindung stehenden Kontext. Dies hat vor allem zum Ziel, diese Artefakte motivierender und Freude bringender zu machen. Solche Artefakte müssen auch nicht gezwungenermaßen auf einen digitalen Kontext begrenzt sein.

Deterding et al betonen ebenfalls, dass Gamification und Serious Games zu trennen sind, da es sich bei letzteren um echte Spiele mit einem ernsteren Hintergrund handelt, bei ersterem aber lediglich um die Anwendung von Game Design Prinzipien. Das Endergebnis dieses Designprozesses muss aber nicht gezwungenermaßen ein Spiel sein. Schlussendlich sind die Grenzen laut Deterding et al hier aber fließend.

Grundsätzlich geht es bei Gamification laut Deterding et al um Designprinzipien und nicht um Technologie. Er hebt die folgenden Designaspekte hervor:

- Interface Design Patterns (Badges, Levels, Bestenlisten, ...)
- Game Design Patterns / Spielmechaniken
- Design Prinzipien & Heuristiken

- Konzeptuelle Modelle für Spielkomponenten (MDA (Mechanics-Dynamics-Aesthetics) Framework, Malone's challenge, curiosity, ...)
- Game Design Methoden (Playtesting, Playcentric design, value conscious game design, ...)

[Koch & Ott, 2012] definieren für den neuen englischen Begriff „Gamification“ den deutschen Ausdruck „Spielifizierung“. Im weiteren Verlauf dieser Arbeit, werde ich aber dennoch auf den englischen Begriff „Gamification“ zurückgreifen, da dieser auch im deutschen Sprachgebrauch weitaus geläufiger ist.

Koch & Ott grenzen Gamification zusätzlich noch von dem sogenannten „Edutainment“ ab. So werden Medieninhalte bezeichnet welche Bildung „Education“ und Unterhaltung „Entertainment“ verbinden. Im Vordergrund steht bei diesen aber, im Gegensatz zur weiter gefassten Gamification, explizit das Lernen.

## **Motivation**

Die Informatik versteht sich als ein multidisziplinäres Studium, was sich vor allem am großen Angebot der unterschiedlichen spezialisierten Studiengängen zeigt, welche die TU Wien innerhalb der Informatik anbietet. Aber auch andere Studienzweige werden zunehmend komplexer und multidisziplinärer. Diese Komplexität an Qualifikationen mit einem Katalog von Lehrveranstaltungen oder einer einzelnen Note abzubilden wird ebenfalls immer schwieriger. Somit liegt es meist an den Studierenden zu dokumentieren, mit welchen Themen sie sich befasst und welche Mehrleistungen sie erbracht haben. Sei es nun um dieses Know-How während und nach dem Studium gegenüber anderen Personen kommunizieren und nachweisen zu können oder einfach nur um den eigenen Studienfortschritt im Auge zu behalten.

An der TU Wien gibt es bereits Bestrebungen den Studierenden die Dokumentation von Projekten mit Hilfe eines digitalen Portfolios zu erleichtern. Dies deckt zwar bereits einen Teil dieser Problematik ab, ist aber nicht auf jede Form der Mehrleistung anwendbar, da es meist nur Sinn macht, abgeschlossene Projekte in so ein Portfolio aufzunehmen. Auch ist die schlussendliche Präsentation im

Portfolio von der Motivation und Zeit des Studierenden abhängig. Ein weiteres Problem bei digitalen Portfolios kann sein, dass Außenstehende sich schwer dabei tun oder nicht die nötige Qualifikation besitzen, um die Qualität eines Projektes richtig einzuschätzen.

Dieses Problem der Quantifizierung von Qualität wurde im Internet bereits gelöst und zwar mit der Hilfe von Bewertungen. Diese sind meistens in Form von fünf oder mehreren Sternen in unterschiedlichsten Online Communities wiederzufinden. Diese Form der Bewertung ist einer von mehreren Reputationmechanismen, die von Online Plattformen eingesetzt werden um virtuelles Vertrauen zwischen den Benutzern zu schaffen und es diesen zu erleichtern interessante und vertrauenswürdige Inhalte, Produkte, Dienstleistungen oder Benutzer zu finden.

Alle Studierenden und Lehrenden der TU Wien bilden im Grunde auch eine Form der Community die zu großen Teilen auch bereits virtuell in Foren und Blogs vernetzt ist. Was wäre, wenn man nun die für Online Communities gängigen Reputationmechanismen im Universitätskontext zum Einsatz bringt?

In dieser Arbeit beschäftige ich mich mit dieser Fragestellung und werde einen besonderen Fokus auf den Einsatz von Badges, eine Form von Gamification, legen. Die Implikationen und Möglichkeiten beim Einsatz solcher Reputationmechanismen sind mannigfaltig und komplex. So könnte es etwa zur Motivation bis hin zur Dokumentation von Projekten, Interessen und des Lernfortschritts eingesetzt werden. Mit der Erfassung der nötigen Informationen und der potentiellen Umfassendheit eines dafür benötigten Systems ergeben sich aber auch Probleme in Bezug auf Privatsphäre, Datenschutz und Betrugsversuche.

## **Reputationsysteme**

Um ein Gefühl für unterschiedliche Konzepte und Methoden zu vermitteln, die bei der Gestaltung von Reputationsystemen zum Einsatz kommen, werde ich in diesem Kapitel einige bekannte Webseiten vorstellen und werde darlegen wie diese Reputation zum Einsatz bringen.

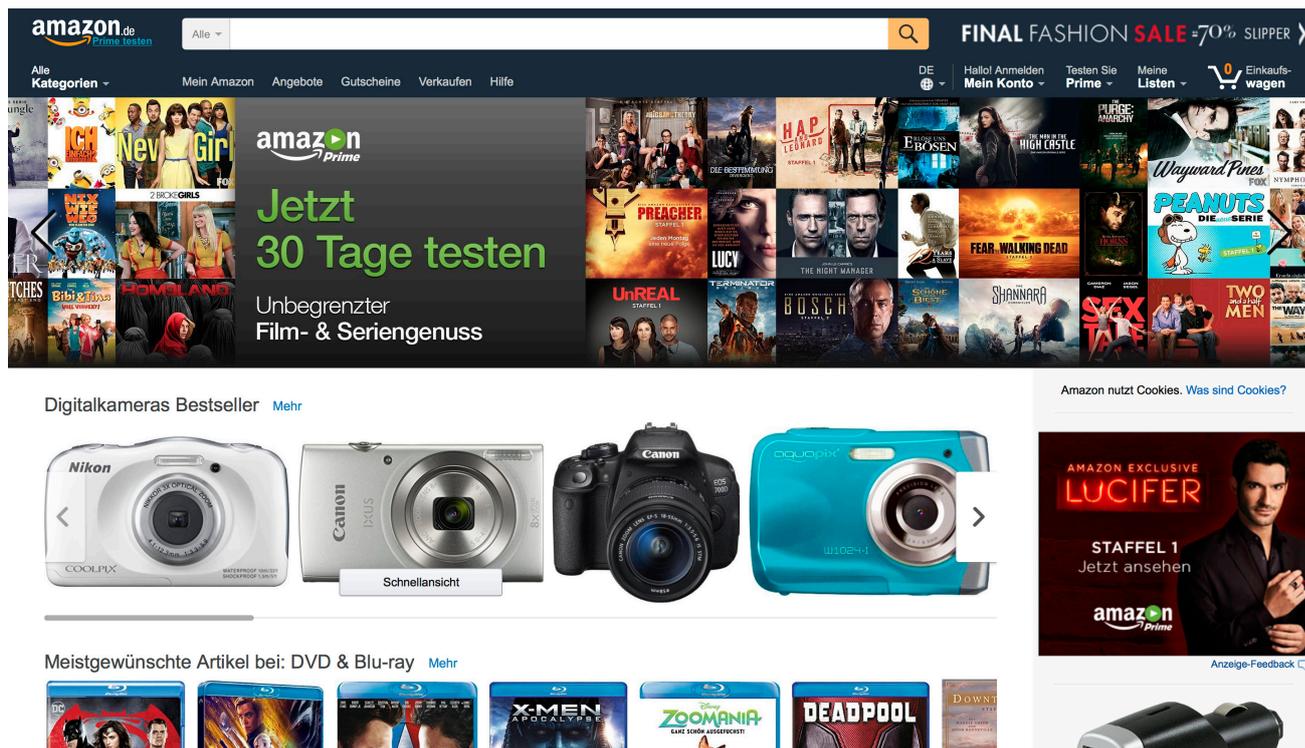
Anhand dieser Beispiele werde ich versuchen, die unterschiedlichen Aspekte von Reputationsystemen näher zu erläutern und auch auf die möglichen Probleme und Risiken der jeweiligen Systeme eingehen. Dies dient vor allem dazu, ein gewisses Basisverständnis zu schaffen, so dass die Überlegungen bezüglich Design und Implementation eines neuen Reputationsystems, im späteren Teil dieser Arbeit besser nachvollzogen werden können.

Da sich Inhalte im Internet in einem stetigen Wandel befinden ist zu beachten, dass es sich hierbei um Momentaufnahmen der jeweiligen Webseiten handelt und sich deren Reputationsysteme auch jederzeit ändern können. Einen solchen Fall bei der Plattform YouTube, werde ich ebenfalls noch genauer beschreiben.

Die Unterkapitel, zu den jeweiligen Webseiten, enthalten alle eine grobe Beschreibung der Plattform selbst und gehen dann näher auf den Einsatz unterschiedlicher Reputationsmechanismen ein.

Anschließend an die Beschreibungen der Plattformen, werde ich auch noch auf eine mögliche Gliederung der Gestaltungselemente für Reputationsysteme eingehen.

# Amazon



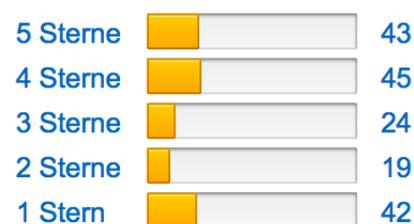
Screenshot der amazon.de Startseite am 12.08.2016

Amazon ist eine der größten Onlineplattform zum Erwerben einer breiten Palette von Produkten sowohl in physischer als auch in virtueller Form.

Um den Benutzern und potentiellen Käufern die Entscheidung, bei der Auswahl der gewünschten Produkte zu erleichtern, bietet Amazon eine Bewertung der jeweiligen erhältlichen Produkte an. Diese Bewertung erfolgt sowohl in numerischer als auch textueller Form und wird mit Hilfe der Benutzer der Plattform generiert. Diese Benutzer können sowohl Kunden, als auch professionelle Rezensenten sein.

Die numerische Bewertung erfolgt mittels der Vergabe einer Ziffer zwischen Eins und Fünf. Diese wird visuell durch fünf ausgefüllte oder leere Sterne dargestellt. Wobei ein ausgefüllter Stern die niedrigste und schlechteste mögliche Bewertung ist und fünf ausgefüllte Sterne die höchste und beste Bewertung darstellen. Diese Einzelbewertungen, der Benutzer und Rezensenten, werden dann gemittelt als Gesamtbewertung dargestellt. Dieser Endwert wird visuell auch wieder mit Sternen dargestellt, wobei neben den Sternen auch der auf eine Kommastelle gerundete gemittelte Wert angegeben wird. Zusätzlich wird auch noch angezeigt wie viele Bewertungen für das jeweilige Produkt

★★★★☆ 173  
3.2 von 5 Sternen



Die Bewertung eines Produktes. Zuerst die gemittelte 5-Sterne Bewertung. Darunter aufgegliedert die abgegebenen Stimmen je Wertung.

abgegeben wurden. Dies hilft vor allem einzuschätzen, wie repräsentativ der gemittelte Wert ist. In einer weiteren Detailansicht, kann der Benutzer auch die Summe der Bewertungen in einem Balkendiagramm aufgeschlüsselt nach vergebener Sternzahl einsehen.

56 von 59 Kunden fanden die folgende Rezension hilfreich

★★★★☆ **Gute Unterwasserkamera für den kleinen Geldbeutel**, 21. Juni 2015

Von **Becy Müller**

Verifizierter Kauf (Was ist das?)

Rezension bezieht sich auf: **Easypix 10012 Unterwasser Digitalkamera "Aquapix W1024-I Splash" in Eisblau (Camera)**

Da die Rezensionen hier ja hoffen ließen, habe ich mir die Kamera für den Ägyptenurlaub bestellt und war wirklich angetan. Es gibt zwar viele verwackelte Bilder, wenn man mitten im Bild von einer Welle erwischt wird, aber wir hatten auch wirklich eine raue See dieses Jahr.

Die unverwackelten Bilder sind wirklich hübsch, auch die Filme werden sehr schön und die Qualität über Wasser kann sich auch sehen lassen. Natürlich kann man keine Profiqualität erwarten, aber das liegt sicher zum Teil auch am Fotografen.

Ich habe die Kamera mit Akkus bestückt, die ich Abends immer aufgeladen habe. Eine Akkulaufzeit hielt etwa einen 8-stündigen Schnorcheltag (mit An/Aus-Schalten, Videos drehen, Bilder auf dem Boot anschauen..).

Wir sind dummerweise oft an den Knopf gekommen der zwischen Foto und Video wechselt, dafür haben wir jetzt ein paar sehr künstlerische Videos vom Weg aus dem Meer zur Strandduche :D

Wer nicht viel Geld für eine Unterwasserkamera ausgeben will/kann, kann sich bedenkenlos dieses Schätzchen bestellen und wird mit hübschen Fotos belohnt. Und wenn man mal nicht im Meer ist, eignet sich die Kamera auch prima für lustige Fotos im Pool. :)



Helfen Sie anderen Kunden bei der Suche nach den hilfreichsten Rezensionen

[Missbrauch melden](#) | [Kommentar als Link](#)

War diese Rezension für Sie hilfreich?  Ja  Nein

Eine ausführliche Bewertung eines Produktes inklusive Beispielbilder.

Zusätzlich zur Sternbewertung können Benutzer auch noch eine Bewertung in geschriebener Form abgeben, sogenannte Rezensionen, um so genauer zu beschreiben, wie es denn zu ihrer endgültigen Sternbewertung gekommen ist. Um die Meinungsbildung anhand der Rezension noch etwas zu verbessern bzw. um den Benutzern eine Möglichkeit zu geben, einschätzen zu können wie vertrauenswürdig oder repräsentativ eine Rezension ist, kann man als Benutzer eine Rezension als hilfreich oder weniger hilfreich markieren. Es werden dann Rezensionen, die als hilfreich markiert wurden, visuell stärker hervorgehoben, als solche, die als weniger hilfreich markiert wurden. Zusätzlich dazu werden auch zwei sich stark in der Bewertung unterscheidende nützliche Rezensionen automatisiert gegenübergestellt. Dies geschieht wohl in der Annahme dem Benutzer so auf kürzestem Weg Pro und Kontra Informationen geben zu können.

Um Benutzer dazu zu motivieren gute Rezensionen zu schreiben, gibt es eine Rangliste der besten Rezensenten. Die Position innerhalb dieser Rangliste kann man verbessern, in dem man mehrere Rezensionen schreibt und diese häufig als hilfreich bewertet werden.

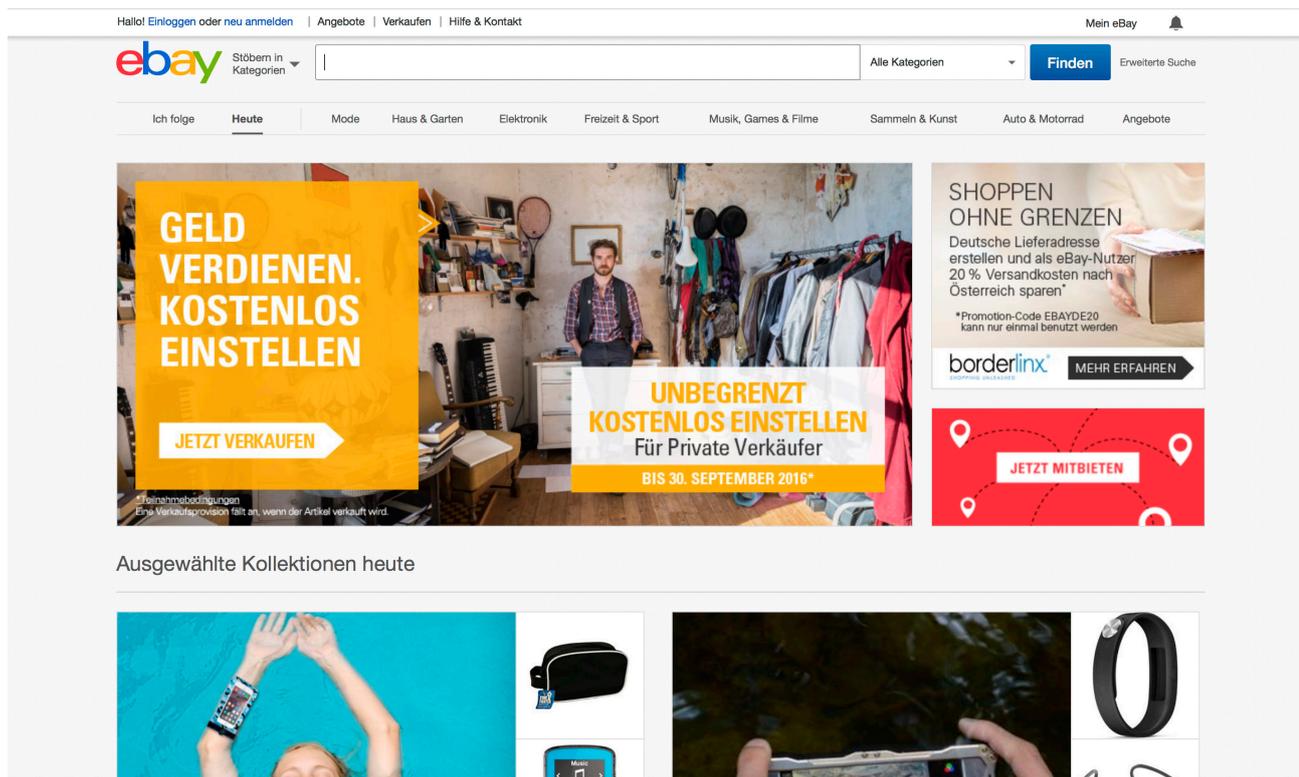
Als weiteren Motivator gibt es auch noch einige wenige Badges beziehungsweise Achievements, falls man etwa in die Top 10/100/1000 der Rangliste kommt.

Dies hat aber laut [David & Pinch, 2005] auch den etwas unerwünschten Effekt gehabt, dass Rezensenten einfach die Bewertungen von anderen Rezensenten oder auch von sich selbst kopiert und bei den selben oder ähnlichen Produkten erneut verwendet haben, um so in der Rangliste höher platziert zu sein. Vor allem das mehrmalige Verwenden eigener Texte und Textpassagen hat einen negativen Effekt für die anderen Benutzer, die sich mit Hilfe dieser Rezension eine Meinung zu den jeweiligen Produkten bilden wollen.

#### TOP 50 REZENSENT

#### #1 REZENSENT

Zwei Beispiele für Badges, die man an entsprechenden Positionen in der Rezensenten Rangliste erhält.



Screenshot der ebay.at Startseite am 12.08.2016

Einer der am stärksten frequentierten [alexa.com] Online-Marktplätze für das Versteigern und Verkaufen von gebrauchten sowie fabrikneuen oder handgefertigten Produkten. eBay selbst tritt nur als Vermittler zwischen den Parteien und Betreiber der Website auf. Das Risiko der Transaktion liegt somit, wenn wir Treuhandservices Außen vorlassen, bei Käufer und Verkäufer. Deshalb ist ein Reputationssystem unabdingbar, um sich als Käufer ein Bild vom Verkäufer machen zu können und umgekehrt. Somit wird die Reputation primär zur Vertrauensbildung zwischen den Benutzern eingesetzt.

Die Bewertung von Benutzern geschieht bei eBay auf Transaktionsbasis. Sobald A und B ein Geschäft getätigt haben, können diese sich gegenseitig im Kontext dieser Transaktion bewerten. Die Verkäuferbewertung geschieht grundsätzlich binär, sprich positiv oder negativ. Aber auch eine neutrale, nicht gezählte, Bewertung ist möglich. Zusätzlich kann ein Käufer mit einer Stern Bewertung zwischen Eins und Fünf ähnlich derer bei Amazon, die unterschiedliche Aspekte der Transaktion (Artikelbeschreibung, Kommunikation, Versandzeit, Versandkosten) bewerten. Eine weitere Option bei der Verkäuferbewertung ist auch noch ein geschriebener Kommentar.

Der Käufer kann vom Verkäufer nur positiv oder gar nicht bewertet werden, was durchaus ein Ungleichgewicht darstellt. Vor 2008 waren negative Bewertungen für Käufer noch möglich [theguardian, 2008]. Diese Umstellung wurde durch den Wandel der Handelskultur auf eBay begründet, da die Plattform immer stärker von professionellen Verkäufern eingesetzt wurde.

Die Reputation wird auf eBay in drei Formen visuell dargestellt. Einmal als Zahl, dem sogenannten Bewertungspunktstand. Zusätzlich mit einem Icon in Form eines farbigen Sterns. So wie einer prozentuellen Angabe der positiven Bewertungen im Verhältnis zu allen erhaltenen Beurteilungen. Jeder Benutzer hat auch noch ein ausführlicheres Bewertungsprofil, in dem alle seine bisherigen getätigten und erhaltenen Bewertungen detailliert aufgelistet und für alle anderen Benutzer einsehbar sind.

Der Bewertungspunktstand wird anhand der erhaltenen Käuferbewertungen berechnet, indem für alle erhaltenen positiven Bewertungen +1 Punkt aufsummiert und für alle negativen Bewertungen -1 Punkt abgezogen wird.

Das Sternicon ändert sich abhängig vom Bewertungspunktstand farblich und auch in seiner Form (Stern, Stern mit Schweif, Stern mit Schweif und Partikeln), um so dem Benutzer eine schnellere Einschätzung der Reputation zu ermöglichen.

Die Berechnung der prozentuellen Angabe geschieht Anhand der Bewertungen die der Verkäufer innerhalb der letzten 12 Monate erhalten hat. Hierzu wird die Anzahl der positiven Bewertungen durch die Anzahl der negativen und positiven Bewertungen dividiert. Die Anzahl der neutralen Bewertungen wird in dieser Berechnung nicht berücksichtigt. Um eine zu starke Verzerrung durch Mehrfachbewertungen des selben Käufers bei mehreren Transaktionen zu verhindern, werden dessen Bewertungen (positiv & negativ) gemittelt und als einzelne positive oder negative Bewertung gezählt.

## Angaben zum Verkäufer

hegibaer (15266  ) mich

99,9% Positive Bewertungen

### Bewertungen

	<b>1.332</b>	Artikel wie beschrieben
	<b>1.454</b>	Kommunikation
	<b>1.329</b>	Versandzeit
	<b>1.327</b>	Versandkosten

 **2.068**     **3**     **1**  
Positiv                      Neutral                      Negativ

Die unterschiedlichen Formen der Bewertungen eines Verkäufers.

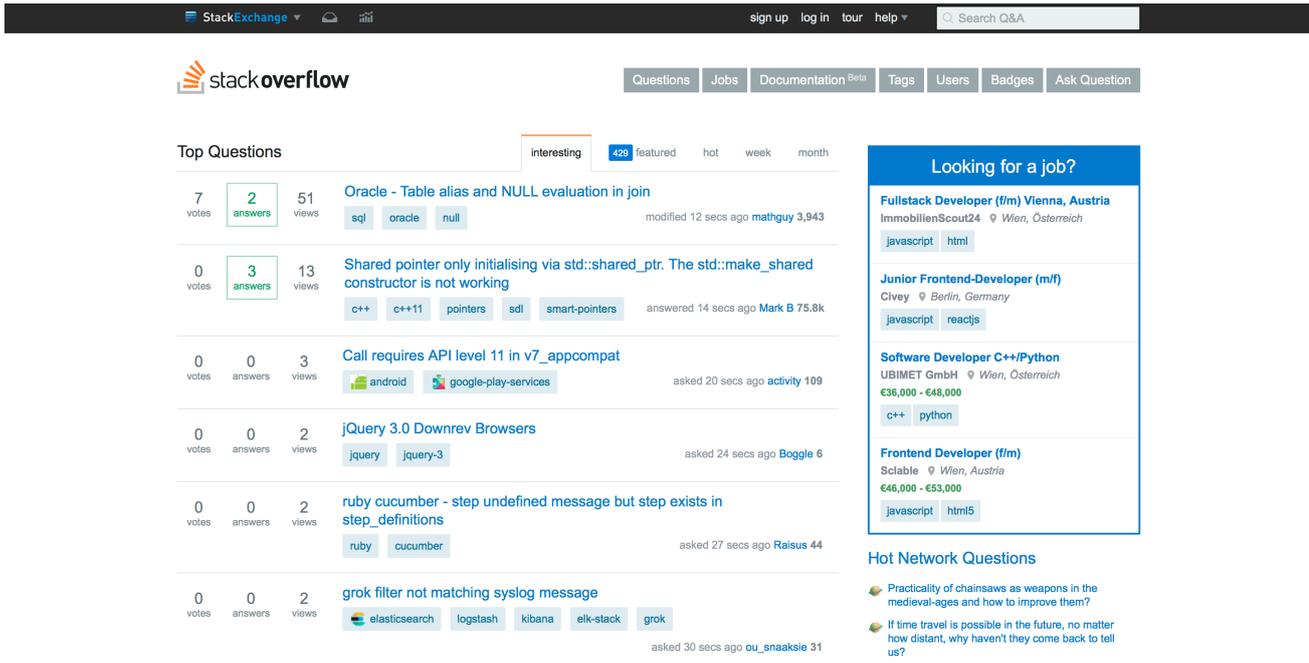
Oben: zusammengefasst mit Punkten, Sternicon & prozentueller Angabe der positiven Bewertungen.

Mitte: Die gemittelten Ergebnisse über alle Bewertungen eines Verkäufers in unterschiedlichen Kategorien.

Unten: Übersicht über die Bewertungen der letzten 12 Monate

- Gelber Stern (  )
- Blauer Stern (  )
- Türkisfarbener Stern (  )
- Violetter Stern (  )
- Roter Stern (  )
- Grüner Stern (  )
- Gelbe Sternschnuppe (  )
- Türkisfarbene Sternschnuppe (  )
- Violette Sternschnuppe (  )
- Rote Sternschnuppe (  )
- Grüne Sternschnuppe (  )
- Silberne Sternschnuppe (  )

Unterschiedliche Sternicons in ihrer Wertigkeit von oben nach unten aufsteigend sortiert.



Screenshot der stackoverflow.com Startseite am 12.08.2016

Wenn man als Programmierer oder Administrator derzeit im Internet nach Problemlösungen sucht ist die Wahrscheinlichkeit sehr hoch, dass man bereits bei einer Diskussion auf Stackoverflow oder einer der anderen Plattformen von Stackexchange gelandet ist. Diese sind Foren in denen von den Benutzern Fragen zu bestimmten Fachgebieten gestellt, diskutiert und beantwortet werden können. Wenn dem Fragenden nun eine bestimmte Antwort geholfen oder seine Frage komplett beantwortet hat, kann jener diese als „richtige“ Antwort markieren. Zusätzlich können auch die anderen Benutzer der Plattform, Diskussionsbeiträge oder auch die Frage selbst hinauf oder hinunter bewerten. Basierend auf diesen Bewertungen erhalten die Autoren der jeweiligen Beiträge Punkte. Aus diesen Punktezahlen errechnet das Reputationsystem von Stackoverflow nun mehrere Reputationsparameter, welche die Benutzer dazu anregen sollen anderen Benutzern zu helfen.

▲ You don't have FULL JOINS

298 For a code SAMPLE transci

▼ with two tables t1, t2:

✓

```
SELECT * FROM t1
LEFT JOIN t2 ON t1.id =
UNION
SELECT * FROM t1
RIGHT JOIN t2 ON t1.id
```

share improve this answer

Ein als die „richtige“ Antwort markierter Beitrag. Ersichtlich am grünen Häkchen Symbol. Dem Autor werden auch 298 Punkte gutgeschrieben, welche sich aus der Differenz von positiven und negativen Bewertungen ergeben.

Benutzer können sowohl Punkte gewinnen als auch verlieren, wobei es leichter ist Punkte zu gewinnen. Der mögliche Zugewinn je Tag ist aber limitiert. Für die folgenden Aktionen können Benutzer laut Stackoverflow [stackoverflow reputation] Punkte erlangen:

- Eine Frage des Benutzers wird hinauf gewertet
- Eine Antwort des Benutzers wird hinauf gewertet

- Eine Antwort des Benutzers wird als „akzeptiert“ gekennzeichnet, sprich der Fragende gibt an, dass der Beitrag des Benutzers, die Lösung zu seinem Problem war. Hier erhält auch der Fragende ein kleines Punkteplus, um so motiviert zu werden, Antworten entsprechend zu markieren.
- Ein Änderungsvorschlag des Benutzers wird angenommen. In Stackoverflow ist es auch möglich Änderungsvorschläge zu Beiträgen von anderen Benutzern zu machen, um so etwa kleinere Fehler wie z.B. Syntaxfehler in Codestücken zu korrigieren oder bei älteren Beiträgen auf den neuesten technischen Stand zu bringen.
- Wenn man Bounty (Kopfgeld) für eine Antwort erhält. Zu diesem System gleich mehr.
- Wenn man seinen Account mit mehreren der Stackexchange Plattformen verknüpft. Das soll vor allem dazu anregen, die anderen Plattformen auszuprobieren.

Verlieren kann man Punkte hingegen aber mit den folgenden Aktionen:

- Wenn eine Frage hinunter gewertet wird
- Wenn eine Antwort hinunter gewertet wird
- Wenn man selbst eine Antwort von jemanden hinunterwertet. Dies soll vermutlich verhindern, dass Benutzer gemobbt werden und das Hinunterwerten nur in angebrachten Fällen eingesetzt wird.
- Wenn man eine Bounty auf seine Frage setzt.
- Wenn eine Antwort mehrmals als Spam markiert wird, verliert der Autor Punkte.

Neue Benutzer beginnen mit einem Punktestand von 1 und können auch nicht unter diesen absinken.

Wie bereits erwähnt, gibt es auf Stackoverflow auch noch ein Bounty System [stackoverflow bounty]. Bei diesem kann ein Benutzer seine bereits erarbeiteten Punkte wie ein Zahlungsmittel einsetzen und eine Bounty auf eine eigene, aber auch eine Frage von einem anderen Benutzer ausrufen. Diesen Betrag an Punkten muss der Benutzer aber bereits im Vorhinein zahlen und kann auch nicht mehr zurück gefordert werden. Ist eine Bounty auf eine

Frage ausgesetzt, wird diese auf der Plattform besonders hervorgehoben. Wird die Frage dann beantwortet und der Fragende akzeptiert diese Antwort, wird der Betrag dem Antwortenden gut geschrieben. Wird vom Fragenden keine Antwort als akzeptiert markiert, erhält nach einer gewissen Zeit der Autor der Antwort, mit der besten Bewertung, einen Teil der Bounty.

Profile Activity Meta User Network Profile



**100,550** REPUTATION

18 170 215

**Max Mustermann** top 0.13% overall

Programmer at ACME Inc.

Programmer from Emerald City, Oz. Love C/C++/C#, Scheme, Python and Java.

**1,909** answers

**51** questions

**~10.7m** people reached

Emerald City, Oz

maxmuster.acme.com

Member for 7 years, 8 months

6,952 profile views

Last seen yesterday

**Communities** (23)

- Stack Overflow 100.6k
- Super User 1.2k
- Server Fault 619
- Programmers 261
- Ask Different 221

[View network profile →](#)

**Top Tags** (1,243)

java	SCORE 2,150	POSTS 356	POSTS % 18
c	SCORE 1,005	POSTS 172	mysql
sql	SCORE 811 POSTS 186	c++	SCORE 773 POSTS 122
		iphone	SCORE 745 POSTS 170

[View all tags →](#)

Das anonymisierte Profil eines Stackoverflow Benutzers.

Basierend auf den Punkten setzt Stackoverflow auch Badges ein. Wobei es hier unterschiedliche Varianten gibt. Am präsentesten sind die sogenannten „Tag Badges“. Diese werden anhand der, den Fragen zugewiesenen Schlagworte berechnet, indem die Punkte je Schlagwort summiert werden und der Benutzer dann einen Badge mit dem Schlagwort als Namen und einer Wertigkeit von Bronze, Silber oder Gold erhält. Diese Wertigkeit wird anhand der Summe an Punkten unter diesem Schlagwort ermittelt.

Zusätzlich dazu gibt es auch noch eine Sammlung von Badges die man für bestimmte Leistungen, in sich auf Plattformfunktionen beziehende Kategorien, erhalten kann. Diese Kategorien sind „Fragen“, „Antworten“, „Teilnahme“, „Moderation“ und „Sonstige“. Die Badges in den Kategorien „Fragen“ und „Antworten“ sind zum Großteil auf das Erreichen gewisser statistischer Werte, wie etwa die Menge an Ansichten oder erhaltenen Punkten bezogen. Aber auch für simple Aktionen und Grundfunktionen, wie

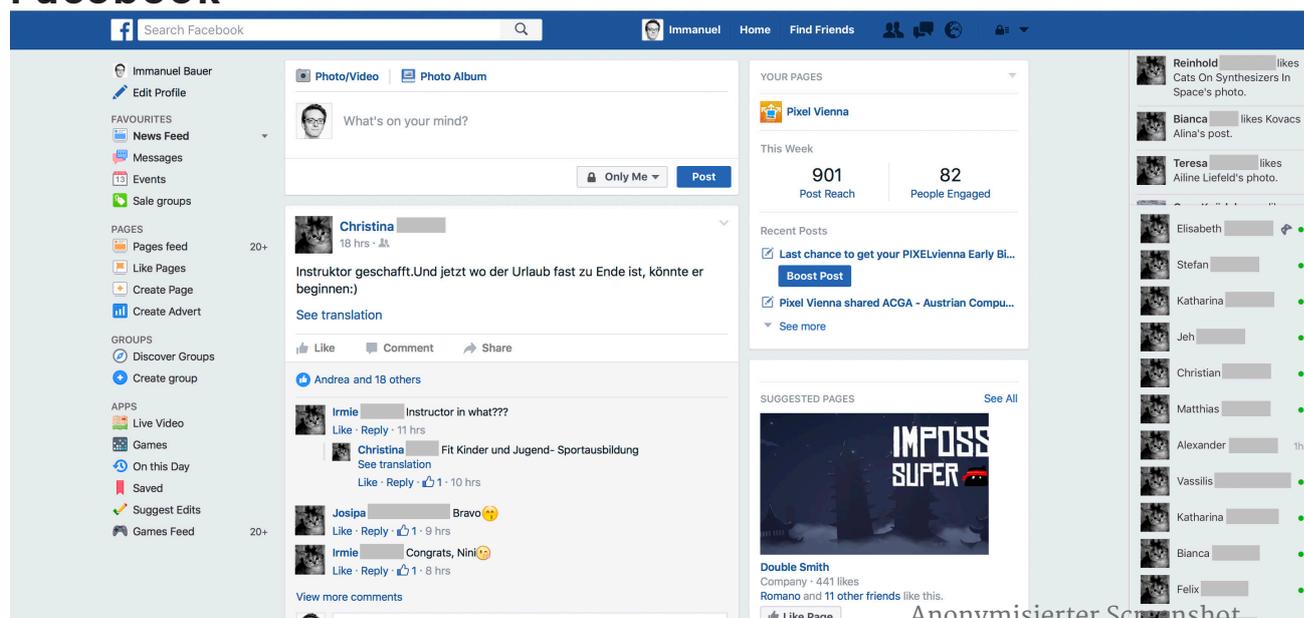
beispielsweise das erste mal eine Frage stellen oder beantworten gibt es Badges, um so Benutzer zu motivieren, diese Funktionen zumindest einmal auszuprobieren.

Die Badges der Kategorie „Moderation“ sollen die Benutzer vor allem dazu motivieren moderative Aktionen zu setzen, wie zum Beispiel Fragen mit Schlagworten versehen, das Erstellen von Änderungsvorschlägen oder das Markieren von hilfreichen oder anstößigen Beiträgen.

Die Badges unter „Sonstiges“ belohnen Benutzer dafür, Informationsseiten wie beispielsweise das FAQ von Stackoverflow durchzulesen.

Wie die Tag Badges, haben auch die Badges der anderen Kategorien zum Teil Stufen von Bronze, Silber und Gold und auch jeweils andere Bezeichnungen z.B. „Nice Answer“ (Bronze), „Good Answer“ (Silber) und „Great Answer“ (Gold).

## Facebook



Anonymisierter Screenshot der persönlichen facebook.com Startseite des Autors am 13.08.2016

Eines der weltgrößten sozialen Netzwerke und stark frequentierten Webseiten ist wohl Facebook. Ein sehr stark, mit dieser Plattform in Verbindung gebrachtes Symbol, ist eine geschlossene Hand mit einem Daumen nach oben und optional dazu der Schriftzug „Gefällt mir“. Dieses ist vor allem in Form einer Schaltfläche unter Beiträgen von Benutzern wiederzufinden. Wenn nun ein Beitrag mit einer solchen „Gefällt mir“ Schaltfläche versehen ist, kann ein Benutzer mittels Klick auf diese Schaltfläche den Beitrag markieren und somit abhängig vom Inhalt des Beitrages seine Zustimmung geben. Abhängig vom Inhalt deswegen,

 Gefällt mir  Teilen 2,8 Mio. Personen gefällt das.

Das „Gefällt mir“ / „Like“ Button Widget zum Einbinden in eine externe Website. Diese Variante bietet zusätzlich auch noch einen „Teilen“ Button, sowie eine Auskunft darüber wie viele Personen die Seite, in die das Widget eingebunden wurde, bereits „geliked“ haben.



Die unterschiedlichen Möglichkeiten auf einen Beitrag zu reagieren bzw. diesen zu markieren.

da es durchaus auch passieren kann, dass negative Inhalte mit „Gefällt mir“ markiert werden, da so Benutzer ihr Beileid o.Ä. kundtun wollen.

Je mehr solcher Likes ein Benutzer mit seinen Beiträgen sammelt, umso „wichtiger“ bzw. verbreiteter sind die Beiträge dieses Benutzers. Soweit handelt es sich hierbei um ein simples binäres Bewertungssystem, das Interessante ist aber, dass das Facebook Like System auch über die eigene Plattform hinaus geht und man dieses auch über einen „Gefällt mir“ Button in externe Webseiten integrieren kann.

Die Omnipresenz des „Gefällt mir“ Buttons ist aber auch gleichzeitig dessen Problem. So führt dessen häufige Verwendung entweder dazu, dass dieser komplett ausgeblendet wird, da er ein Teil des visuellen Rauschens rund um die eigentliche Information wird oder geradezu reflexartig betätigt wird, bei allem, was man annähernd interessant fand und somit Aussagekraft verliert.

Anfang 2016 hat Facebook neben dem „Gefällt mir“ fünf weitere Emotionen eingeführt um eine etwas passendere Reaktion auf den Beitragsinhalt zu erlauben. Bei diesen handelt es sich um die folgenden: „Love“ (Liebe), „Haha“ (Lustig), „Wow“ (Erstaunt), „Sad“ (Traurig), „Angry“ (Wütend).

Die bisher aus der Anzahl der Likes abgeleitete Reputation lässt sich aber weiterhin durch einfaches aufsummieren der Reaktionen vergleichbar berechnen. Zusätzlich gäbe es jetzt auch die Möglichkeit beispielsweise den „traurigsten“ Beitrag zu ermitteln.

Ein weiterer Reputationparameter auf Facebook ist die Anzahl der Freunde. Dieser ist aber eher implizit, da diese Angabe wohl nicht gezielt als Reputationparameter entworfen wurde, aber von manchen Benutzern durchaus dazu genutzt wird um sich mit anderen Benutzern zu messen. Um dem entgegen zu wirken, hat Facebook wohl auch die Anzahl der möglichen Freunde limitiert. Diese Maßnahme soll aber natürlich auch Firmen dazu bewegen Facebook Seiten an Stelle von regulären Profilen anzulegen. Diese Facebook Seiten können an Stelle von Freunden wieder nur Likes sammeln. Dies aber unlimitiert.

Mittels einer Analyse des sozialen Netzwerkes eines Benutzers sowie der Popularität dessen Beiträge kann man ebenfalls eine Reputationwertung herleiten. Dies wird aber nicht von Facebook selbst nach Außen kommuniziert. Es gibt aber Services von Dritten, wie z.B. Klout, die sich die APIs von Facebook zum Nutzen machen, um solche Analysen, mehrere Soziale Netzwerk Plattformen übergreifend, anzustellen.

## Reddit

The screenshot shows the Reddit homepage with a navigation bar at the top containing categories like 'START', 'ALLE', 'ZUFALLIG', 'ASKREDDIT', 'FUNNY', 'GIFS', 'MOVIES', 'VIDEOS', 'TODAYLEARNED', 'PICS', 'NEWS', 'WORLDNEWS', 'GAMING', 'OLYMPICS', 'AWW', 'SHOWERTHOUGHTS', 'IAMA', 'NOTTHEONION', 'MILDLYINTERESTING', 'SPORTS', 'TELEVISION', 'JOKES', 'OLDS', and 'MEHR'. Below the navigation bar, there's a search bar and a login section. The main content area displays a list of posts with their titles, scores, and submission times. The top post is titled 'TIL in the original 1900 novel "The Wonderful Wizard of Oz", the Emerald City isn't actually green; the Wizard requires everyone inside to wear green tinted glasses to make them think it is' with a score of 5213. Other posts include 'Katie Ledecy still waiting for others to finish', 'Helicopter Pilots View', 'Kyrgyzstan president: "Women in mini skirts don't become suicide bombers"', 'Novak Djokovic is classy AF', 'This camera pan of Ledecy is the shot of the Olympics', 'By age 3, kids know when you owe them one.', and 'Irish Olympians Giving a Serious Interview after Winning Silver in Double Sculls'.

Ist ein Social News Aggregator beziehungsweise Forum mit unzähligen von Benutzern erstellten, Unterforen, sogenannten „Sub-Reddits“, und ermöglicht es seinen Benutzern Links zu Inhalten auf anderen Webseiten zu teilen oder Inhalte direkt über das Forum zu veröffentlichen. Diese Inhalte können dann von allen Benutzern der Plattform kommentiert, diskutiert und bewertet werden. Die Bewertung von Beiträgen auf Reddit läuft so ab, dass diese von Benutzern hinauf oder hinunter gewertet werden können. Dies entspricht im Grunde wieder einem Binären System wie es Amazon oder Stackoverflow einsetzen. Wobei Reddit hervorhebt, dass man Links beziehungsweise Beiträge auf Basis dessen bewerten soll, ob man diese als interessant oder uninteressant empfunden hat [reddit voting]. Die Bewertung der Benutzer werden dann zu einem Gesamtscore aufsummiert. Wobei jedes hoch werten +1 und jedes hinunter werten -1 Punkt bringt. Je höher der Score, desto interessanter ist der Beitrag und desto prominenter und höher wird dieser gelistet.

Screenshot reddit.com Startseite am 13.08.2016

This close-up shows a post with a score of 5270 and a question mark icon. The title is 'TIL in the original 1900 novel everyone inside to wear green' and it was submitted 3 hours ago by user 'Notmiefault' with 224 comments.

Der Titel eines Beitrages. Links davon ist eine Grafik, die von der verlinkten Website genommen wird. Ganz links kann man den Beitrag mit einem Klick auf den Pfeil nach oben positiv und mit dem Pfeil nach unten negativ bewerten.

## Max Mustermann

**46.756** post karma

**39.140** Kommentar-Karma

[send a private message](#)

redditor seit 4 Jahre

gifts on behalf of [/u/MaxMustermann](#) have helped pay for **19.92 Stunden** of reddit server time.

### MODERATOR VON

[/r/Muster](#)  
[/r/Anon](#)

### TROPHÄENSAMMLUNG

[Was ist das?](#)



Four-Year Club



Verified Email

Die Reputation Werte im Profil eine Benutzers. Neben dem Karma und den Trophäen wird auch angezeigt in welchen Sub-Reddits der Benutzer Moderator ist sowie auch wie viel der Benutzer Reddit gespendet hat.

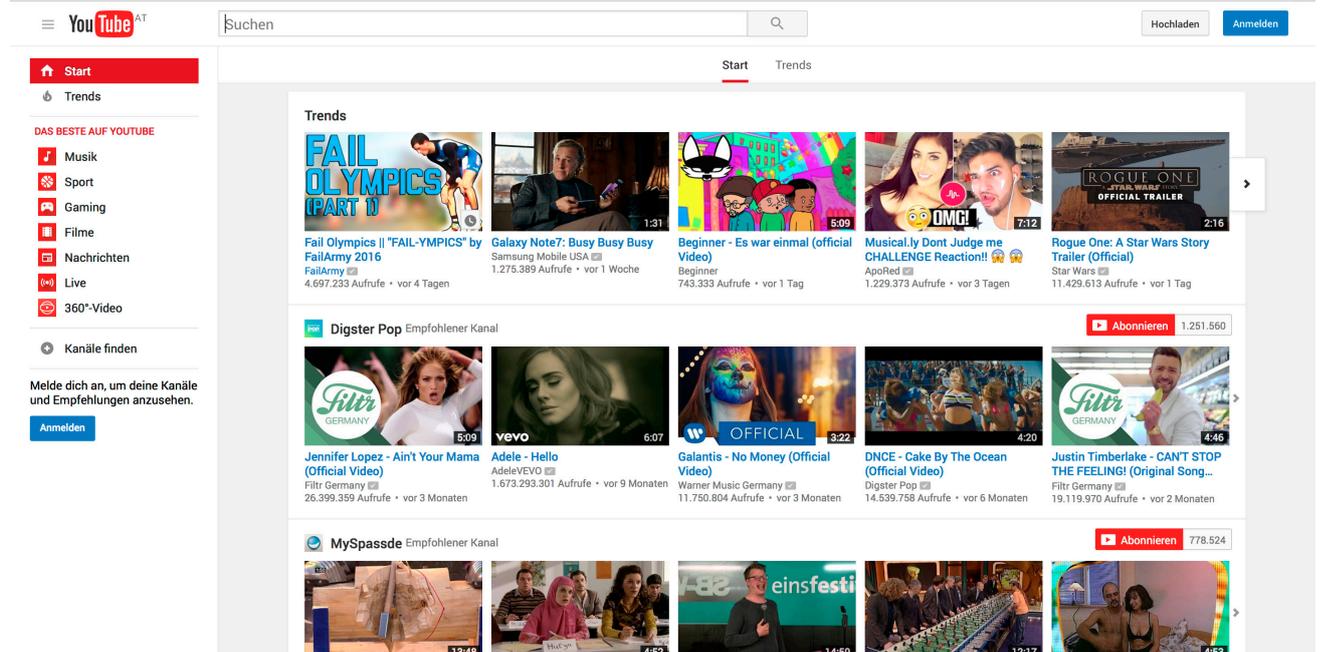
Damit Beiträge nicht von einem Mitläufertum profitieren können, werden die Gesamtpunkte erst nach ein paar Stunden angezeigt. Reddit ist hier leider nicht spezifischer wie lange diese Verzögerung genau ist.

Auf Reddit gibt es auch eine Reputation für die Benutzer. Diese wird als „Karma“ bezeichnet. Die Plattform unterscheidet zwischen einem Post-Karma und einem Kommentar-Karma.

Ein Benutzer erhält Post-Karma, wenn ein von ihm veröffentlichter Link, positiv bewertet wird. Kommentar-Karma bekommen Benutzer für positive Bewertungen auf Kommentare oder Diskussionsbeiträge, zu einem Link von einem anderen Benutzer.

Zusätzlich zum Karma gibt es auch noch Trophäen, sogenannte Awards. Die Gewinner werden einmal täglich automatisch ermittelt und erhalten dann eine digitale Trophäe für ihr Profil. Die Gewinner des vorhergegangenen Tages sind auf einer eigenen Seite gelistet, die für alle Benutzer einsehbar ist. Die Awards sind überwiegend so gestaltet, dass Benutzer dazu motiviert werden Interessantes zu posten und moderativ tätig zu sein, indem sie etwa Anstößiges markieren. Es gibt auch einige Awards die darauf abzielen, dass Benutzer bestimmte Funktionen ausprobieren oder auch welche, die einfach zeigen sollen, wie lange ein Benutzer schon bei Reddit registriert ist. Hier gibt es für jedes Jahr Mitgliedschaft einen eigenen Award. Es gibt auch einige Awards die Benutzer dafür belohnen, dass diese an der Weiterentwicklung der Plattform mithelfen oder die Möglichkeit nutzen, eigene Sub-Reddits grafisch zu gestalten.

Obwohl Karma und Awards manche Benutzer sicher dazu anregen, möglichst viele von diesen zu sammeln, ist deren primärer Zweck auf Reddit neuen Benutzer zu zeigen, wer vertrauenswürdige und erfahrene Benutzer auf der Plattform sind.



Ist die derzeit größte Plattform für das streamen von Videos. Da mittlerweile in jeder Minute etwa 300 Stunden [YouTube statistics] Videomaterial bei YouTube hochgeladen werden, braucht es Hilfsmittel für die Benutzer um die „sehenswerten“ Videos aus dieser Flut heraus zu filtern. YouTube ist diesbezüglich auch ein interessantes Beispiel, da diese Methoden in den letzten Jahren ein gewisse Umgestaltung erfahren haben.

Screenshot youtube.com Startseite am 13.08.2016

In seiner Ursprungsversion bot YouTube die Möglichkeit, Video mittels eines 5-Sterne Systems zu bewerten. Die Benutzer konnten ein Video mit ein bis fünf Sternen bewerten, wobei ein Stern die niedrigste und fünf die höchste Bewertung ist.

Etwa 2009 [YouTube.blogspot, 2009] kam YouTube dann aber zu dem Schluss, dass das 5-Sterne System nicht die gewünschte Effektivität an den Tag legte, da sich der Großteil aller Bewertungen nur auf die extremen Werte 5 oder 1 Stern verteilt haben. Somit wurde das System durch ein binäres Daumen rauf / Daumen runter System ersetzt. Die daraus resultierende Bewertung wird auf der Videoseite in Form eines Balkens und den aufsummierten positiven und negativen Bewertungen angezeigt. In der Listenansicht von YouTube sind diese Informationen nicht mehr direkt einsehbar. Einzig zum sortieren der Liste kann der Wert hergenommen werden.

Der einzige quantitative Wert der durchgängig überall sichtbar ist und als Orientierung bezüglich Reputation verwendet werden kann, ist die Anzahl der Aufrufe beziehungsweise Ansichten. Dies ist aber nur begrenzt ein guter Indikator für die Qualität des Videos, da diese Zahl auch stark von der Beliebtheit des Inhaltes aber auch der Vermarktung des Videos selbst abhängt. So kann es leicht passieren, dass etwas komplexere oder auf Nischen bezogene Themen weit weniger Aufrufe haben als etwa ein amüsantes Katzenvideo.

The screenshot shows a YouTube video player for 'Galaxy Note7: Busy Busy Busy' by Samsung Mobile USA. The video has 1,275,389 views, 9,240 likes, and 1,830 comments. The channel has 345,139 subscribers. A sidebar on the right titled 'Nächstes Video' (Next Video) with an 'Autoplay' toggle shows several recommendations:

- Galaxy Note7 (Galaxy Note7) \_ 제품 소개 영상 (SamsungMobileKorea, 61,529 Aufrufe, 6:02)
- Samsung Galaxy Note7 Unpacked (Official Replay) (Samsung Mobile, 628,596 Aufrufe, 52:53)
- Galaxy Note 7 vs. Note 5 - Hands On Comparison (ListenToJulian, 78,714 Aufrufe, NEU, 4:39)
- Galaxy Note 7: In-Depth Look! (Initial Review) (Erica Griffin, 505,959 Aufrufe, 15:04)
- Samsung Galaxy Note 7 Duos - Unboxing & Hands On! (C4ETech, 233,716 Aufrufe, NEU, 5:18)
- Samsung Galaxy Note 7 Hands-On Review (GarnetMatch)

Die Detailseite eines Videos. Auf der linken Seite sieht man unter dem Titel die Zahl der Abonnenten und auf der rechten Seite die Gesamtzahl der Aufrufe des Videos sowie wie viele Benutzer das Video gemocht haben oder auch nicht.

Damit Benutzer aber dennoch Videos zu sehen bekommen, welche diese interessieren, generiert YouTube eine Liste von Vorschlägen. Wie diese Videovorschläge genau generiert werden, ist leider nicht bekannt, wird aber zumindest teilweise ebenfalls auf die Bewertungen sowie Aufrufzahlen der Videos zurückgreifen.

Etwas Aufschluss darüber welche Werte YouTube für seine Berechnung zu Verfügung stehen, lässt etwa die Statistik Ansicht auf der Detailseite eines jede Videos zu, welche die unterschiedlichen Parameter zeitlich aufschlüsselt.

Für die registrierten Benutzer von YouTube wird ebenfalls keine explizite Reputationwertung ausgewiesen. Man kann aber die Kanalinformationen eines Benutzers als Anhaltspunkt verwenden. In diesen kann man einsehen wie viele Abonnenten (andere Benutzer die den Benutzer explizit abonniert haben und somit über dessen neue Beiträge

informiert werden) ein Benutzer bereits hat und wie viele Ansichten insgesamt.

## Twitter



Diese Website stellt seinen Benutzern einen Microblog zur Verfügung in den diese Einträge von maximal 140 Zeichen Länge machen können. Benutzer können reine Textnachrichten veröffentlichen, aber auch Links und Bilder. Es gibt auch die Möglichkeit Beiträge anderer Benutzer mittels eines sogenannten Retweets, mit einem Verweis auf die original Nachricht, über den eigenen Microblog erneut zu veröffentlichen. Falls dem Benutzer ein Beitrag sehr gut gefällt kann er diesen als Favorit markieren. Benutzer können auch die Microblogs anderer Benutzer abonnieren und bekommt dann die Beiträge dieser abonnierten Microblogs in einer Liste chronologisch angezeigt. Man kann auch auf Nachrichten direkt Antworten oder private Konversationen führen. Somit kann man Twitter auch als Kommunikations- und Diskussionsmedium verwenden.

Twitter selbst setzt auf keine expliziten Reputationmechanismen somit ergibt sich die Reputation eines Benutzers auf dieser Plattform nur implizit durch gewisse öffentlich einsehbaren Parameter.

Der Hauptfaktor der Reputation eines Benutzers auf Twitter ist wohl die Anzahl seiner „Follower“ sprich Abonnen-

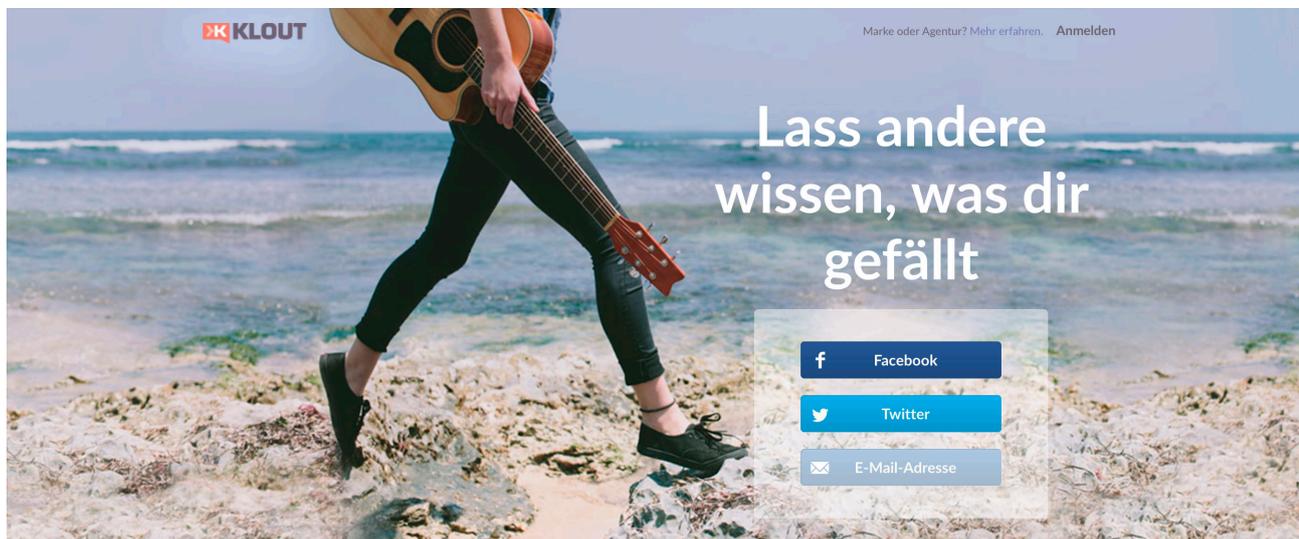
Screenshot des offiziellen @twitter Profils auf twitter.com am 13.08.2016

ten. Obwohl diese Zahl alleine nicht sehr viel Aussagekraft über die Qualität der Beiträge eines Benutzers hat. Beispielsweise hatte etwa der Papst bereits knapp 700.000 Follower [welt.de] ohne noch einen einzigen Beitrag erstellt zu haben.

Etwas Aussagekräftiger ist hier schon die Anzahl der „Retweets“ sprich erneutes veröffentlichen beziehungsweise zitieren des Beitrages durch andere Benutzer. Dazu kommt auch noch die Zahl der Benutzer, die den Beitrag als Favorit markiert haben.

Auch wenn diese Werte nicht explizit als Reputation ausgegeben werden, greift Twitter dennoch auf diese zurück um die sogenannte „Entdecken“ Funktion mit guten Vorschlägen zu versorgen. Insbesondere in der Kategorie „Beliebte Accounts“. Diese Liste kann nach Themengebieten gefiltert werden und schlägt Accounts vor die viele vermutlich relevante, sprich gelesenen, Tweets sowie eine engagierte Interaktionen mit ihren Followern haben.

## Klout



Die beste Möglichkeit, Deinen Online-Einfluss zu verbessern, besteht darin, interessante Inhalte zu erstellen und mit anderen zu teilen. Genau dabei hilft dir Klout.



Screenshot klout.com Startseite  
am 12.08.2016

Diese Plattform versucht, anhand der Aktivitäten einer Person und der Reaktionen von anderen Benutzern, den Einfluss dieser Person anhand einer einzigen Zahl zwischen 0-100 darzustellen. Diese Zahl bildet dann den sogenannten Klout Score. Dieser Score wird anhand des Einflusses auf acht unterschiedlichen Social Media Platt-

formen, wie z.B. die bereits erwähnten Plattformen Facebook und Twitter ermittelt.

Klout analysiert dazu die Aktivitäten, der letzten 90 Tage, einer Person, die ihre Accounts mit Klout verknüpft hat. Wobei sich der Score vor allem aus dem Verhältnis von der Anzahl der eigenen Aktivitäten und der Anzahl der Reaktionen errechnet. Man kann seinen Score auch verbessern, indem man mit anderen Leuten interagiert, die ebenfalls einen hohen Klout Score haben. Dieses Detail birgt aber die Gefahr, dass sich eine Gruppe von „Einflussreichen“ bilden könnte, die sich gegenseitig den Score aufbessern. Um dem etwas entgegen zu wirken, sind mehrfache Reaktionen auf Aktivitäten einer Person von der selben Person weniger „Wert“ als die einer neuen Person oder einer Person die seltener reagiert.



Das dahinterstehende Businessmodell ist mittlerweile das Verkaufen eines Services an Unternehmen, mit dessen Hilfe man den Social Media Auftritt messen und optimieren kann. Dies reicht von identifizieren von Zielgruppen bis hin zu gezieltes Kontaktieren von „Schlüsselpersonen“ in diesen Gruppen um diese als Evangelist zu gewinnen.

Eine Informationsgrafik von klout dazu, wie Ihre Wertung zu Stande kommt.



## Programming Tips & Tricks

- [Stop / remove all Docker containers](#)  
149 [fgrehm](#) · 21 responses · [linux](#) [docker](#)
- [JavaScript iterate through object keys and values](#)  
15 [stevenseski](#) · 5 responses · [jquery](#) [javascript](#)
- [Don't use Array.forEach, use for\(\) instead](#)  
33 [afshinn](#) · 20 responses · [performance](#) [loop](#) [javascript](#)
- [Uninstall all those broken versions of MySQL and re-install it with Brew on Mac Mavericks](#)  
64 [corysimmons](#) · 40 responses · [mysql](#) [mac](#) [brew](#)
- [How to create an image slider with javascript](#)  
23 [guangyi](#) · 7 responses · [practice](#) [javascript](#) [image](#) [slider](#)
- [Fully custom select box, simple css only](#)  
92 [oldboy](#) · 20 responses · [css](#) [select](#)

### Categories

- [git](#) [Node.js](#) [Vim](#) [Ruby](#) [Rails](#)
- [Command Line](#) [Javascript](#) [HTML/CSS](#)
- [.NET](#) [Devops](#) [iOS](#) [Android](#)
- [OS Hacks](#)

### Latest Programming Jobs

- [Manager of Mobile App Development, Android- Digital Ventures](#)
- [Starbucks](#) · Seattle, Wa · Full Time
- [Senior Software Engineer - Infrastructure](#)
- [FiveStars](#) · San Francisco · Full Time
- [Senior Software Engineer - Mobile \(Android or iOS\)](#)
- [FiveStars](#) · San Francisco · Full Time

Screenshot der coderwall.com  
Startseite am 12.08.2016

Diese Website ist sowohl eine Onlineplattform für Softwareentwickler, als auch ein Aggregator von Informationen von anderen Plattformen. Hauptsächlich von Github, einem sehr beliebten Git Hoster. Entwickler haben auf coderwall die Möglichkeit Tipps und Tutorials, sogenannten „Protips“, zu ihren Spezialgebieten zu veröffentlichen. Diese Beiträge können dann auch wieder mit Schlagwörtern versehen werden und von anderen Benutzern bewertet und kommentiert werden. Wenn der Benutzer seinen Coderwall Account mit seinem Github Account verknüpft hat, kann Coderwall die dort veröffentlichten Git Repositories analysieren. Auf Basis der Beiträge und der verknüpften Account vergibt Coderwall eine Anzahl von fix definierten Badges beispielsweise erhält ein Benutzer Badges dafür, dass dieser eine gewisse Anzahl von Projekten auf GitHub hat oder mehrere Programmiersprachen verwendet. Die Aktivität bei den Protips wird ebenfalls mit Badges und einem Score belohnt.

65 Karma · Joined November 2013 · [muster](#)

---



## Max Muster

Musterland, Mustering · [🌐](#)

[Blog](#)

JavaScript Ruby Open Source CSS CoffeeScript

5 Achievements
6 Protips
0 Comments

---



**WALRUS**

The walrus is no stranger to variety. Use at least 4 different languages throughout all your repos



**FORKED**

Have a project valued enough to be forked by someone else



**CHARITY**

Fork and commit to someone's open source project in need



**MONGOOSE 3**

Have at least three original repos where Ruby is the dominant language



**MONGOOSE**

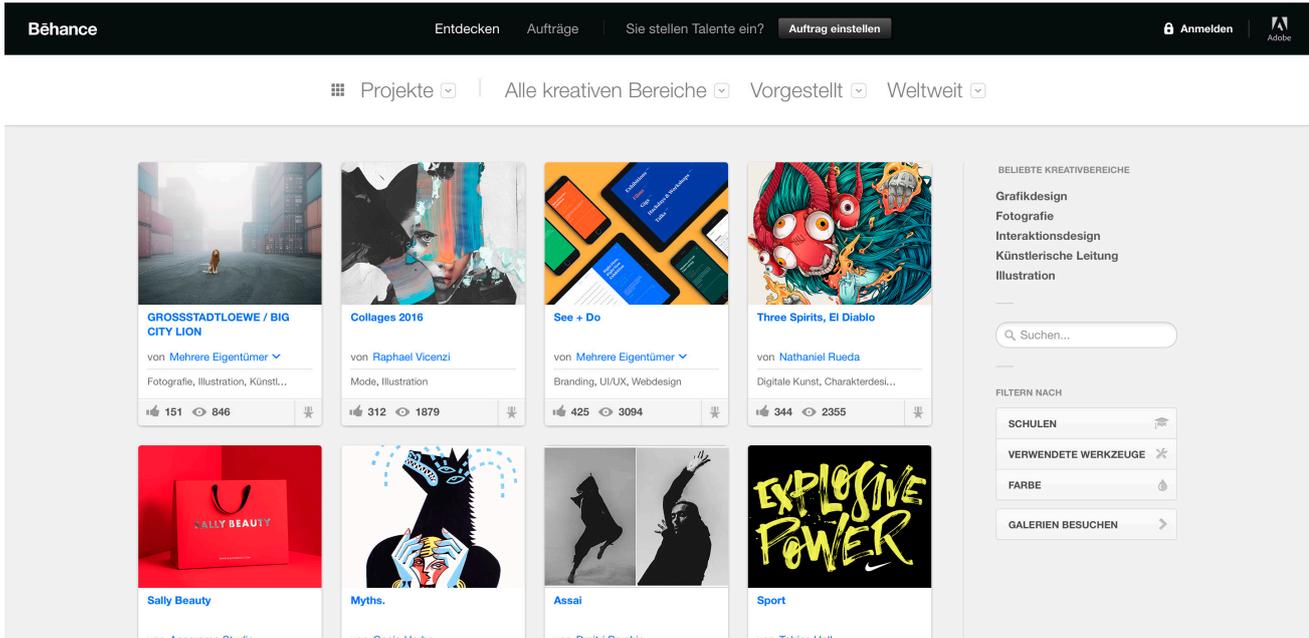
Have at least one original repo where Ruby is the dominant language

Das öffentliche Profil eines Benutzers. Zu sehen ist eine Liste der Achievements, die der Benutzer erarbeitet hat. Zusätzlich kann man sich auch Protips und Kommentare des Benutzers anzeigen lassen.

Mehrere Benutzer können sich auch zu Teams, mit einer eigenen Teamseite auf Coderwall zusammenschließen. Diese Teams werden dann anhand der Aktivität der Teammitglieder mit einer Punktezahl bewertet. Anhand diese Punktwertes kann man sich dann mit anderen Teams messen. Das soll wieder dazu führen, dass Benutzer noch motivierter sind aktiv zu sein.

Coderwall bietet auch eine Jobbörse an, bei der Unternehmen Jobs eintragen können und Programmierer, die auf der Plattform angemeldet sind, sich zu diesen bewerben können. Hier kommen die ganzen Reputationparameter dann wieder zum Zug, indem Sie dem Arbeitgeber die Möglichkeit geben, den potentiellen Programmierer besser und durch die Bewertung von Profis aus dem selben Themengebiet vor allem qualitativer einzuschätzen.

# Behance



Screenshot der behance.net Startseite am 13.08.2016

Ist eine Webplattform zum erstellen von Online Portfolios, die primär auf Künstler und Gestalter abzielt.

Benutzer können in ihr Portfolio sowohl fertige Projekte als auch sogenannten WIPs (Works in Progress / Projekt in Arbeit) stellen. Diese Einträge können dann von den anderen Benutzern angesehen und bewertet bzw. mittels Kommentarsystem diskutiert werden. Ziel der Plattform ist es, dass die Benutzer ihre Arbeit präsentieren können und so potenzielle Kunden, Auftraggeber, Arbeitgeber und Kooperationspartner auf diese aufmerksam werden.

👁 3097  
👍 328  
💬 26

Die Zahl an Ansichten, Appreciations sowie Kommentaren zu einem Projekt.

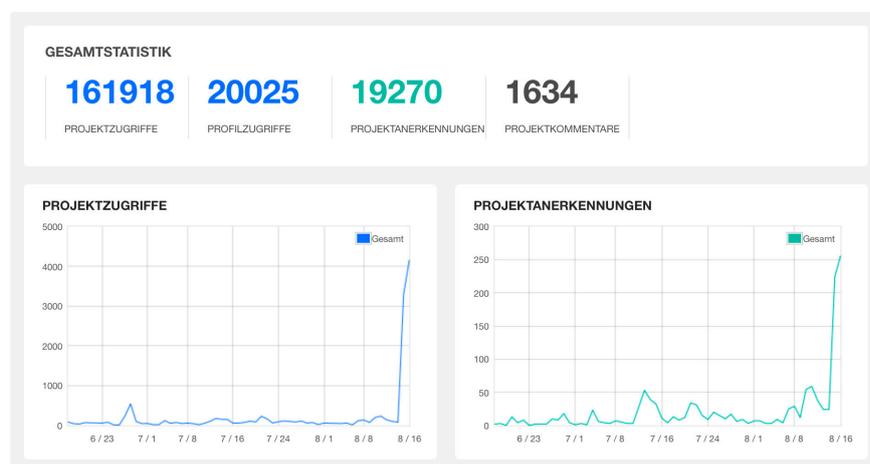
Für Projekte gibt es auf Behance zwei primäre Reputationsparameter. Der erste und explizite Parameter ist die Anzahl an „Appreciations“, die das Projekt von anderen Benutzern erhalten hat. Im Gegensatz zu anderen bereits besprochenen binären Bewertungssystem kann man ein Projekt nicht herunter werten. Es gibt aber die Möglichkeit Projekte zu melden, dies hat aber keine Auswirkungen auf die eigene Reputation und hat nur einen moderativen Zweck.

Der zweite implizite Faktor für die Reputation ist die Anzahl an Ansichten des Projektes durch andere Benutzer. Zusätzlich dazu gibt es auch noch die Anzahl der Kommentare auf das Projekt. Diese wird aber erst auf der Detailseite des eigentlichen Projektes angezeigt.

Die Reputation der Benutzer selbst setzt sich aus der Gesamtzahl der Ansichten aller Projekte sowie der Gesamtzahl der „Appreciations“ aller Projekte zusammen. Zusätzlich gibt es auch noch die Zahl an Benutzern, die dem Benutzer folgen, sowie die Zahl an Benutzern, denen der Benutzer selbst folgt.

Mehrere Benutzer können sich auch in Teams zusammenschließen, für die auch wieder die selben Reputationparameter gelten.

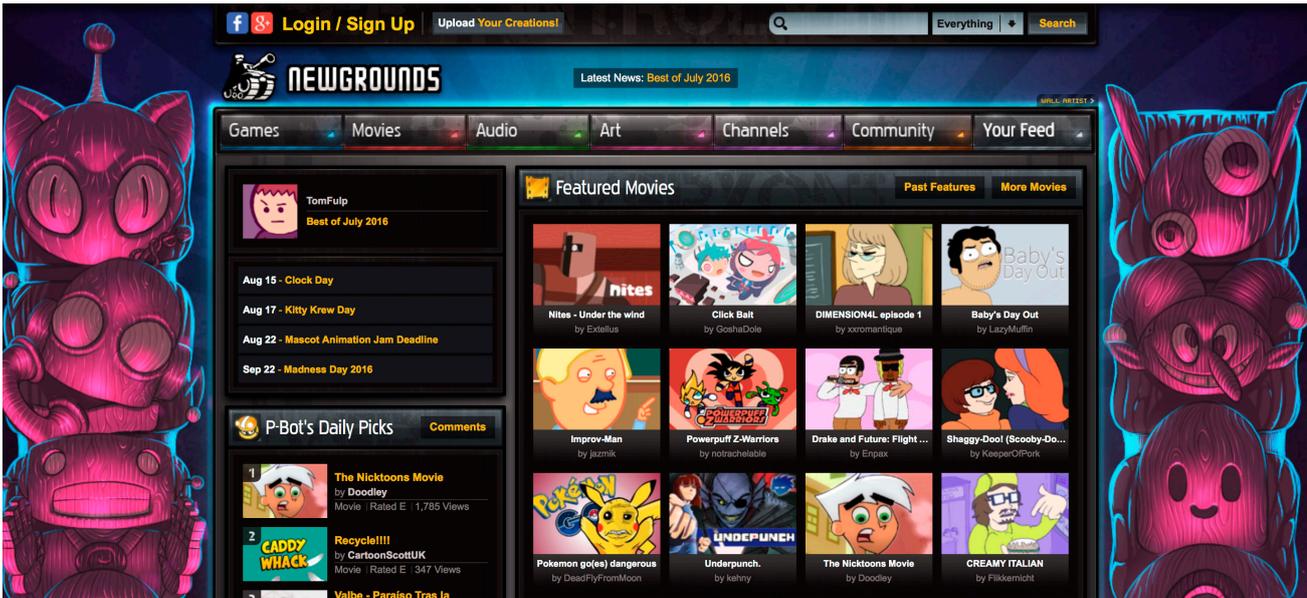
Für Statistik interessierte Benutzer gibt es auch eine genaue Aufschlüsselung der beschriebenen Parameter auf einer zeitlichen Achse und ein Leaderboard der eigenen Projekte in Bezug auf Ansichten, Appreciations und Kommentare.



Die ausführliche Gesamtstatistik zu einem Benutzer.

Behance setzt Reputationmechanismen nur sehr dezent und mit dem primären Ziel ein, gute Arbeiten einfach zu finden und durch Moderatoren auf die Frontpage der Plattform zu stellen. Für die registrierten Benutzer haben die Mechanismen kaum Motivationsfaktor, außer vielleicht möglichst viele Arbeiten zu veröffentlichen um so die Gesamtzahl der Aufrufe und Appreciations möglichst hoch zu treiben. Da es aber keine Möglichkeit gibt diese Zahlen in Form eines Leaderboards mit anderen Benutzern zu vergleichen ist die Motivation vermutlich relativ gering.

# Newgrounds



Screenshot der newgrounds.com Startseite am 13.08.2016



Der Statshot eines newgrounds.com Benutzers.

Die Icons v.l.n.r.:

Level: 7 „Blank Slate“

Whistle: Normal

Rang: „Civilian“

War eine der ersten Webplattformen für das Teilen von mit Flash erstellten Animationen und Spielen. In der aktuellen Version von Newgrounds gibt es auch eine große Bandbreite an Benutzerstatistiken, Trophäen und Medaillen.

Um die Komplexität der Reputation für neue Benutzer etwas leichter erfassbar zu machen, gibt es im Benutzerprofil einen sogenannten „Statshot“. Dieser fasst die wichtigsten Benutzerstatistiken zu drei Werten und dazugehörigen Grafiken zusammen. Einem Level zwischen 1-60, einem „Whistle Status“ sowie einem aus 30 möglichen Rängen.

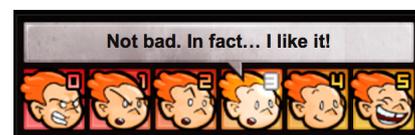
Neben dem numerischen Level, haben die einzelnen Stufen auch noch zur Thematik passende Bezeichnungen. Benutzer können im Level aufsteigen, indem sie Erfahrungspunkte sammeln. Wobei die benötigten Punkte für die nächste Stufe mit jedem Level kontinuierlich ansteigen. Dies ist somit sehr ähnlich dem Levelsystem in vielen RPGs (Rollenspielen). Erfahrungspunkte können von Benutzern verdient werden, indem diese Projekte auf der Plattform bewerten. Es gibt aber eine Beschränkung für den täglich möglichen Zugewinn an Erfahrungspunkten. Somit müssen Benutzer über lange Zeit konstant bewertend aktiv sein, um ein hohes Level zu erreichen. Das Level eines Benutzer wirkt sich vor allem auf die Gewichtung, die sogenannte „Voting Power“, dessen Bewertungen von Projekten aus.

Dies soll vor allem eine Motivation für die Benutzer sein, aktiv Inhalte zu Bewerten, so dass interessante Projekt einfacher gefunden werden können. Das Level ist aber auch ein guter Anhaltspunkt für neue Benutzer, um die Erfahrung oder Aktivität anderer Benutzer einzuschätzen.

Der bereits erwähnte „Whistle Status“ soll anderen Benutzern zeigen, wie gut der betreffende Benutzer dabei ist, fragwürdige Inhalte zu markieren. Das betrifft vor allem Kommentare zu Projekten. Den eigenen Whistle Status kann man verbessern, in dem man einen Beitrag meldet. Wurde ein Beitrag von genug Benutzern gemeldet, wird dieser von einem Moderator überprüft. Waren die Meldungen gerechtfertigt, erhalten alle, die den Beitrag markiert haben Pluspunkte für ihren Whistle Status. Wurde der Beitrag aber zu unrecht gemeldet, verlieren alle Benutzer, die den Beitrag gemeldet haben Whistle Status Punkte. Wenn ein Benutzer zu viele Punkte verliert, geht die Trillerpfeife (Whistle) „kaputt“ und eine Meldung wird nicht mehr zur Gesamtzahl der Meldungen eines Betrags hinzugezählt. Neben der normalen und kaputten Trillerpfeife, gibt es auch noch aufsteigend die Stufen Bronze, Silber, Gold und Göttlich.

Der Rang des Benutzer wird Anhand dessen „Blam“ und „Protection“ Punkte ermittelt. Diese erhält ein Benutzer, wenn er ein Projekt innerhalb der „judgement“ Phase bewertet. Dies ist die Phase zwischen der Veröffentlichung eines Projektes und der ersten 150 erhaltenen Bewertungen. Projekte können mit 0-5 Punkte bewertet werden. Diese werden zwar bei der Ausgabe als Sterne ausgegeben, bei der Eingabe aber als das Maskottchen der Plattform mit unterschiedlichen Gemütszuständen und Aussagen. Das soll dem Benutzer dabei helfen, eine angemessene Bewertung zu finden und hat natürlich auch einen gewissen Unterhaltungsfaktor.

Ein Projekt muss nun in der erwähnten „judgement“ Phase gemittelt mindestens 1.75 Punkte erreichen. Ist die Gesamtbewertung darunter, wird das Projekt wieder gelöscht. Wenn nicht, kann es auf der Plattform bleiben. Wenn nun eine Benutzer ein Projekt mit 0-1 Punkten bewertet hat und dieses tatsächlich gelöscht wird, erhält der Benutzer



5-Punkte Bewertung eines Projektes auf newgrounds.com. 3 Punkte sind ausgewählt und es wird ein dazu passender beschreibender Text angezeigt.

einen „Blam“ Punkt. Wenn der Benutzer ein Projekt mit 2-5 Punkten bewertet und es dieses schafft auf der Plattform zu bleiben, erhält der Benutzer einen „Protection“ Punkt.

Die „Blam“ und „Protection“ Punkte beeinflussen ebenfalls, die bereits beim Level erwähnte, Voting Power eines Benutzers. Dem liegt die Annahme zu Grunde, dass je mehr „Blam“ und „Protection“ Punkte ein Benutzer hat, desto besser ist dieser im Einschätzen der Qualität eines Projektes.

## **Yahoo Design Patterns**

Abschließend möchte ich in diesem Abschnitt noch eine zusammenfassende Beschreibung von möglichen Patterns innerhalb von Reputationsystemen anführen und kurz auf deren spezifische Vorteile und Probleme eingehen. Hierzu werde ich von einer von Yahoo [Yahoo Reputation] erarbeiteten Liste ausgehen und diese gegebenenfalls noch etwas erweitern.

### **Das Kompetitivitätsspektrum**

Einer der wichtigsten Aspekte bei der Gestaltung eines Reputationsystems ist die adäquate Wahl der eingesetzten Reputationmechanismen basierend auf der vorhandenen oder gewünschten Kompetitivität oder Wettbewerbsstreben in der Community der Plattform. Hier gilt es vor allem kompetitive Impulse aus nicht kompetitiven Kontexten herauszuhalten. Yahoo geht hierzu von fünf mögliche Stufen der Kompetitivität aus und weist diesen mögliche Reputation Patterns zu:

Hilfsbereit

Kollaborativ

Aufrichtig

Kompetitiv

Kämpferisch

### Ziel

Benutzer werden dadurch motiviert, sich gegenseitig zu helfen

Benutzer haben ähnliche Ziele und arbeiten zusammen.

Jeder Benutzer hat seine eigenen intrinsischen Ziele, diese müssen sich aber nicht widersprechen

Benutzer haben die selben Ziele, müssen aber gegeneinander handeln um diese zu erreichen

Benutzer haben widersprechende Ziele, um das eigene Ziel zu erreichen, können andere ihres nicht erreichen

### Reputation für

Identifizieren von erfahrenen und/oder beliebten Benutzern

Hervorheben von vertrauenswürdigen Benutzern

Darstellung der Aktivität, so dass sich Benutzer besser gegenseitig einschätzen können

Präsentieren von erreichten Zielen und Leistungen um so Respekt und Anerkennung zu gewinnen

Anzeigen von Leistungen, Siegen und Verlusten als Basis für Angeberei

### Darstellung

„Identifying Labels“ für Benutzer z.B. „Hilfreich“, „Top-User“, „alter Hase“, ...

„Named Levels“ um die Geschichte und Status eines Benutzers darzulegen. Je höher der Rang desto Vertrauenswürdiger.

Eine statistische Auswertung und Aufbereitung der Aktivitäten. Die Community soll anhand der offenen Daten selbst entscheiden.

Einfaches Vergleichen mit „Numbered Levels“ und optional noch eine zusätzliche Motivation durch „Collectible Achievements“

Benutzer können ihren eigenen Fortschritt anhand von Punkten verfolgen. Benutzer sind anhand eines direkten „Rankings“ vergleichbar.

### **Named Levels**

Bei diesem Pattern definiert man eine Menge von stetig in ihrer Wertigkeit steigenden Stufen (Levels), denen man beliebige unterscheidende Namen zuweist. Ein Problem bei diesem Verfahren ist, dass der direkte Vergleich zwischen den Stufen schwer ist, selbst wenn die Namensgebung einem gewissen Muster folgt oder eine vorhandene Hierarchie abbildet.

Abgesehen vom Vergleich mit anderen Benutzern, kann der Benutzer laut Yahoo mit „*Named Levels*“ auch den eigenen Fortschritt und Status in der Community besser abschätzen. Dieses Pattern kann Benutzern auch dabei helfen, andere vertrauenswürdige Benutzer zu identifizieren. Somit sind „*Named Levels*“ vor allem für kollaborative bis leicht kompetitive Communities geeignet.

„*Named Levels*“ werden oft in Online Foren eingesetzt, um die Erfahrung der einzelnen Benutzer darzustellen, wobei diese auch öfters noch mit „*Identifying Labels*“ kombiniert werden um Benutzer mit Sonderrollen wie Moderatoren und Administratoren hervorzuheben.

### **Numbered Levels**

Im Gegensatz zu den „*Named Levels*“ sind die „*Numbered Levels*“ einfach stetig aufsteigende Nummern. Der Vorteil dieser Variante von Stufen ist, dass ein direkter Vergleich möglich ist.

Es wäre natürlich auch möglich, die beiden Level Varianten zu kombinieren und numerische Stufen zu den „*Named Levels*“ zur Verfügung zu stellen. Ähnlich wie die „*Named Levels*“ helfen „*Numbered Levels*“ Benutzern vor allem dabei, den eigenen Fortschritt in der Community, sowie die Erfahrung anderer Benutzer einzuschätzen. Diese Variante der Levels ist aber eher für kompetitive Communities geeignet, da der Reihungscharakter mehr in den Fokus gerückt wird.

„*Numbered Levels*“ sind meist in RPG Computerspielen wie zum Beispiel Blizzards World of Warcraft anzutreffen. Manche Communities, wie zum Beispiel Newgrounds, kombinieren „*Numbered Levels*“ mit „*Named Levels*“.

## Identifying Labels

Bei diesem Pattern geht es darum gewisse Gruppen, Verhalten oder Qualitäten eines Benutzers innerhalb in einer Community explizit mit einem Namensschild hervorzuheben. Diese haben aber anders als „*Named Levels*“ keine steigende beziehungsweise vergleichbare Wertigkeit. Somit können neue oder weniger erfahrene Benutzer leicht *besondere* Benutzer finden. Auf Twitter gibt es etwa die Möglichkeit seinen Account zu verifizieren. Dieser erhält dadurch eine besondere Kennzeichnung, die besagt, dass der Benutzer die reale Person ist, die dieser vorgibt zu sein. Mit diesem Verfahren kann man vor allem gewolltes Verhalten in den Vordergrund rücken, in dem man Benutzer für dieses mit Labels belohnt. So können Benutzer auch dazu motiviert werden, freiwillig Aufgaben innerhalb der Community zu übernehmen wie z.B. als Moderator in einem Forum. Dieses Pattern kann laut Yahoo über das volle kompetitive Spektrum eingesetzt werden.

## Points

Hierbei handelt es sich um das direkte Anzeigen der kumulativen Punkte, die innerhalb der Community verdient wurden. Dies hat vor allem den Sinn, die Gesamtleistung, im Gegensatz zu den Einzelleistungen, eines Benutzers zu belohnen. Wie diese Punkte erarbeitet werden, ist abhängig von der jeweiligen Community oder Plattform. Dieses Pattern ist vor allem für sehr kompetitive Communities geeignet. Allgemein sollte man möglichst auf Systeme mit Punktebewertungen verzichten, außer diese sind ein elementarer Bestandteil der Plattform. Welche Aktionen wie viele Punkte wert sind, muss gut ausbalanciert und wohl überlegt sein um gewisse erwünschte Tätigkeiten nicht ihrer Wertigkeit und somit Motivation zu berauben. Man sollte mit Punkten möglichst besondere Errungenschaften oder Verhalten belohnen und nicht einfach nur Aktivität, um so etwa Grinding (wiederholtes durchführen einfacher Aktionen, mit dem Ziel möglichst viele Punkte über längere Zeit zu generieren) zu verhindern. Punkte können auch mit Levels kombiniert werden, so dass die Level die eigentliche Reputationinformation sind und die Punkte nur eine zusätzliche Detailinformation um z.B. zu sehen wie viele Punkte man noch zum nächsten Le-

vel benötigt. Stackoverflow etwa verwendet so ein System, in dem man Punkte sammelt und in den drei Stufen Bronze, Silber und Gold aufsteigen kann. Ein weiteres Beispiel wäre die Verkäuferbewertung auf eBay, bei der man für gewisse Punktestufen eine bestimmte Grafik neben seinem Namen angezeigt bekommt.

### **Collectible Achievements**

Sind im Grunde Auszeichnungen die man für definierte Ziele erhält. Diese haben laut Yahoo vor allem den Sinn, dass Benutzer Teile des Angebots ausprobieren oder entdecken, welche diese sonst ignoriert oder nicht zu Gesicht bekommen hätten. Coderwall etwa bietet so ein Badge System an. Dieses ist sogar über mehrere Plattformen verteilt und soll dazu anregen deren Angebote auszuprobieren und zu verwenden. Wenn Achievements gut gestaltet sind, können diese einen gewissen Sammeltrieb bei den Benutzern auslösen, wobei sich das bei unpassend gestalteten Achievements auch konträr dem gewünschten Effekt wirken könnte. Auch die visuelle Aufmachung dieser Achievements kann über den Erfolg oder Misserfolg bei den Benutzern entscheidend sein.

Ähnlich, wie bei den Punkten, sollte man Benutzer eher für besondere Leistungen und nicht nur für simple Aktivität mit Trophäen, Badges o.Ä. belohnen. Auch muss man auf eine gute Durchmischung von Schwierigkeitsgraden achten. Ein weiterer wichtiger Faktor bei Achievements ist, dass diese im Vorhinein inklusive der Bedingungen einsehbar sein sollten, so dass der Benutzer weiß, dass es diese überhaupt zu verdienen gibt und was er dazu tun muss.

### **Ranking**

In Kombination mit *Points* aber auch *Levels* kann man diese Punkte oder Stufen auch in einer zusammenfassenden Ansicht für einen besseren Vergleich darstellen. Yahoo definiert hier zwei mögliche Varianten eines sogenannten *Rankings*:

- **Leaderboard**

Ist eine simple Liste alle Benutzer sortiert nach ihren Punkten. Yahoo rät auch dazu unterschiedliche z.B. zeitlich eingegrenzte Ansichten anzubieten, da ein all-

gemeines *Leaderboard*, welches die gesamte Lebenszeit einer Community umfasst, relativ schnell einfrieren kann und sich nicht mehr allzu stark ändert bzw. neue Benutzer kaum noch Chancen haben, an die Spitze zu kommen.

- **TopX**

Anders als bei einem simplen *Leaderboard* werden bei dieser Variante Benutzer anhand ihres Rankings gruppiert und so etwa in die Top 10, Top 100 usw. zusammengefasst. Auch hier sind zeitlich eingegrenzte Listen sinnvoll.

Im Großen und Ganzen rät Yahoo eher davon ab *Leaderboards* zu verwenden, da diese meist kontraproduktiv in einer Community sind. Auch ein System so zu bauen, dass man sich keine Topplatzierung erschlummeln kann ist nahezu unmöglich. Somit hätten faire Benutzer der Plattform kaum eine Chance eine gute Platzierung zu bekommen, obwohl man deren Verhalten eigentlich fördern möchte. Ein *TopX* System andererseits hat einen ähnlichen gewünschten Effekt. Benutzer werden ebenfalls dazu motiviert Aktionen durchzuführen um Punkte zu erlangen, das Ranking ist aber dennoch unscharf genug um nicht allzu sehr zu einem Austricksen des Systems herauszufordern. Wobei es aber selbst bei Plattformen wie Amazon bei den Top-Reviewern Probleme mit diesem System gibt.

## **Reputation Modelle**

Farmer & Glass definieren in Ihrem Buch „Web Reputation Systems“ [Farmer & Glass] eine Sammlung von typischen einfachen Reputation Modellen, die meistens dann zum Einsatz kommen, wenn es darum geht eine Web Plattform um Reputationmechanismen zu bereichern.

## **Favoriten und Markierungen**

Markierungen und Favoriten werden dazu verwendet, den Benutzern der Plattform die Möglichkeit zu geben, Ausreißer zu markieren. Sowohl im positiven, wie auch im negativen Sinn. Dies können sowohl explizite Elemente wie *Like* oder *Melden* Schaltflächen sein, aber auch implizite, wie z.B. ein in die Plattform integriertes Lesezeichen System. Diese Markierungen können dann wiederum von der

Plattform dazu genutzt werden, die angebotenen Inhalte zu optimieren oder gewünschtes und unerwünschtes Verhalten von Benutzern zu erkennen.

### **Dies-oder-Das Wahlen**

Hierunter verstehen Farmer & Glass das Auswählen zwischen zwei oder mehreren Möglichkeiten durch die Benutzer. Sie geben etwa das Markieren von hilfreichen Beiträgen als ein mögliches Beispiel an, wobei in diesem Fall wohl die Wahlmöglichkeiten *hilfreich* oder *nicht hilfreich* je Beitrag gemeint sind. Grundsätzlich würde ich dieses Modell eher als Sonderfall dem vorhergehenden Modell der *Favoriten* zurechnen. Laut Farmer & Glass besteht der Unterschied vor allem in der Menge der markierbaren Inhalte innerhalb eines Kontext. So sind diese etwa auf einer Plattform wie Stackoverflow oder Amazon innerhalb einer Diskussion oder in Bezug auf ein Produkt stärker auf einen Kontext begrenzt und die Wahl eines hilfreichen Beitrages spezifischer. Im Gegensatz dazu, sind Favoriten oder Markierungen eher im Kontext der gesamten Community oder Plattform gültig.

### **Bewertungen**

Hierunter versteht man Fünfstern Bewertung und andere ähnliche Systeme, die es den Benutzern ermöglichen, einen Inhalt über einen gewissen Bereich von Wertigkeiten zu beurteilen.

### **Kritiken (Reviews)**

Ist eine Sammlung von mehreren Bewertungen, die zusammengefasst eine komplexere Bewertung des selben Objektes durch einen Benutzer ergeben. Dies können sowohl mehrere numerische Bewertungen in vorgegebene Kategorien sein, wie z.B. bei der Verkäuferbewertung auf eBay oder auch ein freier Text, bei dem der Benutzer selbst eine Struktur und Bewertungsmetriken vorgibt.

### **Punkte**

Hierunter versteht man das Sammeln von Punkten durch Benutzer, die durch das Durchführen bestimmter Aktionen auf der Plattform verdient werden können. Dieses Model ist nur sehr schwer richtig einzusetzen und sollte

nur mit Vorsicht verwendet werden, da man Aktionen auf der Plattform gewisse Wertigkeiten zuweisen muss und so sehr direkt das Verhalten der Benutzer beeinflussen kann. Auch muss man vorsichtig sein, wenn man Punkte sowohl für Karma als auch als Zahlungsmittel einsetzen möchte. Wenn hierzu die selben Punkte eingesetzt werden, kann es leicht passieren, dass die eigentlichen *guten* Benutzer, die Punkte sammeln und auch ausgeben, weniger hervorstechen als Punkte hortende Benutzer.

## **Karma**

Dies ist, wie bereits bei den Begriffsdefinition erwähnt, die Reputation eines einzelnen Benutzers bzw. einer einzelnen Person.

Farmer & Glass unterscheiden zwischen zwei grundlegenden Karma Modellen. Einmal dem sogenannten *Mitwirkungs-Karma* (Participation Karma), welches dazu dient, die Aktivität eines Benutzers innerhalb der Community abzubilden. Die zweite Grundform ist das *Qualitäts-Karma* (Quality Karma), dass man dazu einsetzt die Qualität der Beiträge eines Benutzers darzustellen.

Das *Mitwirkungs-Karma* wird meist in Form eines Punktesystems implementiert, bei dem man Punkte sammeln kann, indem man bestimmte Aktionen durchführt.

Das *Qualitäts-Karma* ergibt sich im besten Fall indirekt durch die Bewertung von Inhalten oder Aktionen des Benutzers durch andere Benutzer.

Alleine ist vor allem das *Mitwirkungs-Karma* wenig aussagekräftig. Um dem entgegen zu wirken sollte man beide Grundformen kombinieren. Hierbei sprechen Farmer & Glass von einem *Robustem Karma* (Robust Karma). Dieses Karma erlaubt dann sowohl einen Einblick in die Quantität der Beiträgen als auch in deren Qualität bzw. Relevanz für die Community.

## **Kombinationen**

Die beschriebenen Reputation Modelle werden in den seltensten Fällen alleinstehend implementiert, sondern werden meist in kombinierter Form eingesetzt. Die Plattform eBay setzt beispielsweise auf eine Kombination von Kritiken, Bewertungen und Karma. Amazon hat zusätzlich dazu auch noch eine Dies-oder-Das Wahl ob ein Kommen-

tar hilfreich war. Die Diskussionsplattform Stackoverflow setzt beispielsweise auf ein Karma System mit Punkten in Kombination mit Dies-oder-Das Wahlen.

## Gamification

In diesem Kapitel werde ich einige Anwendungsbeispiele von Gamification vorstellen und versuchen auf deren Vor- und Nachteile einzugehen.

Wie bereits einleitend in den Definitionen erwähnt, handelt es sich bei Gamification um das Anwenden von Game Design Prinzipien im Designprozess von Artefakten, welche nicht unbedingt mit einem Spiel in Verbindung stehen. Was man sich primär davon erhofft, ist ein Artefakt, die es schafft die Benutzer, ähnlich wie ein Spiel, in dessen Bann zu ziehen und zu motivieren.

Andererseits gibt es natürlich berechtigte Bedenken ob dies überhaupt funktioniert oder etwa sogar negative Effekte hat, in dem die spielerischen Elemente von dem eigentlichen Zweck des Artefaktes ablenken.

Mit der Wirkung von Gamification in einem firmeneigenen Sozialen Netzwerk (SNS) haben sich [Thom et al, 2011] beschäftigt. Hierzu haben Sie zuerst das SNS eines Unternehmens mit international verteilten Standorten beobachtet, in dem bereits Gamification Elemente implementiert waren. Bei diesen handelte es sich primär um ein Punktesystem, bei dem Benutzer Punkte für Beiträge und andere Aktivitäten erhalten. Nach dieser ersten Beobachtung wurden die Gamification Elemente entfernt und erneut das Verhalten der Benutzer im SNS beobachtet.

Thom et al. konnten im zweiten Beobachtungsdurchgang einen Rückgang der allgemeinen Aktivität beobachten sowie eine Halbierung der Menge an geschriebenen Beiträgen. Vor allem die Kommunikation mit Kollegen bei weiter entfernten Standorten hatte gelitten. Auch die Zahl der Seitenaufrufe pro Benutzer ist zurückgegangen, da diese zuvor öfter durch neue Kommentare bei Diskussionen auf die Plattform zurück gelockt wurden.

Mit Gamification in einem schulischen Kontext haben sich [Abramovich et al.] beschäftigt. Genauer gesagt mit der Nützlichkeit von Badges im Unterricht beziehungsweise in der Lehre. Sie erkennen Badges vor allem als formalisierte Form der Anerkennung. Weiter können Badges laut

Abramovich et al. den Schülern dabei helfen, einen besseren Überblick über alle Aufgaben zu erlangen, sowie dabei unterstützen, sich im Vorhinein Ziele zu stecken.

Abramovich et al. beschreiben in ihrem Paper, dass Badges kaum etwas an der eigentlichen Motivation der Schüler geändert haben. Andererseits ist aber die Angst davor gesunken, als nicht fähig wahrgenommen zu werden und somit auch die Angst davor neue Aufgaben anzugehen.

Auch konnte ein steigendes Interesse in das Fach, in dem die Badges zum Einsatz gekommen sind, erkannt werden. Eine weitere wichtige Erkenntnis des Papers ist, dass unterschiedliche Badgetypen, unterschiedliche Lerntypen besser oder schlechter ansprechen. So muss man laut Abramovich et al, wenn man Badges einsetzen möchte, darauf achten, ein gutes Gleichgewicht zwischen auf Fähigkeit und Anwesenheit basierten Badges zu finden.

Ein weiteres Beispiel für die Anwendung von Gamification wird von [Fitz-Walter et al, 2011] beschrieben, welche eine mobile Applikation gestaltet haben, die frischen Studenten dabei helfen soll, sich auf dem Campus zurechtzufinden und vor allem auch dazu motivieren soll, aktiv den Campus zu erkunden. Dies gelingt mit Hilfe von Achievements, welche die Studenten ergattern können, in dem Sie gewisse Aufgaben, mit einem steigenden Schwierigkeitsgrad lösen. Fitz-Walter et al. haben stark darauf geachtet, dass die Gamification möglichst nicht die nützlichen Elemente der App (Karten, Termine, ...) maskiert beziehungsweise von diesen ablenkt oder gar einschränkt.

Laut den Autoren wurde die Applikation gut angenommen und vor allem Aufgaben mit kontextbasierten Eingaben kamen sehr gut an.

[Koch & Ott, 2012] definieren eine Liste der gängigsten eingesetzten Gestaltungselemente im Rahmen der Gamification. Einige dieser Punkte überschneiden sich auch mit typischen Elementen von Reputationsystemen, was durchaus zeigt, wie nahe diese beiden Gebiete einander sind.

- **Sichtbarer Status**

Hiermit ist die Darstellung des Status des Spielers beziehungsweise Benutzers gemeint. Gängige Methoden

hierfür sind etwa Badges oder Level Stufen. Diese Mittel ermöglichen vor allem einen Vergleich mit anderen Benutzern und fördern somit den Wettbewerb.

- **Einsehbare Rangliste**

Ähnlich wie bei einem sichtbaren Status werden die Information zu Benutzern direkt nebeneinander gestellt und gereiht und somit der Wettbewerbscharakter noch einmal verschärft.

- **Quests**

Hierunter verstehen Koch & Ott entdeckbare Aufgaben, welche klar definierte Ziele und eventuelle Zusatzziele haben und auch an einen bestimmten Zeitrahmen gebunden sein können.

- **Resultatstransparenz**

Hiermit ist gemeint, dass dem Spieler vermittelt beziehungsweise offengelegt wird, welche Auswirkungen sein Handeln haben wird und dieser somit im Vorhinein weiß, was dessen Entscheidungen bewirken werden.

- **Rückmeldung**

Oder auch „Feedback“, versteht sich als eine unmittelbare Rückmeldung auf das Verhalten des Benutzers. Wobei es aber dem Benutzer freisteht, wie er mit diesem Feedback umgeht.

- **Epic Meaning**

Meint, dem Spieler das Gefühl zu geben an einem größeren, „epischen“, Ziel zu arbeiten. Optional auch mit anderen Mitspielern.

- **Fortschrittsanzeige**

Eine visuelle Repräsentation des Fortschritts innerhalb einer gewissen Aufgabe. Wobei sowohl das bereits Erreichte, als auch das noch zu Erreichende dargestellt wird.

- **Community Collaboration**

Das gemeinsame Erreichen von Zielen oder lösen von Aufgaben. Durch die soziale Komponente hat dieses Gestaltungselement einen sehr motivierenden Charakter.

- **Cascading Information**

Hierbei geht es darum, dem Benutzer nur die, für die Lösung der aktuellen Aufgabe nötigen Informationen zu geben.

## Interviews

Als Teil der Recherchen wurde eine Reihe von Interviews mit Lehrenden durchgeführt, um ein Gespür für wichtige Faktoren und Features eines Designs für ein Reputationssystem zu bekommen. Primär sollte herausgefunden werden, inwieweit bereits versucht wird die Studenten zu außergewöhnlichen Leistungen zu motivieren und wie gut diese womöglichen Versuche angenommen werden und wie Probleme gelöst wurden oder nicht gelöst werden konnten. Mit den Interviews sollte auch ermittelt werden, ob überhaupt ein zusätzliches System wünschenswert ist und was die Ansprüche an dieses wären.

Als Methode wurde ein Leitfadeninterview gewählt um flexibler auf den, vor dem Interview weitgehend unbekanntem, Status Quo der Lehrenden eingehen zu können.

## Leitfragen

Um die Leitfadeninterviews durchführen zu können, wurden die folgenden acht Fragen mit der Zielsetzung ausgearbeitet, die wichtigsten Punkte, die in Erfahrung gebracht werden sollten abzudecken, aber dennoch genügend Freiraum für die Entwicklung eines Gesprächs zu lassen.

*Halten Sie Noten für ein geeignetes Mittel, die Qualifikationen von Studierenden zu repräsentieren? (z.B. bei Bewerbungen, Vergleichbarkeit zwischen Lehrveranstaltungen mit selbem Inhalt, ...)*

Diese Frage wurde als erste gewählt, um direkt eine Diskussion über die Art der Bewertung von Leistungen der Studenten anzuregen und so eingehend eine Basis für die weiteren Fragen zu schaffen oder diese bereits zum Teil zu beantworten.

*Wie bestimmen Sie die Endnoten Ihrer LVA Teilnehmer?*

Um einen eventuellen Übergang zu ermöglichen, bleibt diese Frage bei der Notenthematik. Mit dieser Frage sollte herausgefunden werden, inwieweit Lehrende den Studenten Möglichkeiten geben Ihre Endnote zu beeinflussen und welche Faktoren in eine Note komprimiert werden. Auch

eine Diskussion über den Umgang mit eventuellen Zusatzleistungen und ob diese auf der Strecke bleiben können und nicht direkt in die Note einfließen, sollte mit dieser Frage eingeleitet werden.

*Kennen Sie Fälle bzw. haben Sie Fälle erlebt, in denen außergewöhnliche Studierende nur mittelmäßige Noten bekommen haben (oder umgekehrt). Was tun Sie in solchen Fällen?*

Diese Frage schließt direkt an die Thematik der Notenvergabe an und soll, falls nötig, das Gespräch zu der Problematik führen, dass Noten oft nur die Leistung zu einem einzigen oder einigen wenigen Zeitpunkten bewerten. Es soll auch ermittelt werden, wie Lehrende mit solchen Extremsituationen umgehen, beziehungsweise wie viel Spielraum sie sich selbst in der Notenvergabe lassen.

*Was tun Sie, um außergewöhnliche Leistungen in ihren LVAs zu fordern/fördern?*

Mit dieser Frage soll eine Diskussion darüber ins Rollen kommen, wie man Studierende zu Mehrleistungen motivieren kann beziehungsweise in welche Form dies bereits geschieht.

*Verwenden Sie über die Bewertung durch die Note hinausgehende Anreize oder Belohnungen, um die Studierenden zu motivieren?*

Diese Frage soll, falls dies nicht bereits zu Sprache gekommen ist klären, welche Mittel die Lehrenden in ihren Lehrveranstaltungen einsetzen um ihre Studierende extrinsisch zu motivieren und wie diese Motivatoren angenommen werden.

*Können die Studierenden das verwenden, um ihre Qualifikation (z.B. bei Bewerbungen) dann auch nachzuweisen?*

Dies ist wiederum eine Anschlussfrage an die Vorhergehende. Wobei hier vor allem herausgefunden werden soll, ob es bereits gewisse interne oder auch externe Plattfor-

men gibt, mit denen Zusatzaufgaben auch für Dritte einsehbar wären und falls nicht, warum.

*Würden Sie ein System verwenden wollen, das genau das ermöglicht?*

*Welche Möglichkeiten müsste Ihnen so ein System bieten bzw. welche Anforderungen hätten Sie an ein solches?*

Die letzten beiden Leitfragen sind wieder Anschlussfragen die sich zum Großteil wohl schon durch den bisherigen Gesprächsverlauf geklärt haben sollten. Falls nicht, soll mit diesen noch einmal der Fokus auf die Problemstellung dieser Arbeit lenken und gezielt Input für den späteren Designprozess liefern.

Es wurden sechs Lehrende der TU Wien aus unterschiedlichen Bereichen der Informatik interviewt. Die unterschiedlichen Bereiche waren wichtig, da die Inhalte der Lehrveranstaltungen in gewissem Maße auch vorgeben, wie diese abgehalten werden, da man etwa Inhalte mit einem Fokus auf praktische Designerfahrung nur schwer auf die gleiche Art wie mathematisch theoretische Inhalte vermitteln kann. Auch gibt es andere Möglichkeiten und Rahmenbedingungen, wie man Studenten zu außergewöhnlichen Leistungen motivieren kann.

## **Ergebnisse**

Es wurden insgesamt sechs Lehrende aus den Bereichen Algorithmen, Programmierung, Informationsvisualisierung und Theoretische Informatik interviewt, welche alle vorwiegend reine Übungen oder Vorlesungen mit Übungen mit unterschiedlich großen Teilnehmerzahlen betreuen. Zusammengefasst aus den Gesprächen haben sich die folgenden Erkenntnisse ergeben:

Für die Verwaltung der Übung wird meist ein eigenes internes System oder die von der TU Wien angebotene e-Learning Plattform TUWEL eingesetzt. Wobei es sich bei letzterer schwierig gestaltet die Informationen zur Leistung nach dem Ende der Lehrveranstaltung aus dem System wieder herauszubekommen um diese anderweitig zu verwenden. Andererseits bietet TUWEL die Möglichkeit eines

Dialogs mit den Studierenden außerhalb der Sprechstunden und regulären Terminen und kann auch dazu verwendet werden Punkte für gute Mitarbeit zu vergeben. TUWEL bietet aber für diese Form der Punktevergabe kein eigenes dafür gedachtes System an.

Für die Kommunikation mit den Studierenden wird alternativ auch das „informatik-forum.at“ eingesetzt, wobei es sich bei diesem um ein, von Studenten organisiertes, online Forum handelt. Da es sich hier um eine von den Interviewten durchaus erwünschte Form der Selbstorganisation von Studierenden handelt, greifen diese auch nicht moderierend ein, sondern beteiligen sich, wenn überhaupt, nur manchmal in Form von Hilfestellungen zu Übungsaufgaben. Da die Lehrenden somit auch keinen vollständigen Überblick darüber haben, wie sehr sich ihre Studenten im Forum einbringen, wird deren Aktivität auch nicht für die Benotung herangezogen.

Ein wichtiger Faktor beim Dialog mit den Studierenden war den Interviewten vor allem eine qualitativere Form des Feedbacks zur Lehrveranstaltung selbst, sowohl als auch eine Verbesserung der Lehre durch das Einbinden der Studierenden in den Übungsablauf. Auch sind sie der Meinung, dass die Kommunikation zwischen den Studierenden auf Plattformen wie TUWEL oder dem informatik-forum, positive Lerneffekte durch gemeinsames Erarbeiten, Kooperation und Hilfe haben.

Um Studierende zu motivieren, greifen die Interviewten gerne auf eine interessante und möglichst praxis- und zeitnahe Gestaltung der Übungsaufgaben zurück. In Übungen mit kleineren Teilnehmerzahlen wird den Studierenden auch gerne etwas mehr Freiraum eingeräumt, so dass diese gewisse Aspekte der Aufgabe selbst bestimmen können. In manchen Lehrveranstaltung gibt es sogar die Möglichkeit in deren Kontext an externen Wettbewerben teilzunehmen.

Ein weiteres beliebtes Mittel zur Motivation der Studenten, aber auch zur Verbesserung der Lehre, ist das Präsentieren der besten oder auch besonderer Arbeiten, von an der Lehrveranstaltung teilnehmender Studenten, in einem

eigens dafür eingerichteten Blog, während der Lehrveranstaltung selbst oder auf einer eigenen Hall of Fame Seite, welche auch nach Abschluss der Lehrveranstaltung noch existiert und einsehbar ist. In manchen Lehrveranstaltungen werden die Studierenden dann auch noch bei einer weitergehenden Publikation ihrer herausragenden Arbeiten unterstützt.

Viele der abgehaltenen Übungen werden mittels Punkten benotet, die man mit, über das Semester verteilten, Aufgaben erarbeiten kann. Manche der Interviewten geben den Studierenden zusätzlich die Chance durch Zusatzaufgaben weitere Punkte zu ergattern, um somit die Endnote zu verbessern, was aber eher dazu dient verlorene Punkte durch verpatzte Aufgaben zu korrigieren, da die Punkte am Ende immer noch nach einem fixen Schlüssel auf Noten abgebildet werden.

Was die Benotung der Leistungen der Studierenden betrifft, werden mehrere kleine Tests einem großen Test vorgezogen, nur leider ist dies wegen der meist größeren Teilnehmerzahlen, auf Grund der begrenzten Ressourcen problematisch bis nicht möglich.

Eine Art Reputationsystem wäre für die meisten der interviewten Personen durchaus interessant, nur müsste dieses vor allem kaum bis keinen Zusatzaufwand im Gegensatz zu den derzeit eingesetzten Methoden bedeuten, da es bereits jetzt schon an den nötigen Arbeitskräften mangelt. Es wurden auch Bedenken bezüglich des Austricksen eines solchen Systems geäußert, sowie, dass auch die Möglichkeit einer Selbstorganisation der Studierenden gegeben sein sollte.

## **Badges Prototyp**

Da das Implementieren eines Prototyps für eine komplette Reputationinfrastruktur den Rahmen dieser Arbeit gesprengt hätte, sowie ein Fokus auf dem Einsatz von Badges lag, wurde ein Prototyp einer Online Plattform zum Vergeben von digitalen Badges gestaltet und implementiert. Diese Plattform könnte dann auch als Teil in eines umfassenden Reputationsystem eingesetzt werden.

Bei diesem Badgeprototyp handelt es sich um eine einfache Webseite, welche es Lehrenden erlaubt, Badges zu erstellen und diese an Studierenden zu vergeben. Zusätzlich wird eine Wertigkeit dieser Badges berechnet.

In diesem Teil der Diplomarbeit werde ich zuerst im Abschnitt Design, anhand einer Beschreibung des grafischen Userinterfaces des Prototypen, die allgemeinen Überlegungen erläutern, die in die Gestaltung des Prototypen eingeflossen sind und wie diese schlussendlich implementiert wurden. Im zweiten Abschnitt, zur technischen Implementation, werde ich vor allem die technischen Aspekte des Prototypen erläutern, welche wiederum auch als technische Dokumentation für spätere Weiterentwicklungen und Adaptionen des Prototypen dienen sollen.

Einleitend werde ich auch noch kurz darauf eingehen was denn nun ein Badge ist und diesem Prototypen ähnliche Projekte erläutern.

### **Was ist ein Badge?**

Grundsätzlich sind Badges nichts anderes als eine explizite Aussage oder Auszeichnung von Person A zu einer von Person B erbrachten Leistung C. Diese Form der Anerkennung kennt man natürlich auch in der realen Welt etwa in Form von Abzeichen, zu Englisch „Badges“, die z.B. bei den Pfadfindern auch als Aufnäher an das Uniformhemd genäht werden können.

Diese visuelle Repräsentation wurde kurzerhand in die digitale Welt übernommen, der Aufnäher durch eine Grafikdatei ersetzt und das Uniformhemd durch das Profil einer Person auf der Online Plattform.

## Ähnliche Projekte

Bevor ich weiter auf die Details des Designs des Badgeprototypen eingehe sei erwähnt, dass es verständlicherweise bereits mehrere Bestrebungen gibt, Systeme zur Verfügung zu stellen, die es ermöglichen Badges zu vergeben. Ich werde nun auf drei solcher Systeme eingehen, um so eine Basis für das darauf folgende eigene Design eines solchen Systems zu schaffen.

### Open Badges

Ist ein von Mozilla entwickelter Standard um digitale Badges vergeben und von unterschiedlichen Quellen zentral sammeln zu können. Der eigentliche Badge ist im Grunde eine simple Textdatei, eine sogenannte „Assertion“, mit in der Spezifikation vorgegebenen Feldern und Werten, die nach der JSON Spezifikation formatiert sind. Diese Felder enthalten unter anderem die Informationen wer von wem, welchen Badge für was erhalten hat. Benutzer können diese Assertions dann in sogenannten „Backpacks“ unabhängig von dem Badges ausstellenden Service oder Plattform sammeln und präsentieren.

Zusätzlich dazu gibt es auch noch sogenannte „Baked Badges“. Hierbei handelt es sich um PNG Bilddateien in deren Dateihader die Assertion eingebettet ist oder um SVG Dateien mit einem speziellen XML Tag der die entsprechende Assertion enthält.

Dadurch ist das System relativ flexibel und Badges können zwischen unterschiedlichen Systemen leicht herumgereicht werden. Diese Services können in zwei Kategorien fallen.

Einerseits die *Issueservices*, welche die Badges erstellen und andererseits *Displayservices*, welche die Backpacks von Benutzern darstellen können. Natürlich können diese Teilservices auch in einem gemeinsamen Service vereint werden.

Um garantieren zu können das Badges auch von der Quelle stammen, von der diese vorgeben, zu stammen, sind die zwei Verifikationsmethoden *hosted* & *signed* spezifiziert worden.

Bei der *hosted* Methode, muss im Badge eine URL hinterlegt werden, die zu einer Datei auf dem Server des Badge aus-

stellenden Services verweist, welche die selben Informationen enthält.

Bei der *signed* Variante, muss der Badge mit einer Public/Private Key Methode signiert sein und eine URL zum Public Key des Issuers im Badge hinterlegt sein. Mit diesem Key kann dann die Signatur des Badges überprüft werden.

Der Nachteil der *hosted* Variante ist, dass im Grunde für jeden erstellten Badge eine JSON Datei aufbewahrt werden muss.

### **Purdue Passport**

Hierbei handelt es sich um ein Projekt der Purdue Universität, dass sich aber derzeit noch in einer mehr oder weniger geschlossenen Beta befindet. Zu den Features zählen ein relativ umfangreicher Badge Konfigurator zum Erstellen eigener Badges sowie Kompatibilität mit dem zuvor beschriebenen Open Badges Standard. Zusätzlich gibt es eine iPad App, die es ermöglicht, die eigenen Badges anzusehen. Für Lehrende gibt es zusätzlich noch die Möglichkeit die Badges und Achievements in numerische Werte zur Benotung umzurechnen.

### **Classbadges**

Ist ebenfalls ein Open Badges kompatibler Service für Schulen und Lehrer. Die Verwaltung der Accounts geschieht aber dennoch über eine zentrale Plattform und Lehrer sowie deren Schüler müssen sich auf der Webseite [classbadges.com] registrieren. Anders als Purdue Passport bietet Classbadges keinen Badgekongurator, hat aber einen großen Katalog an bereits fertigen Badgegrafiken der ständig erweitert wird. Es gibt aber auch die Möglichkeit eigene Grafiken in das System hochzuladen. Somit ist man entweder etwas in seinen Optionen eingeschränkt oder muss relativ viel Zeit für die Gestaltung von eigenen Badges investieren.

## Design

In das Design des Prototypen sind Erkenntnisse sowohl aus der Literaturrecherche als auch den im Rahmen dieser Arbeit geführten Interviews eingeflossen. Zusätzlich wurden auch Konzepte aus ähnlichen Plattformen aufgegriffen und entsprechend angepasst.

Aus den bereits erläuterten Interviews ergab sich vor allem die Anforderung, dass die Bedienung schnell und einfach von der Hand gehen muss. Somit sollte die nötige Einarbeitungszeit möglichst gering gehalten werden und die Bedienung weitestgehend selbsterklärend sein.

Ziel der grafischen Gestaltung war es, möglichst simpel zu sein und sich an der CI der TU Wien zu orientieren um so auch eine Art Proof of Concept darzustellen, wie so ein größeres und komplexeres Reputationsystem grafisch aussehen könnte.

Komplexe Grafiken wurden möglichst vermieden und eine textlastige Gestaltung forciert. Das hat mehrerlei Gründe. Primär wird dadurch die grafische Darstellung der Badges in den Vordergrund gerückt und der Benutzer wird nicht durch andere grafische Elemente abgelenkt. Ein weiterer Effekt ist, dass die Inhalte allgemein leichter zu erfassen sind, da man nicht durch verzierende grafische Elemente unnötig abgelenkt wird. Zu guter Letzt trägt eine minimalistische grafische Gestaltung natürlich auch zu einer Verkürzung der benötigten Implementationszeit bei.

## Gliederung

Die Badge Plattform unterteilt sich in folgende primäre Seiten:

- Login
- Badge Liste
- Preset Liste

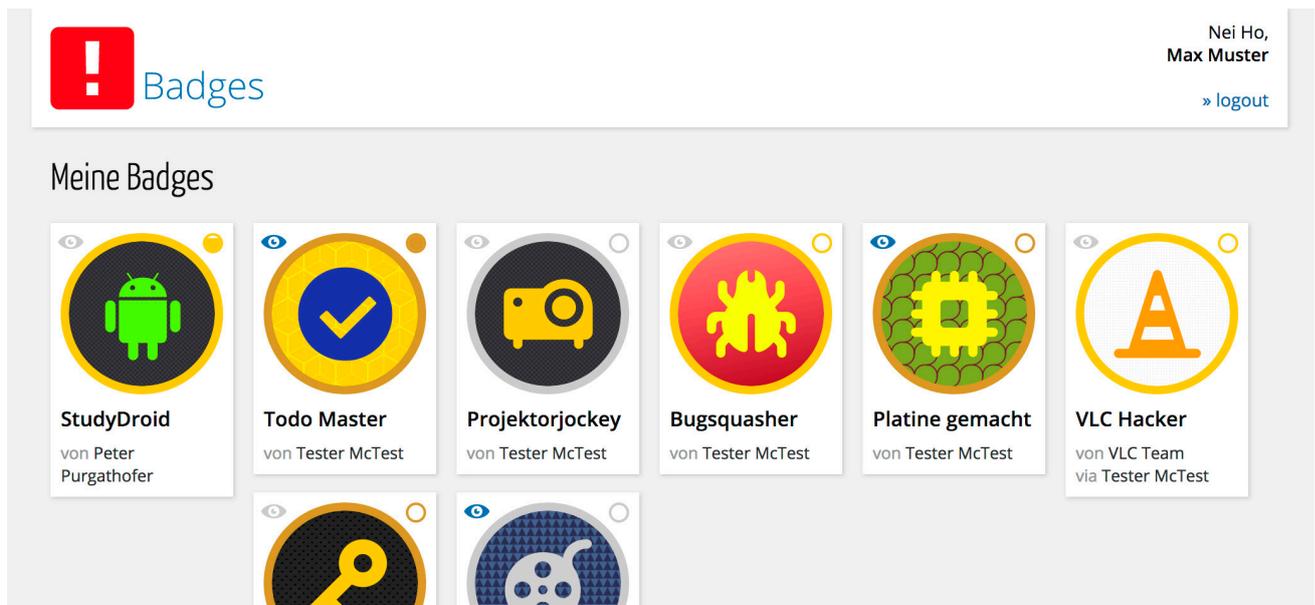
Zusätzlich gibt es noch Formulare und Detailansichten, welche als Modalfenster beziehungsweise Overlay über den Listenansichten eingeblendet werden:

- Badge Details
- Preset Details

- Vergabe Formular
- Preset Formular

Die folgenden Beschreibungen, der unterschiedlichen Ansichten, sind nicht nach dem üblichen Workflow geordnet, sondern so angeordnet, dass die Konzepte hinter dem Prototypen Schritt für Schritt näher gebracht werden.

## Badge Liste



Die Badge Liste ist beim Prototypen vor allem für Studenten von Bedeutung, da hier alle Badges gelistet werden, die ihnen verliehen wurden. Die Badgelisten eines bestimmten Benutzers ist auch über dessen ID öffentlich ohne Login einsehbar. Dies hat vor allem den Sinn, dass ein Student seine Badges etwa bei einer Bewerbung angeben oder anderen Dritten zeigen kann, die über keinen TU Wien Account verfügen.

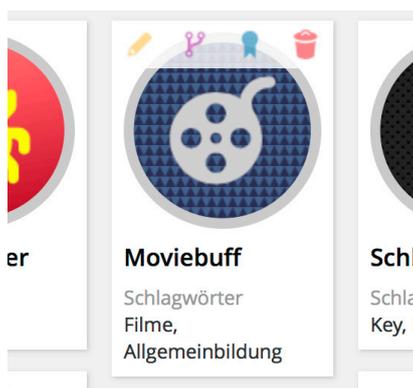
Da aber eventuell ein Student nicht alle seine Badges öffentlich zur Schau stellen möchte, ist das Veröffentlichen von Badges eine Opt-In Option. Dies wird so gelöst, dass wenn ein Student einen Badge verliehen bekommt, dieser vorerst einmal nur privat sichtbar ist. Also nur dann, wenn der Student sich mit seinem Account eingeloggt hat und seine Badge Liste betrachtet. Über ein Icon, das ein Auge darstellt, das entweder leicht durchsichtig oder vollkommen opak ist, bekommt der Student angezeigt welche seiner Badges öffentlich sind und welche nicht. Über einen Klick auf dieses Icon, kann der Student diesen Status

Screenshot der Badge Liste mit Testdaten.

wechseln. Sobald nun ein Badge auf veröffentlicht gestellt wurde ist dieser im öffentlichen Profil zu sehen.



Screenshot der Badge Detail Ansicht mit Testdaten.



Screenshot des Badge Preset Overlays mit Optionen zum editieren, duplizieren, vergeben und löschen.

## Badge Details

In der Badge Detailansicht sieht man zusätzlich zu den Informationen, die man bereits in der Liste erhält, den ausführlicheren Kommentare des Badgevergebers zur Leistung des Studenten, der den Badge erhalten hat. Auch die gesamte Liste an Schlagworten wird angezeigt, sollte diese zu lange und in der Listenansicht abgeschnitten sein.

## Preset Liste

Diese Ansicht bekommen alle Benutzer angezeigt, die berechtigt sind Badges zu vergeben. Auf dieser Seite werden alle bereits erstellten Badge Presets angezeigt und können von hier aus direkt editiert, dupliziert und vergeben werden.

Um grundsätzlich einen Badge vergeben zu können, muss zuerst ein Preset erstellt werden, welches dann im Anschluss vergeben werden kann. Dies hat vor allem den Sinn, dass nachvollziehbar bleibt, welche Studenten den selben Badge erhalten haben. Solange ein Preset nicht vergeben wurde, kann es weiterhin editiert werden. Sobald ein Preset verliehen wurde, ist es nicht mehr editierbar. Dieses Verhalten ist notwendig um sicherstellen zu können, dass der Badge für alle, die diesen erhalten, inhaltlich gleichwertig und somit auch vergleichbar bleibt.

Um dennoch das Editieren eines bereits vergebenen Presets zu ermöglichen, kann man dieses duplizieren und so ein neues Preset mit den selben Inhalten anlegen. Diese Verwandtschaft wird ebenfalls erfasst, so dass die Badges dennoch vergleichbar bleiben.

## Preset Detail

In der Preset Detailansicht bekommt der Benutzer zusätzlich zu den Informationen, die bereits in der Preset Liste zu sehen sind, eine Auflistung der Studenten, die bereits einen Badge basierend auf dem Preset erhalten haben.

## Preset Formular

Badge Preset erstellen

Projektorjockey  
Schlagwörter  
Beamer, Tech  
Support

Reader  
Schlagwörter  
reader

Name \* Chipjockey

Schlagwörter \* Elektrotechnik, Bauteilkunde

Leistung URL oder kurze Beschreibung

Kommentar

» Preset erstellen » Abbrechen  
» Preset und weiteres erstellen

Mittels des Preset Formulars können berechtigte Benutzer Vorlagen für Badges erstellen. Da jeder Badge durch eine Grafik repräsentiert werden soll, würde viel Aufwand für Gestaltung auf Benutzer zukommen, die sich dazu entschließen, mehrere verschiedene Badges zu vergeben. Um diesen Aufwand möglichst gering zu halten, kann sich der Benutzer, ähnlich wie beim Purdue Passport, seine Bad-gegrafiken über den, in das Preset Formular integrierten, Editor zusammenstellen, indem er aus einer Sammlung von Symbolen und Hintergründen auswählt und für das Symbol eine Farbe definiert.

In diesem Formular kann der Benutzer auch den Namen des Badges angeben, sowie eine Liste von Schlagwörtern, die immer fix mit diesem Badge verknüpft werden sollen. Zusätzlich kann der Benutzer auch einen Text definieren, der das entsprechende Feld im Vergabe Formular vorbe-füllt, aber dann noch geändert werden kann.

Wenn der Benutzer das Preset fertig erstellt hat, kann er dieses entweder speichern oder sowohl speichern als auch das Formular leeren um ein weiteres Preset zu erstellen.

Screenshot des Formulars zum erstellen von Badge Presets.

## Vergabe Formular

Badge Presets + P

**Projektorjockey**  
Schlagwörter  
Beamer, Tech Support

**Platine gemacht**  
Schlagwörter  
Platine, Elektrotechnik

**VLC Hacker**  
Schlagwörter  
VLC, Plugin, Modifikation

**Direktor**  
Schlagwörter  
Regie

### Badge vergeben

Der Student 08362821, Thomas Muster

hat im Sommersemester 2017

in 123.456 Test LVA

als einer von 12 LVA-TeilnehmerInnen

die folgende Leistung erbracht:  
Hat VLC plugin geschrieben

und erhält dafür den silbernen Badge

**VLC Hacker**  
Schlagwörter  
VLC, Plugin, Modifikation

von VLC Inc. via Tester McTest

mit folgendem Kommentar:  
\_\_\_\_\_

[» Badge vergeben](#) [» Abbrechen](#)  
[» Badge und weiteren vergeben](#)

Screenshot des Formulars zum Vergeben von Badges.

In dieser Ansicht wird ein zur Badgevergabe berechtigter Benutzer voraussichtlich die meiste Zeit verbringen, weswegen hier besonderer Wert darauf gelegt wurde, dass dem Benutzer repetitive und umständliche Eingaben bestmöglich abgenommen werden. Dies hat vor allem zwei Gründe. Erstens soll dem Benutzer so möglichst viel Zeit erspart bleiben, so dass das Vergeben von Badges überhaupt erst zeitlich in Frage kommt. Zweitens soll der Benutzer so auch mehr Zeit für das qualitative Bewerten in Form von Kommentaren zum Badge freigeräumt bekommen. Um das Formular etwas leichter verständlich und intuitiver zu machen, ist es wie ein Lückentext gestaltet, in den der Benutzer nur noch die restlichen benötigten Informationen eintragen muss.

Das Feld zur Eingabe des Studierenden wird durch eine automatische Vervollständigung gestützt, so dass dem Benutzer, nach Eingabe der ersten Ziffern der Matrikelnummer oder Buchstaben des Namens, Vorschläge gemacht werden, welchen Studierenden er denn gemeint haben könnte. Hier wird auch berücksichtigt ob ein Student die-

sen Badge schon erhalten hat und wird dann auch nicht gelistet.

Wenn der Benutzer das Formular fertig ausgefüllt hat, kann dieser es entweder bestätigen und zu seiner Preset Liste zurückkehren oder bestätigen und das Preset erneut vergeben. Wenn sich der Benutzer für letzteres entscheidet, werden Informationen wie z.B. die LVA übernommen um so dem Benutzer wieder etwas Zeit zu sparen.

## **Login**

Da der Prototyp möglichst in laufenden Lehrveranstaltungen getestet werden soll, muss sich dieser möglichst gut in die bestehende TU Wien Single Sign On (SSO) Infrastruktur einfügen, die es Studierenden und Lehrenden ermöglicht, sich bei den unterschiedlichen Online Services der TU Wien einzuloggen. Im Rahmen dieses Prototypen soll SSO primär dazu verwendet werden, Studierenden einen Login zu ermöglichen. Lehrende bzw. Benutzer die Badges vergeben, können sich vorerst nur über einen für diese Plattform spezifischen Account anmelden.

## **Badge Wert**

Damit ein Benutzer einschätzen kann wie viel ein Badge wert ist, sind drei Werte vorgesehen, an denen man sich orientieren kann.

Primär gibt es die *Wertigkeit*, die durch den Ausstellenden angegeben wird. Dieser kann entweder Gold, Silber oder Bronze wählen. Ähnlich wie bei Sportmedaillen ist Gold der Höchstmögliche Wert und Bronze der niedrigste. Wie von der Leistung, des Studenten, welcher den Badge erhält, auf diese Wertung zu schließen ist, wird dem Badgevergeber überlassen.

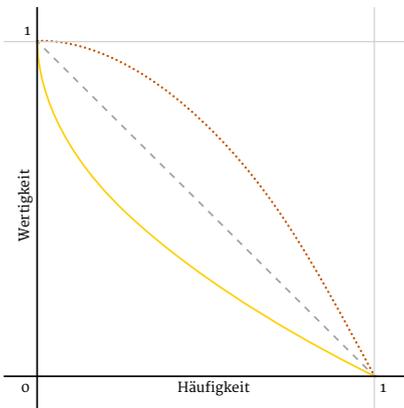
Basierend auf dem Wert des Badges und dem Vergabeverhalten des Badge Ausstellers, wird ein *Vertrauen* als ein Wert in  $[0-1]$  berechnet. Dieser geht davon aus, dass die Häufigkeit der Badges mit ihrer Wertigkeit sinken sollte. Vergibt jemand etwa nur goldene Badges, ist das Vertrauen entsprechend niedrig. Dieser Wert wird mit Hilfe von drei Kurven berechnet, wobei die Bewertung des Badges vorgeht, welche Kurve gewählt wird.

$$f(r) := 1 - (C_r / C_t)^{g(r)}$$

$$g(r) := \begin{cases} 2 & \text{wenn } r = 0 \\ 1 & \text{wenn } r = 1 \\ 0.5 & \text{wenn } r = 2 \end{cases}$$

$C_r$  .... Die Anzahl aller vergebenen Badges mit dem Rating  $r$

$C_t$  .... Die Anzahl aller vergebenen Badges



Visualisierung der Wertigkeitskurven für Gold ( $r=2$ ), Silber ( $r=1$ ), Bronze ( $r=0$ ).

Dies ist natürlich nur ein sehr rudimentäres System und auch die gewählten Kurven, welche zum Berechnen des *Vertrauens* dienen, sind eher arbiträr gewählt. In einem laufenden Betrieb würde diese Funktion voraussichtlich regelmäßig dem Nutzungsverhalten angepasst werden müssen.

Der dritte Wert ist die *Seltenheit* eines Badges. Diese wird anhand der Anzahl von vergebenen Badges vom selben Badgeprototypen innerhalb eines Semesters und der vom Badgevergebenden angegebenen Kandidatenzahl errechnet.

## Technische Implementation

Ich werde nun etwas auf die technischen Details und die Software Architektur des Prototypen eingehen. Im Rahmen dessen, werde ich auch auf bei der Umsetzung aufgetretene Probleme eingehen und erklären, wie diese gelöst wurden. Ich werde zuerst auf die serverseitigen Aspekte der Implementation, wie etwa die Verwaltung der Daten eingehen und dann auf die clientseitigen Aspekte, welche vor allem das Userinterface betreffen.

### Server

Der serverseitige Teil des Prototyp wurde mit dem Django Framework implementiert. Django ist ein Python basiertes Web-Framework, welches viele der typischen Funktionalitäten zur Verfügung stellt, die man zur Implementation einer Webplattform benötigt. Unter anderem eine Benutzer- und Rechteverwaltung, Hilfsmethoden zum einfachen erstellen von Datenmodellen und den entsprechenden Modellklassen in Python sowie eine Templateengine.

Die eigentlichen Badgegrafiken werden als SVG Dateien vom Server geliefert. Dies hat mehrere Gründe. Primär war es eine technische Entscheidung, da das Erstellen von Bitmapgrafiken in Django verhältnismäßig umständlich ist,

da entsprechende Python Bibliotheken erst nachinstalliert werden müssen, was den Rollout des Prototypen auf einem Live Server zusätzlich verkompliziert hätte. Dazu kommt, dass die Bitmapgrafiken relativ hochauflösend hätten ausgegeben werden müssen, da sich die Darstellungsgröße der Badges an die Fenstergröße des Browsers anpassen soll. Mit SVG konnten beide Probleme auf einen Schlag gelöst werden. Da SVG ein XML basiertes Vektorformat ist, kann dieses einfach über die in Django integrierte Templateengine erstellt werden. Zusätzlich kann die Badgegrafik durch ihren Vektorcharakter beliebig hochauflösend dargestellt werden und hat doch eine verhältnismäßig kleine Dateigröße und könnte zusätzlich noch gut komprimiert über HTTP übermittelt werden, da es sich ja um ein textbasiertes Format handelt. Was wiederum der Performance und Ladezeiten der Plattform zu Gute kommt. Auch die Unterstützung von SVG ist zum Zeitpunkt dieser Arbeit bereits in allen modernen Browsern gegeben.

Eine mögliche Limitierung der grafischen Gestaltungsmöglichkeiten der Badges durch das Vektorformat trifft auch nicht zu, da SVG viele Filter wie etwa Weichzeichner und Schlagschatten unterstützt, die man bei einem Rasterformat hätte zusätzlich implementieren müssen. Außerdem können in SVG Dateien auch Rastergrafiken eingebettet werden, sollte man einen gewünschten Effekt tatsächlich nicht erzielen können.

Beim Generieren der Badges, werden drei Teile miteinander verbunden. Die Grundlage ist ein Basis SVG, in welches je ein SVG für die Form sowie ein SVG für den Hintergrund eingefügt werden. Für die Hintergründe wurde eine Sammlung von frei verfügbaren SVG Mustern von [philbit.com] verwendet. Für die Formen wurde eine Auswahl von frei verfügbaren SVG Icons von [iconfinder.com] hergenommen. Um die Formen einfärben zu können, mussten diese noch so aufbereitet werden, dass sie das Attribut „fill“ mit einem Wert von „#000000“ (Hex Wert für die Farbe Schwarz) definiert haben. Dieser wird dann mittels eines Regex Ausdruck durch die eigentliche Farbe ersetzt.

Im bereits erwähnten Badge Konfigurator, kann der Benutzer aus einer Liste von Formen und Hintergründen auswählen. Diese befinden sich im „static/images/“ Ordner des Django Projektes in den zwei Unterordnern „shapes“

für die Formen und „patterns“ für die Hintergründe. Diese beiden Ordner werden dann bei einem Aufruf des Badge Preset Formulars ausgelesen und die gefundenen Grafiken als Liste in das entsprechende Template eingefügt. So können jederzeit weitere Grafikbausteine eingespielt werden, indem diese in die entsprechenden Ordner kopiert werden.

Wie bereits erwähnt, hat ein Benutzer der Plattform die Möglichkeit sich sowohl über das TU Wien SSO Service anzumelden, als auch über einen extra vergebenen Benutzernamen und Passwort. Um dies zu ermöglichen, wurde darauf verzichtet, dass mit Django mitgelieferte Benutzersystem für die primäre Funktionalität des Prototyp zu verwenden. Es wird aber dennoch für den von Django automatisch generierten Adminbereich verwendet, welcher dazu dient, die Benutzer anzulegen, die sich über einen eigenen Login anmelden können sollen.

Nachdem der Benutzer vom SSO Login auf die Startseite des Prototypen weitergeleitet wird, überprüft das Skript ob für die, mit dem Login mitgesendete Objekt ID (*oid*) bereits ein BadgeUser angelegt ist. Falls dem so ist, wird der Benutzer als dieser BadgeUser eingeloggt. Falls nicht, wird überprüft ob es bereits einen BadgeUser ohne Objekt ID (*oid*) und der selben Matrikelnummer (*student\_id*) gibt. Falls ja, wird der BadgeUser mit der Objekt ID (*oid*) verknüpft und der Benutzer als dieser BadgeUser eingeloggt. Falls kein passender BadgeUser gefunden wurde, wird ein neuer BadgeUser angelegt und der Benutzer als dieser eingeloggt.

Die Logout Funktion des TU Wien SSO wurde so implementiert, dass wenn der Logoff Request an den Prototypen geschickt wird, die aktuelle Zeit im BadgeUser mit der übergebenen Object ID (*oid*) im *logout\_date* Feld gespeichert wird. Zusätzlich wird bei jedem Aufruf der Seite durch einen eingeloggten Benutzer die aktuelle Zeit in der Session des Benutzers als *last\_action\_date* gespeichert. Sollte das im BadgeUser gespeicherte *logout\_date* aktueller als das *last\_action\_date* sein, wird die Session des Benutzers beendet und dieser ausgeloggt.

Wie bereits erwähnt, kommen mehrere Modalfenster zum Einsatz. Dies wird durch eine handvoll URLs ermöglicht, die über asynchronen JavaScript XMLHttpRequests (kurz AJAX) aufgerufen werden können, um so die nötigen Inhalte nachzuladen oder Formulardaten an den Server zu schicken. Um die Struktur des Prototypen etwas übersichtlicher zu gestalten, wurden deswegen alle Ansichten und Aktionen, die über AJAX aufgerufen werden, in eine gemeinsame Python Datei „ajax.py“ ausgelagert. Die Request verarbeitenden Methoden retournieren entweder HTML Snippets oder JSON encodierte Objekte. Sollte ein Fehler beim Verarbeiten des Requests auftreten wird wenn HTML als Response erwartet wird, der Inhalt des „error.html“ Templates retourniert. Alle JSON encodierten Responses enthalten das Feld *error* welches entweder den Wert *False* oder im Fehlerfall den Wert *True* hat. Im Fehlerfall hat das JSON encodierte Objekt zusätzlich noch das Feld *msg* mit einer Fehlermeldung.

Über AJAX werden die folgenden Requests abgearbeitet:

- **issue\_badge\_form**  
GET: *pid* - Die ID des zu vergebenden BadgePresets.  
Retourniert den HTML Code des Badgevergabe Formulars.
- **badge\_detail**  
GET: *id* - Die ID des anzuzeigenden Badges  
Retourniert den HTML Code für die Detailansicht eines Badges.
- **badge\_preset\_form**  
GET: *id (optional)* - Die ID des anzuzeigenden BadgePresets  
Retourniert den HTML Code des BadgePreset Formulars. Wenn der optionale Parameter *id* im Request enthalten ist, wird das Formular mit den Daten dieses BadgePresets vorbefüllt.
- **badge\_preset**  
GET: *id* - Die ID des anzuzeigenden BadgePresets  
Retourniert den HTML Code für die Listenansicht eines einzelnen BadgePresets.
- **badge\_preset\_detail**  
GET: *id* - Die ID des anzuzeigenden BadgePresets

Retourniert den HTML Code für die Detailansicht eines BadgePresets.

- **issue\_badge**

POST: *awardee* - Die Matrikelnummer des Badgeempfängers.

*awardee\_id (optional)* - Die ID des BadgeUsers des Badgeempfängers

*awarder (optional)* - Der Badgevergebende

*comment (optional)* - Ein Kommentar zum Badge

*keywords (optional)* - Mittels Komma (,) getrennte Schlagwörter

*lva (optional)* - Der Name der LVA bzw. des Kontext in dem der Badge vergeben wird.

*lva\_id (optional)* - Die ID der LVA

*pid* - Die ID des BadgePresets

*proof* - Die Begründung für die Badgevergabe (Text oder URL)

*rating* - Die Bewertung des Badges (1 - Bronze, 2 - Silber, 3 - Gold)

*semester (optional)* - Das Semester, für das der Badge vergeben werden soll

*students (optional)* - Die Anzahl der Studenten, die für den Badge in Frage kamen.

*year (optional)* - Das Jahr, für das der Badge vergeben werden soll.

Retourniert den Erfolg oder Misserfolg als JSON Objekt mit dem Feld *id*, welches die ID des neu erstellten Badges enthält.

- **save\_badge\_preset**

POST: *comment (optional)* - Kommentar zum BadgePreset

*id (optional)* - ID des zu editierenden BadgePresets

*img* - Die URL zur Badge Grafik

*keywords (optional)* - Mittels Komma (,) getrennte Schlagwörter

*name* - Der Name des BadgePresets

*proof (optional)* - Proof URL oder Text

Retourniert den Erfolg oder Misserfolg als JSON Objekt mit dem Feld *id*, welches die ID des neu erstellten oder editierten BadgePresets enthält.

- **duplicate\_badge\_preset**

POST: *id* - ID des zu duplizierenden BadgePresets

Sucht das BadgePreset mit der übergebenen *id* zum eingeloggten Benutzer heraus und legt ein neues BadgePreset mit den selben Werten an. Dem Namen wird ein Stern (\*) angehängt, damit der Benutzer in der BadgePreset Liste erkennt, welches BadgePreset das Neue ist. Retournt den Erfolg oder Misserfolg als JSON Objekt mit dem Feld *id*, welches bei Erfolg die ID des neuen BadgePresets enthält.

- **delete\_badge\_preset**

POST: *id* - ID des zu löschenden BadgePresets

Sucht das BadgePreset mit der übergebenen *id* zum eingeloggten Benutzer und überprüft ob dieses bereits vergeben wurde. Falls nicht, wird das BadgePreset gelöscht. Retournt den Erfolg oder Misserfolg als JSON Objekt mit dem Feld *id*, welches die ID des gelöschten BadgePresets enthält.

- **toggle\_public**

POST: *id* - Die ID des Badges

Ändert den Veröffentlichungsstatus des Badges mit der gegebenen *id* zum jeweilig negierten Booleschen Wert und retournt den resultierenden Veröffentlichungsstatus in einem JSON Objekt mit dem Feld *public*.

- **get\_user**

GET: *q* - Der Suchstring

Reicht den in *q* übergebenen Suchstring an die `BadgeUser.get_matching_user` Methode weiter und retournt das Ergebnis als JSON Objekt mit dem Feld *uid* welches die ID des Benutzers enthält, der gefunden wurde.

- **get\_users**

GET: *q* - Der Suchstring

*pid (optional)* - Eine BadgePreset ID

Reicht den in *q* übergebenen Suchstring an die `BadgeUser.get_matching_users` Methode weiter. Wenn *students True* ist, werden aus dem Ergebnis alle `BadgeUser`, die keine Studenten sind herausgefiltert. Wurde auch noch eine *pid* im request übergeben, werden auch alle `BadgeUser` herausgefiltert, die bereits einen Badge von diesem BadgePreset erhalten haben. Die übergebenen `BadgeUser` werden als Objekt mit den Werten *value* (Name des Benutzers) und *data* (ID des Benutzers) innerhalb eines JSON Objekt retournt.

*students (optional)* – Wenn True werden nur Studenten retourniert.

- **get\_courses**

GET: *q* – Der Suchstring

Ermittelt eine Liste von LVAs, die auf den Suchstring zutreffen und retourniert diese als JSON enkodiertes Array von Objekten im Feld *suggestions*. Die Objekte selbst haben die Felder *value* mit dem Namen der LVA und das Objekt *data* mit dem Feld *id*, welches die ID der LVA enthält und dem Feld *students* welche die Anzahl der Studenten in der LVA enthält. Diese Struktur rührt daher, dass das Ergebnis direkt von der [jquery.autocomplete] jQuery Erweiterung verwendet wird.

Diese Methode testet den Suchstring auf drei Szenarien: Zuerst wird überprüft, ob es sich bei dem Suchstring um eine komplette LVA Nummer mit Instituts- und zumindest teilweiser LVA Kennzahl handelt (z.B. „123.45“ findet 123.456). Falls nicht, wird überprüft ob es sich um eine 1-3 stellige LVA Nummer ohne Institut Kennzahl handelt (z.B. „45“ findet 123.456). Sollte dies auch nicht zutreffen, wird *q* innerhalb des LVA Titel gesucht (z.B. „Projekt“ findet „123.456 Projektmanagement“).

- **get\_tags**

GET: *q* – Der Suchstring

Ermittelt eine Liste der Tags, die den in *q* übergebenen String enthalten und retourniert diese als ein JSON enkodiertes Array von Objekten im Feld *suggestions*. Die Objekte selbst haben die Felder *value* mit dem Namen des Tags und das Objekt *data* mit dem Feld *id*, welches die ID des Tags enthält. Diese Struktur rührt daher, dass das Ergebnis direkt von der [jquery.autocomplete] jQuery Erweiterung verwendet wird.

- **get\_candidate\_count**

GET: *id* – Die ID des BadgePresets

*s* – Das Semester

*y* – Das Jahr

Retourniert das Ergebnis der BadgePresetSemesterCounts.*count* Methode im Feld *count*.

Die JSON enkodierten Ergebnisse der AJAX Aufrufe sind wie folgt aufgebaut:

- **error**  
*True* oder *False*, entsprechend dem Erfolg oder Misserfolg des Aufrufes.
- **msg**  
 Eine Nachricht, die dem Benutzer angezeigt werden kann um diesen über den Status des Aufrufes zu informieren.

Zusätzlich zu diesen beiden Feldern, enthält das Resultat aufrufspezifische Felder, wie bereits bei diesen beschrieben.

Da der Prototyp in Django implementiert wurde, ist die Datenmodellierung über die Implementation von Modell Klassen geschehen, welche wiederum automatisch die nötigen Tabellen in der verwendeten MySQL Datenbank anlegen und verwalten. Aus diesem Grund werde ich nur auf die Modellklassen eingehen und die eigentliche Datenbankstruktur außen vor lassen. Durch die Verwendung von Modellklassen, können auch komplexere Berechnungen und Hilfsfunktionen, die das jeweilige Modell betreffen, schön gekapselt werden. Ich werde auch diese kurz erläutern.

Die Datenverwaltung im Prototypen geschieht über die folgenden Modelle:

- BadgeUser
- BadgePreset
- Badge
- BadgePresetSemesterCounts
- LVA
- Tag

### BadgeUser

In diesem Modell werden alle Benutzerinformationen abgebildet. Diese bestehen aus den folgenden Feldern:

- **email**  
 In diesem Feld wird die E-Mail des Benutzers gespeichert. Dies kann entweder eine direkt definierte oder auch eine über die TU Wien SSO Schnittstelle angelieferte E-Mail Adresse sein.

- **password**  
Das Passwort Feld wird nur gesetzt, wenn der Benutzer manuell angelegt wurde und sich über den Login der Plattform anmelden können soll. Das Passwort wird gehashed in der Datenbank abgelegt. Für die Passwortoperationen wird das mit Django mitgeliefert *django.contrib.auth.hashers* Package verwendet.
- **firstname**  
Der Vorname des Benutzers
- **lastname**  
Der Nachname des Benutzers
- **student\_id**  
Die Matrikelnummer des Benutzers.
- **object\_id**  
Die ID des TU Wien SSO Accounts, der mit diesem Benutzer verknüpft ist.
- **role**  
Die Rolle des Benutzers. Diese kann entweder Student (stud) oder Professor (prof) lauten.
- **logout\_date**  
Speichert das TU Wien SSO Logout Datum.

Die BadgeUser Modell Klasse enthält auch noch die folgenden Hilfsfunktionen:

- **check\_password(raw\_password)**  
Diese Methode überprüft ob *raw\_password* mit dem Passwort der BadgeUser Instanz übereinstimmt.
- **set\_password(raw\_password)**  
Hasht *raw\_password* und setzt das Passwort der BadgeUser Instanz auf den resultierenden Wert.
- **credibility(rating)**  
Berechnet die Zuverlässigkeit bzw. Vertrauenswürdigkeit des Benutzers für das übergebene *rating*. Das Ergebnis ist ein Wert in [0 - 1]. Wobei 0 einem niedrigen Vertrauen und 1 einem sehr hohen Vertrauen entspricht.

Zusätzlich hat das Modell auch noch die folgenden statischen Methoden:

- **get\_matching\_user(query)**  
Diese Methode sucht eine BadgeUser Instanz in der Datenbank, welche auf den übergebenen query Parameter exakt zutrifft. Wobei der *query* Parameter darauf überprüft wird, ob es sich dabei um eine E-Mail Adresse, Matrikelnummer oder Vor- & Nachname handelt.
- **get\_matching\_users(query)**  
Diese Methode retourniert eine Liste von BadgeUser Instanzen, die den übergebenen *query* Parameter in E-Mail, Matrikelnummer, Vor- oder Nachname enthalten.

### **BadgePreset**

Dieses Modell speichert die Informationen der von den Benutzern angelegten BadgePresets und hat die folgenden Felder:

- **owner**  
Ein Fremdschlüssel von BadgeUser. Gibt den Eigentümer des BadgePresets an.
- **original**  
Ein Fremdschlüssel von BadgePreset. Dient dazu beim duplizieren eines BadgePreset abzubilden, welches BadgePreset der Vorgänger ist.
- **name**  
Der Name des BadgePresets
- **img**  
Die URL zum Bild des BadgePresets
- **keywords**  
Eine ManyToMany Verknüpfung zum Tag Modell. Speichert die vorgegebenen Schlagwörter.
- **proof**  
Ein optionales Feld für den Beweis beziehungsweise Begründung der Badge Vergabe.
- **comment**  
Ein optionales Feld für ein allgemeines Kommentar.

### **Badge**

Das Badge Modell bildet die vergebenen Badges ab.

- **awardee**  
Der Fremdschlüssel einer BadgeUser Instanz. Referenziert den BadgeUser, der den Badge erhalten hat.
- **issuer**  
Der Fremdschlüssel einer BadgeUser Instanz. Referenziert den BadgeUser, der den Badge vergeben hat.
- **awarder**  
Dieses Feld enthält den Namen der Person oder Institution oder den Fremdschlüssel von einer BadgeUser Instanz, die den Badge vergeben hat.
- **preset**  
Der Fremdschlüssel einer BadgePreset Instanz. Referenziert das BadgePreset, dass als Vorlage für den Badge verwendet wurde.
- **rating**  
Die Bewertung des Badges. Diese kann entweder Gold (2), Silber (1) oder Bronze (0) sein.
- **name**  
Der Name beziehungsweise die Bezeichnung des Badges.
- **img**  
Die URL zum Badge Bild.
- **keywords**  
Eine ManyToMany Verknüpfung zum Tag Modell. Speichert die Schlagwörter zum Badge.
- **proof**  
Ein optionales Feld für den Beweis beziehungsweise Begründung der Badge Vergabe.
- **comment**  
Ein optionales Kommentar zum Badge.
- **lva**  
Der Fremdschlüssel zu einer LVA Instanz. Ein optionales Feld für die LVA, in dessen Rahmen der Badge vergeben wurde.
- **context**  
Ein optionales Feld für einen frei definierbaren Kontext, in dem der Badge vergeben wurde.
- **public**  
Ein boolesches Feld, welches angibt, ob der Badge öffentlich einsehbar ist.

- **candidates**  
Anzahl der Studenten, die den Badge hätten erhalten können.
- **date**  
Das Datum, an dem der Badge in die Datenbank eingetragen wurde.
- **semester**  
Das Semester für das der Badge vergeben wurde. Sommer- (s) oder Wintersemester (w).
- **year**  
Das Jahr für das der Badge vergeben wurde.

Zusätzlich enthält die Modellklasse die folgenden Hilfsmethoden.

- **candidate\_count**  
Ruft die Methode *BadgePresetSemesterCounts.count()* mit den eigenen Feldern *year*, *semester* und *preset\_id* auf und retourniert das Ergebnis.
- **rating\_name**  
Retourniert den Namen für das numerische Rating des Badges. (2 - Gold, 1 - Silber, 0 - Bronze)
- **awarder\_name**  
Überprüft ob das eigene *awarder* Feld auf einen BadgeUser referenziert oder einen frei vergebenen Namen enthält. Wenn auf einen BadgeUser referenziert wird, werden Vor- und Nachname dieser Instanz als String retourniert ansonsten der Inhalt des *awarder* Feldes.
- **via**  
Falls der *awarder* ein frei vergebener Name ist, retourniert diese Funktion den Vor- und Nachnamen des als *issuer* referenzierten BadgeUser.
- **rarity**  
Ermittelt die Seltenheit des Badges und retourniert diese als Wert in [0-1] für weniger selten bis hin zu sehr selten. Die Seltenheit wird mit den selben Funktionen wie das Vertrauen berechnet.
- **rarity\_percent**  
Ermittelt die Seltenheit des Badges mittels *rarity()* und retourniert diese als prozentualen Wert in [0-100].
- **issued**  
Ermittelt wie oft der Badge bereits vergeben wurde.

- **value**  
Retourniert das Ergebnis der *credibility()* Methode des BadgeUsers, der in *issuer* referenziert ist.
- **value\_percent**  
Retourniert das Ergebnis der *credibility()* Methode des BadgeUsers, der als *issuer* referenziert ist als prozentualen Wert.
- **factor(rating)**  
Retourniert den Faktor für das übergebene *rating*. Dieser wird vor allem bei den internen Berechnungen für *rarity* und *rating* verwendet.

### BadgePresetSemesterCounts

Bei diesem Modell handelt es sich um ein Hilfsmodell, welches die Anzahl von Kandidaten für ein BadgePreset innerhalb eines Semester speichert.

- **preset**  
Der Fremdschlüssel einer BadgePreset Instanz für die die Anzahl der Kandidaten gespeichert werden soll.
- **semester**  
Das Semester für das die Zahl der Kandidaten gespeichert werden soll. Sommer- (s) oder Wintersemester (w).
- **year**  
Das Jahr, für welches die Anzahl der Kandidaten gespeichert werden soll.
- **candidates**  
Die Anzahl der Kandidaten.

Das BadgePresetSemesterCounts Modell hat zusätzlich noch die Folgenden Methoden:

- **count(year, semester, preset\_id)**  
Eine Hilfsfunktion um die Anzahl der Kandidaten für ein bestimmtes BadgePreset (*preset\_id*), Jahr (*year*) und Semester (*semester*) zu ermitteln.  
Falls es keinen passenden Eintrag gibt, retourniert die Funktion 1.

### LVA

Dieses Modell bildet Lehrveranstaltungen ab.

- **institute**  
Die Institutsnummer der Lehrveranstaltung.
- **number**  
Die Nummer der Lehrveranstaltung.
- **title**  
Der Titel der Lehrveranstaltung
- **tutors**  
Eine ManyToMany Beziehung mit dem BadgeUser Modell. Speichert welche BadgeUser bereits Badges bzw. BadgePresets für diese LVA erstellt haben.
- **students**  
Die Anzahl der Studenten in der Lehrveranstaltung.

### Tag

Das Tag Modell bildet Schlagwörter ab. Das Modell hat nur die folgende Eigenschaft:

- **name**  
Der Name des Schlagwortes.

### **Client**

Der clientseitige Teil wurde mit HTML5, CSS3 und JavaScript implementiert. Da CSS3 zu diesem Zeitpunkt noch nicht von allen Browsern komplett unterstützt wurde, ist auch darauf geachtet worden, die herstellereinspezifischen Präfixe bei den CSS Befehlen und entsprechende Fallbacks zu verwenden, so dass die Plattform, auch mit älteren Browsern, noch bedienbar bleibt.

JavaScript wird benötigt, um die Plattform verwenden zu können, da viele Inhalte mittels AJAX nachgeladen werden. Nachdem sich ein Benutzer angemeldet hat, bekommt er die Liste aller Badges angezeigt, die er bereits erhalten hat. Mittels CSS Mediaqueries wird die Breite der einzelnen Badges abhängig von der Fenstergröße so verändert, dass entweder Fünf bis Zwei Badges nebeneinander aufgelistet werden oder auf Smartphones die Badges nur noch untereinander dargestellt werden. Die Positionierung der Badges geschieht mittels der JavaScript Bibliothek „isotope“, welche es ermöglicht beliebige HTML Elemente auf unterschiedliche Weise anzuordnen, die in dieser Form nicht mit HTML und CSS umsetzbar wären. In diesem Fall habe

ich das „masonry“ Layout gewählt, welches die Elemente verschachtelt bzw. geziegelt anordnet und so möglichst Leerflächen reduziert.

Der Badge Konfigurator besteht im Grunde aus vier Komponenten: Vorschau, Symbol- & Hintergrundauswahl sowie die Farbauswahl.

Die Vorschau ist einfach ein als HTML `<img>` Element eingebundenes SVG, dass bei jeder Änderung neu vom Server angefordert wird, indem das `src` Attribut des Elements entsprechend abgeändert wird.

Die Symbol- & Hintergrundauswahl sind einfache HTML Listen, die über JavaScript in Seiten aufgeteilt und blätterbar gemacht werden.

Die Farbauswahl ist eine visuell angepasste Variante des jQuery Plugins „Spectrum“.

Um die Formulare möglichst schlank zu gestalten und um den gewünschten visuellen Effekt eines Lückentextes zu erhalten war es nötig, eine automatische Größenanpassung für `<input>` und `<textarea>` Elemente zu implementieren. Hierzu werden die CSS Formatierungen sowie der Inhalt der Eingabefelder in ein verstecktes `<div>` Element kopiert und die Größe dieses Elements ermittelt. Die so erhaltenen Abmessungen werden dann wiederum auf das Eingabefeld angewendet. Um die Größe jederzeit entsprechend der Eingabe anzupassen und dennoch Ressourcenschonend bei der CPU Auslastung zu bleiben, wird dieser Vorgang nur bei jedem *change*, *input*, *keydown*, *keyup* DOM Event auf dem entsprechenden Eingabefeld ausgeführt und somit nur, wenn sich etwas am Inhalt ändert.

Für die Autovervollständigung mancher Felder wurde das jQuery Autocomplete Plugin von Thomas Kirda [jQuery Autocomplete] verwendet. Dieses achtet auf Änderungen des Inhalts im gewünschten Eingabefeld. Sobald eine Änderung auftritt, wird eine Anfrage an den Server geschickt, passende Vorschläge zu dem aktuellen Inhalt zurück zu liefern. Werden vom Server Vorschläge retourniert, werden diese in einer Liste Unterhalb des Eingabefeldes angezeigt und können mittels der Maus oder den Pfeiltasten auf der Tastatur ausgewählt werden. Der Inhalt dieses Vorschla-

ges wird dann 1:1 in das Eingabefeld übernommen. Auf die nötigen serverseitigen Komponenten, um passende Vorschläge zu ermitteln, bin ich bereits im vorherigen Kapitel eingegangen.

Das Django Framework bietet unter anderem einen CSRF (Cross Site Request Forgery) Schutz. Dieser soll verhindern, dass Formulareingaben in einem Request von einer anderen Quelle als der aktuellen Clientsession geschickt werden. Dies wird erreicht, indem in jedem Formular ein sogenannter CSRF Token in einem versteckten `<input>` Element eingebaut wird. Wenn dieses Formular nun an den Server abgeschickt wird, überprüft Django zuerst ob der mitgesendete CSRF Token gültig ist, bevor der Request an die eigentlichen verarbeitenden Methoden weitergeleitet wird. Dies funktioniert quasi automatisch, wenn man ein Formular wie üblich mittels submit button über POST absendet. Nur in diesem Prototypen werden alle Formularedaten, ausgenommen dem Login, mittels AJAX an den Server geschickt. Auch für dieses Problem bietet Django eine Lösung in Form von einem Javascript Snippet, welches man in seinen eigenen Code einbinden muss. Dieses Snippet hängt sich an die in jQuery inkludierten AJAX Funktionen und fügt den CSRF Token zum Http Request Header hinzu. Den Wert für den CSRF Token erhält das Script aus einem Cookie. Es kann aber manchmal passieren, dass dieses Cookie nicht richtig gesetzt wird. Um dies zu beheben, habe ich das Script so erweitert, dass der CSRF Token zusätzlich aus dem bereits erwähnten versteckten `<input>` Element im Formular ausgelesen wird.

## **Evaluation**

Um die vorgestellten Konzepte und Implementierung vollständig zu überprüfen, wäre ein Feldversuch über mehrere Semester nötig und würde die Zielsetzung dieser Arbeit weit überschreiten. Um aber dennoch eine Evaluation in einem überschaubaren Rahmen zu ermöglichen, wurde der Prototyp an einige der Professoren verteilt, die bereits bei den eingehenden Interviews zu dem Thema Lehre und Reputation befragt wurden. Zusätzlich wurde eine Gruppendiskussion mit Studenten und Studentinnen geführt,

die mit dem System konfrontiert wurden, um so Feedback von allen primären Stakeholdern zu erhalten.

Bei der Gruppendiskussion gab es vor allem Bedenken zu den Wertigkeiten Gold, Silber und Bronze, die man Badges zuweisen kann, da diese das Herausstellungsmerkmal eines Badges selbst zu sehr verwässern könnten oder dazu beitragen könnten, dass Badges nur als Punkte- oder Checklisten-System für Aufgaben eingesetzt werden.

Trotz der potentiellen Gefahr eines Einsatzes als Checkliste, hätten sich die Studierenden dennoch gewünscht, im Vorhinein zu wissen, welche Badges es zu ergattern gibt. Sei es über das Badge System selbst oder in einer anderen Form.

Im Großen und Ganzen hat diese kurze Erhebung bei den Studierenden größtenteils die Ergebnisse von [Haaranen et al] widerspiegelt, welche die Badges aber nur im Rahmen eines Online Kurses untersucht haben. Eine der Erkenntnisse ihrer Untersuchung sowie auch der kurzen Erhebung im Rahmen dieser Arbeit ist, dass die richtige Kommunikation und adäquate Transparenz der Funktionsweise sehr wichtig dafür sind, wie die Badges von den Studierenden angenommen werden, sollten diese voll im Unterricht eingesetzt werden. Vor allem dahingehend, Abstoßreaktionen durch Studierende vorzubeugen, so dass Badges keinen negativen Einfluss auf die Lehre haben.

Der Aspekt der Kommunikation und Funktionsweise hat sich auch bei den Gesprächen mit den Professoren wieder hervorgetan, aber aus einer umgekehrten Perspektive. Diese hatten vor allem Fragen dazu, wie Sie denn die Badges nun am besten einsetzen sollten.

Ein weiterer Punkt war auch die noch etwas limitierten Möglichkeiten beim erstellen von Badges, welche aber leicht in der Zukunft erweitert werden könnten.

Auch kritisiert wurde der Wert des *Vertrauens* in die Wertigkeit (Gold, Silber & Bronze) eines Badge. Um ein hohes Vertrauen zu erreichen müssten Professoren stark darauf achten, die Wertigkeiten in einem guten Verhältnis zu verteilen, was aber im Lehralltag eher eine Hemmschwelle für den Lehrenden darstellen würde, die Badges einzusetzen.

Auch wenn das Webinterface leicht und schnell zu bedienen war, ist es doch etwas unhandlich direkt in der Lehrveranstaltung einzusetzen, da man für jeden Studierenden dennoch die nötigen Informationen eintragen muss. Eine technische Lösung dies zu beschleunigen wäre also laut der Lehrenden wünschenswert.

Durch die Evaluation zeigte sich vor Allem, dass man neben einer Plattform auch eine gut verständliche Anleitung bzw. Liste von Best Practices generieren sollte, um einen möglichst reibungslosen Ablauf und eine einheitliche Erfahrung mit den Badges zu garantieren.

Eine weitere Erkenntnis sowohl aus den Feedbackrunden mit Studierenden als auch den Gesprächen mit den Lehrenden ist wohl, dass die Vergabe einer Wertigkeit zu einem Badge nochmals zu überdenken bzw. zu entfernen wäre, da diese mehr Fragen in der Benutzung aufwirft, als dass sie Klarheit über die Art der ausgezeichneten Leistung schafft.

## **Mögliche Erweiterungen**

Der implementierte Prototyp beschränkt sich nur auf das Nötigste, könnte aber durchaus noch um ein paar nützliche Funktionen erweitert werden, um zu einer vollwertigen Plattform zu wachsen, die im Studienalltag zum Einsatz kommen könnte. In diesem Abschnitt möchte ich auf solche möglichen Erweiterungen und deren potentiellen Nutzen eingehen.

### **Open Badges Unterstützung**

Die mit Abstand wichtigste Erweiterung der Plattform wäre eine Unterstützung des bereits beschriebenen Open Badges Standards von Mozilla [openbadges.org]. Um dies zu erreichen müsste man vor allem die nötigen Funktionen für das „Baking“ von Badges, sprich den Export von Grafiken mit eingebetteten Badge Assertion Informationen implementieren. Der Grundstein dazu ist mit der Generierung der Badges als SVG bereits gelegt. Diese SVGs müssten nur noch mit dem nötigen Tag für die Badgeinformationen versehen, beziehungsweise in PNG Bilddateien mit einem entsprechenden Datei Header umgewandelt werden.

Etwas aufwendiger, aber genauso wichtig, wäre das Implementieren eines Verifikationsservice, über den die exportierten Badges auch von anderen Open Badges Services verifiziert werden können.

### **Badgevergabe durch Studenten**

Ein weiteres interessantes Feature für Studenten könnte die Möglichkeit darstellen, selbst Badges zu vergeben. Wobei noch zu überlegen wäre, ob Studenten auch selbst Badges designen können oder ob es nur fix vorgegebene Badges geben sollte. Auch beide Varianten zu realisieren ist denkbar. Bei der ersten Variante bestünde vor allem die Gefahr einer gewissen Verwässerung der Effektivität der Badges, wenn es zu einer unverhältnismäßigen Flut von unterschiedlichsten Badges kommt, welche nur noch eingeschränkt vergleichbar sind.

Fix definierten Badges könnten so umgesetzt werden, dass Studenten etwa Badges für die beste Lehrveranstaltung eines Semester an einen ihrer Professoren vergeben können oder auch eine limitierte Zahl an „Helferlein“ Badges an Studenten die Ihnen geholfen haben.

Eine weitere Variante von Badges, die durch Studenten vergeben werden könnten, sind Systemweite Badges, die etwa semesterweise vergeben werden und bei denen alle Studenten Kandidaten vorschlagen können. Am Ende der Periode erhält der Student mit dem größten Zuspruch dann diesen Badge. Dies könnte auch auf den Rahmen einzelner Lehrveranstaltungen begrenzt werden, um so zum Beispiel das beste Projekt oder die eleganteste Lösung zu einer Problemstellung zu küren.

### **Badge Stacking**

Derzeit kann man einen bestimmten Badge nur einmal erhalten. Es wäre aber auch interessant, wenn man den selben Badge mit unterschiedlichen Begründungen öfters erhalten könnte und dieser etwa im Wert aufsteigen kann (z.B. 3 mal Bronze gibt 1 Silber) oder in eigens dafür eingeführten Abstufungen.

## **Statistik & Revisionen**

Um nicht den Überblick über die erhaltenen und vergebenen Badges zu verlieren, wäre eine Statistik zu den einzelnen Badges eine interessante Erweiterung, die es auch ermöglicht, die Wertigkeit von Badges noch besser einzuschätzen. Denkbare statistische Auswertungen wären etwa eine semesterweise Aufschlüsselung der Vergabe eines Badge Presets oder des Badgevergabebehavior eines Badgevergebenden Benutzers.

Ebenfalls einfach zu realisieren wäre es, eine Art Revisionsgraph aus den Änderungen sowie Duplikaten eines Badge Presets zu generieren.

## **Bessere TISS Anbindung**

Eine bessere Anbindung an das TISS wäre auch erstrebenswert, um so etwa Benutzer automatisch als Professoren registrieren zu können.

Derzeit müssen im System Lehrveranstaltungen vom Badgevergebenden selbst angelegt und mit einer Teilnehmerzahl versehen werden. Mit einer TISS Anbindung wäre es möglich, diese Informationen automatisch dem Benutzer zur Verfügung zu stellen, vorausgesetzt die Lehrveranstaltung ist ordentlich im TISS eingetragen.

Auch eine Übersicht der eigenen Studenten je Lehrveranstaltung für Professoren wäre denkbar, um so das Vergabe von Badges etwas zu beschleunigen.

## **Sonstiges**

Zusätzlich zu den bereits genannten Erweiterungen gäbe es auch noch einiges an weiteren Detailkorrekturen. Allen voran wäre eine bessere WAI [WAI] Konformität, so dass eine angemessene Barrierefreiheit gegeben ist.

Der Badgekonfigurator müsste auch noch um einige Funktionen, wie etwa eigene Formen und Hintergrundmuster erweitert werden. Außerdem kann man derzeit nur kreisrunde Badges erstellen, eigene oder auswählbare Silhouetten, würden durchaus zu einem besseren Wiedererkennungswert von Badges beitragen.

## **Mobile Applikation**

Die Infrastruktur des Prototypen könnte auch als Grundlage für eine Smartphone Applikation dienen, die es ermöglicht Badges direkt über NFC oder ähnliche Technologien zu vergeben, indem die vergebende Person nur das Badge Preset, eine Bewertung und Lehrveranstaltung auswählen müsste und dann sein Smartphone mit dem des Studenten aneinander halten. Die beiden Smartphones würden dann noch die Informationen bezüglich des Studenten austauschen und das Smartphone des Lehrenden schickt den Issue Request anschließend an den Server.

So eine Applikation könnte vor allem für die Badge vergebenden Personen eine Zeitersparnis bedeuten, da diese, beim aktuellen Stand des Prototypen, sich merken müssten wem sie welche Badges ausstellen wollen, um dies dann zu einem späteren Zeitpunkt in das System eintragen zu können. Auch wenn das System um die Möglichkeit erweitert wird, dass Studenten Badges vergeben können, wäre so eine Applikation sehr hilfreich, da man sich nicht die ganzen Matrikelnummern merken müsste, um seinen Kollegen Badges vergeben zu können.

## **Zusammenfassend**

In diesem Kapitel habe ich den, im Rahmen der Recherchen zu dieser Diplomarbeit erstellten, Badge Prototypen etwas näher erläutert. Ich bin auf ähnliche Projekte und den Open Badges Standard eingegangen, um so einen Kontext für die darauf folgenden näheren Erläuterungen zu schaffen. Danach habe ich das Design der implementierten Badge Plattform so wie technische Implementationsdetails näher besprochen.

Nach der Darlegung von Design und Umsetzung habe ich die Erkenntnisse aus einer Reihe von Gesprächen und Diskussionen zusammengefasst bei denen der Prototyp Studenten und Lehrenden vorgestellt und von diesen Feedback eingeholt wurde.

Abschließend bin ich dann, basierend auf dem Feedback, sowie bereits bewussten Schwachstellen noch auf mögliche Erweiterungen eingegangen, die den Prototypen näher an einen Einsatz im Studien- und Lehralltag rücken würden.

## TU Wien Reputation

Der im vorhergehenden Kapitel vorgestellte Badge Prototyp ist hauptsächlich als ein Beispiel, für die Implementation eines komplett neuen und alleinstehenden Teils eines, TU Wien umfassenden Reputationsystems zu verstehen. In diesem Kapitel möchte ich darauf eingehen, wie so ein gesamtes Reputationsystem aussehen könnte und wie man auch bereits vorhandene Onlineportale der TU Wien in dieses einbinden könnte. Ich werde mögliche Probleme und Spannungsfelder ansprechen und bereits existierende Lösungsansätze dazu vorstellen und diskutieren, wie diese in diesen Kontext übernommen werden können.

### Ziele & Konzepte

Ein schwieriger Aspekt bei der Gestaltung von Reputationsystemen für Online Communities ist laut [Farmer & Glass] die richtigen Parameter und Erfassungsmethoden zu wählen, so dass diese optimal funktionieren und auch einen verlässliche Orientierungshilfe bieten. Erschwerend kommt im Falle der TU Wien dazu, dass selbst innerhalb der Informatik keine, an einem Punkt konzentrierte, Online Community existiert. Statt dessen gibt es eine Unzahl von kleineren und größeren Plattformen, welche wie TUWEL oder TISS von der TU Wien bzw. dem ZID zur Verfügung gestellt werden. Unterschiedliche Institute und Professoren verwenden aber auch selbstgebaute Lösungen, die auf ihre jeweiligen Bedürfnisse angepasst sind. Dazu kommen dann auch noch, von Studenten organisierte, Plattformen wie etwa das informatik-forum.at oder das Vorlesungs Wiki [VoWi].

Um angemessene Reputationmechanismen gestalten zu können, möchte ich vorerst kurz definieren welche Auswirkung diese auf die Benutzer bzw. Studenten haben sollen. Der Fokus liegt, wie bereits in der Motivation erwähnt, vor allem darauf, außerordentliche Leistungen besser präsentierbar zu machen, aber Studenten auch zu diesen Leistungen zu motivieren. Somit sollte das Reputationsystem die Möglichkeit bieten, den Studenten Anreize geben zu können, sich mit neuen oder tiefergehender mit Themen zu befassen.

Ein weiterer wichtiger Aspekt war für mich, die Zusammenarbeit zwischen den Studierenden zu stärken. Sowohl bei Lehrveranstaltungen mit Gruppenarbeiten, als auch bei Einzelaufgaben.

Kompetitive Faktoren sollten meiner Meinung nach eine möglichst geringe Rolle spielen, aber dennoch möglich sein, da es durchaus in manchen Lehrveranstaltungen als Motivator Sinn macht. Was es meiner Meinung nach auf keinen Fall geben sollte, ist jegliche Form von Leaderboards oder Rankings.

Bisher wurde Reputation in dieser Arbeit vorwiegend in direktem Zusammenhang mit einer einzelnen Person und dessen Leistungen behandelt. Oft ist es aber so, dass diese Leistung auch eine Teilleistung einer größeren Gesamtleistung darstellt. Etwa bei Gruppenarbeiten, bei denen die Gruppenmitglieder eine oder mehrere Teilaufgaben erledigen um ein gemeinsames Gesamtziel zu erreichen. Das Reputationsystem sollte also möglichst auch solche Konstellationen erfassen und darstellen können.

Im Laufe von Recherchen und eigenen Überlegungen, sowie in Designworkshops wurden einige Konzepte erarbeitet, um dieser Fragmentierung von Plattformen und Aufgabenstellungen Herr zu werden. Diese lassen sich von den Grundkonzepten her in drei Kategorien zusammenfassen:

- **Reputation Widgets**  
Hierbei handelt es sich um kleine HTML oder JavaScript Code Stücke, die einfach auf einer Website eingebaut werden können. Mit Hilfe dieser Reputation Widgets kann man bestehende Plattformen mit relativ wenig Aufwand um eine Reputation Funktion erweitern.
- **Zentraler Reputation Aggregator**  
Eine zentrale Plattform auf der die Reputationinformationen verschiedener Plattformen gesammelt und präsentiert werden können. Dies umfasst sowohl TU interne oder nahe Plattformen, als auch externe Plattformen, wie etwa [stackoverflow.com](https://stackoverflow.com) oder [coderwall.com](https://coderwall.com).
- **Neue (zusammenfassende) Plattformen**  
Ein großer Teil der Fragmentierung ist darauf zurück zu führen, dass gewisse Lösungen von mehreren Leh-

renden bzw. Instituten gebraucht werden, aber nicht verfügbar sind. Dies führt dazu, dass jeder seine eigene Lösung implementiert oder mit verfügbaren Teillösungen „zusammenbastelt“.

Weitere Plattformen bergen natürlich die Gefahr einer zusätzlichen Fragmentierung. Deswegen sollte die Entwicklung dieser möglichst in Zusammenarbeit mit den Personen geschehen, die aktuell bereits eigene, ähnliche Lösungen einsetzen um so die Hemmschwelle für einen Wechsel auf ein Minimum zu bringen.

## **TURep**

Das Herzstück meines Vorschlags für ein allgemeines Reputationsystem wäre das sogenannte TURep, kurz für „TU Reputation“. In diesem Kapitel möchte ich vor allem auf die gestalterischen und konzeptionellen Aspekte dieses Systems eingehen. Probleme bezüglich Datenschutz, Privatsphäre und Überlegungen zu Implementationsdetails werde ich danach in separaten Kapiteln diskutieren.

TURep soll zwei der einleitend angesprochenen Konzepte erfüllen. Es soll sowohl ein Reputation Aggregator sein als auch Reputation Widgets zur Verfügung stellen.

Grundsätzlich soll TURep zu jedem arbiträren Objekt beziehungsweise Identifikationsnummer, Reputation Informationen sammeln können, insofern ein gültiger TU Wien SSO Account mit diesem Objekt verknüpft ist. Wobei gerade das Validieren dieser Verknüpfung eine der größten Herausforderungen darstellt. In späteren Kapiteln werde ich anhand von Beispielen beschreiben, wie man mit solchen Verknüpfungen umgehen kann. Für den Rest dieses Kapitels gehe ich einmal davon aus, dass die Verknüpfung ordentlich zustande gekommen und valide ist.

## **Datenbank**

Wie bereits bei der Definition des Begriffes Reputation erwähnt, handelt es sich bei dieser im Grunde um eine Ansammlung von Daten zu einem Objekt. Diese Daten müssen dann entsprechend aufbereitet werden, um im richtigen Moment oder Kontext eine aussagekräftigste Reputationinformation liefern zu können.

Somit sollte TURep konzeptionell gesehen, vor allem eine Art von Datenbank sein, die es erlaubt Informationen jeglicher Art zu erfassen. Die explizite Bewertungsinformation bestehend aus dem Tripel Bewertender, Wertung & Bewerteter ist hierbei das absolute Minimum. Als nächstes wäre schon die Information des „Was“ wichtig. Diese könnte aus einer simplen URL bestehen. Könnte aber auch die Sache selbst sein oder zumindest eine digitale Variante oder abstrahierte Beschreibung, wie etwa in einem digitalen Portfolio üblich. Auch den Kontext der Bewertung darf man nicht vernachlässigen.

Alle diese Daten zentral und wahrscheinlich teilweise auch noch redundant zur Datenbank der ursprünglichen Plattform zu speichern, ist aber nicht sonderlich sinnvoll oder gut planbar und würde unabschätzbar viele Ressourcen verschlingen. Aus diesem Grund schlage ich vor, TURep als eine dezentrale Datenbank bzw. Service zu implementieren, der zumindest teilweise, bei den jeweiligen bereits existierenden Plattformen integriert wird und diese somit zu TURep Nodes aufwertet. Technisch wäre es sehr naheliegend hier auf bereits vorhanden Protokolle wie RDF, WebDAV und SPARQL zurückzugreifen, wie dies etwa für dezentrale Soziale Netzwerke von [Yeung et al, 2009] vorgeschlagen wird. Als Alternative könnte man auch noch proprietäres REST / HTTP Protokoll ähnlich dem von CouchDB [couchdb] oder MongoDB [mongodb] einsetzen, da man diesen Aspekt von TURep im Grunde wie einen „dummen“ Objektspeicher implementieren könnte. Damit dies funktionieren kann, müssen natürlich alle Knoten in diesem Netzwerk die Protokolle unterstützen können, um die Anfragen der anderen Peers verstehen und entsprechende Ergebnisse retournieren können. Hierzu bräuchte es eine schlanke Bibliothek in einigen gängigen Serverseitigen Scriptsprachen, die es erlaubt, die einzelnen TU Wien internen Plattformen um diese Funktionalität zu erweitern. Oder aber auch eine zwischengeschaltete Applikation, welche die Anfragen entsprechend übersetzen kann, in dem etwa eine bereits vorhandene API verwendet wird oder die nötigsten Informationen von der Zielplattform, z.B. über das Analysieren der HTML Seiten extrahiert

werden und in einer separaten Datenbank zwischengespeichert werden.

Derzeit lässt es sich nicht wirklich abschätzen, ob die von Yeung vorgeschlagenen Protokolle sich auf längere Sicht etablieren werden. Selbst Opensource Projekte wie das verteilte soziale Netzwerk Diaspora [diaspora], setzen auf proprietäre Protokolle für den Austausch zwischen den einzelnen Servern. Aus praktischen Überlegungen würde ich daher eher die Variante realisieren, bei der TURep eine Basis API zur Verfügung stellt, die von externen Plattformen und Crawling Servern verwendet werden kann.

### **Kontext & Schlagworte**

Das TURep System sollte zu jeder Reputationinformation möglichst versuchen, den Kontext und konkretisierende Schlagworte mit zu speichern. Der grundlegende Kontext ergibt sich aus der Plattform auf der ein Widget eingesetzt wird oder die als externe Datenquelle dient. Dennoch kann es sein, dass auf dieser Plattformen mehrere unterschiedliche Themen behandelt werden. Beispielsweise TUWEL wird von mehreren unterschiedlichen Lehrveranstaltungen genutzt und auf Stackoverflow werden verschiedenste technische Probleme besprochen. Hierzu sollte mit der Reputationinformation möglichst eines oder mehrere Schlagworte zusätzlich mitgegeben werden um einen Kontext genauer zu definieren.

Durch die Schlagworten ist es möglich die Reputationinformationen von Widgets und externen Quellen nicht nur plattformgebunden anzuführen, sondern z.B. auch nach Themen zu gliedern. Hierdurch entsteht der Mehrwert, dass der Student oder ein anderer Benutzer sehen kann, wo seine Stärken oder Fokus liegen. Dies kann vor allem zur Selbstreflexion dienen und somit auch dazu motivieren den eigenen Horizont zu erweitern, indem man versucht, in möglichst vielen Gebieten eine gute Reputation zu erlangen. Schlagworte können auch bei der Gruppensuche für Übungen dazu beitragen, dass sich Studenten mit ähnlichen oder ausgleichenden Fähigkeiten leichter zusammenfinden und so der Lerneffekt dieser Lehrveranstaltungen verbessert werden kann.

Die Beschlagwortung kann auch dazu herangezogen werden, Beiträge zu einem bestimmten Thema zentral auf einer weiteren Plattform darzustellen, welche auf die Daten von TURep zurückgreift. Somit könnte man die Fragmentierung der Diskussionsforen innerhalb der TU Wien etwas abmildern, indem man deren Inhalte zusammenführt. Dies könnte unter anderem auch die Lehre verbessern, da die selben Fragen von Studenten nicht an mehreren Stellen erneut gestellt und beantwortet werden müssen oder die Lehrenden überhaupt erst darauf aufmerksam gemacht werden, dass die Frage gestellt wurde, da sie die entsprechenden Foren sonst nicht aktiv betreuen.

### **Umgang mit Gruppenarbeiten**

Ein etwas heikles Thema, ist meiner Meinung nach, der Umgang mit Gruppenarbeiten, da sich die daraus ergebende Reputation nicht 1:1 auf jedes Gruppenmitglied übertragen lässt. Da es aber beim Informatikstudium und auch bei anderen Studiengängen, immer wieder Gruppenarbeiten mit verteilten Aufgabengebieten gibt, sollte diese Problematik nicht vernachlässigt werden.

Hier kommen auch wieder die Schlagworte ins Spiel. Mein Vorschlag wäre es, dass bei der Verknüpfung des Studenten mit dem Projekt, mit Hilfe der Schlagworte angegeben wird, was dieser zu dem Projekt beigetragen hat. Dies setzt natürlich voraus, dass diese Angabe ehrlich bzw. überhaupt von den involvierten Studenten gemacht wird. Bei einer etwas komplexeren Handhabung von Gruppenprojekten könnte man auch noch vorsehen, dass alle Gruppenmitglieder sich gegenseitig die Schlagwörter bestätigen müssen. Das dürfte aber kaum Praxistauglich sein, da man so einen erhöhten zusätzlichen Aufwand hat und der Student am ehesten gar keine Schlagwörter angibt.

Das Arbeiten auch gut als Gruppenarbeit abgebildet werden können, setzt auch voraus, dass die Verknüpfungen jeweils auf das selbe Ziel verweisen. Sei es nun die URL zum Projektwebseite oder eine andere Form von Identifier. Auch eine zusätzliche Funktion, für das Zusammenführen von unterschiedlichen Verweisen zu dem selben Projekt wäre denkbar, sollte aber eher von den Plattformen gehandhabt

werden, die TURep einsetzen. Will man aber dennoch eine Zusammenführung erwirken, könnte man dies über ein gemeinsames Schlagwort, z.B. den Projekttitle, bewerkstelligen.

### Reputation Widgets

Mit dem Angebot von Reputationwidgets kann man bereits bestehende Plattformen einfach um Reputation erweitern, um so den Benutzern einen Anreiz zu geben, die Plattform zu verwenden. Diese sollen einerseits als einfache HTML oder JavaScript Snippets zu Verfügung stehen, aber auch als API, so dass die zugrunde liegende Datenerfassung auch bei selbst gestalteten Reputationwidgets verwendet werden kann. Sie stellen somit zwei mögliche Zwischenstufen zu einer vollständigen Integration einer Plattform als TURep Knoten dar.

Das **Like** Widget, wäre die einfachste Form eines Widgets und sehr ähnlich dem von Facebook zur Verfügung gestellten „Like“ Button. Es dürfte wohl die meisten Einsatzzwecke abdecken, bei denen eine Form von Reputation interessant wäre. Etwa beim Wertschätzen einer guten Arbeit in einem Portfolio oder zum Markieren eines guten Beitrages in einem Forum. Um noch flexibler eingesetzt werden zu können, sollte man beim Einbinden die Möglichkeit haben, den Wortlaut und visuelle Präsentation an die der eigenen Plattform anpassen zu können. Somit wäre es im Grunde auch möglich eine negative Bewertung zu implementieren. Hiervon sollte man aber eher absehen, da ein negatives Bewerten von den einzelnen Plattformen gehandhabt werden soll, da dieses meist dazu eingesetzt wird, anstößige oder andere unangebrachte Inhalte heraus zu filtern.



Das **N-Stern** Widget, ermöglicht es eine feinere Wertung in eine Plattform einzubauen. Diese dürfte vor allem bei Portfolio ähnlichen Systemen interessant sein, um den Studenten die Möglichkeit zu geben, Arbeiten feingradiger zu bewerten.



Das Widget wäre in der Anzahl der Bewertungsstufen flexibel. Somit wären 2-N Stufen möglich, wobei die Bewertung als Wert zwischen 0-1 gespeichert werden würde. Somit kann man im Nachhinein die Menge an möglichen Wer-

ten einfach ändern, falls man bisher etwa zu viele oder zu wenige Abstufungen hatte, ohne bereits getätigte Bewertungen wegwerfen zu müssen.

Zusätzlich könnte das N-Stern Widget auch frei für die Zielplattform definierbare Unterbewertungen mit dazugehörigen Gewichtungen für die Gesamtwertung definieren. Dadurch ist eine noch granularere Bewertung, wie etwa die Verkäuferbewertung auf eBay möglich.

Beide Widget Varianten sollten auch die Möglichkeit bieten einen Kommentar zu hinterlassen, um so die Bewertung besser nachvollziehbar zu machen und dem Bewerteten konstruktives Feedback zu geben. Diese Kommentare könnte wiederum, ähnlich wie auf Amazon auch optional bewertet werden um so eine gute Kommentar Kultur zu fördern.

Auch eine Funktion, Widgets zu melden, ist wichtig um Betrugsfälle möglichst vorzubeugen oder schnell aufzudecken und zu klären. So können etwa Benutzer mit einem einfachen Klick auf eine Schaltfläche und eine Texteingabe den Moderatoren mitteilen, dass Sie der Meinung sind, dass ein Widget beispielsweise zu Unrecht von einem Benutzer bei einer Arbeit von jemand anders eingebunden wurde.

Sollte sich herausstellen, dass diese Meldung gerechtfertigt war, würde der betroffenen Benutzer auf jeden Fall seine erhaltenen Bewertungen durch das Widget verlieren. Zur Abschreckung könnte man in diesem Fall auch noch auf negative Reputation setzen.

Der Benutzer der das Widget gemeldet hat könnte je nach Ausgang ebenfalls eine positive oder negative Wertung erhalten. Dies könnte etwa so wie das Trillerpfeifen System auf [newgrounds.com] funktionieren oder sich auf eine TURep weite Verhaltensreputation auswirken.

Bewertungen im Nachhinein ändern zu können, wäre in manchen Kontexten eventuell auch eine wünschenswerte Option. Ein Beispiel hierfür wäre eine Übung, bei der man ein Projekt in einem ersten Schritt prototypisch umsetzt und dann, nach einer ersten Feedbackrunde weiter verfeinert, mit einer abschließenden weiteren Feedbackrunde.

Diese Feedbackrunden könnte man mit Widgets online realisieren.

Um Bewertungen dennoch nachvollziehbar zu machen, sollte TURep jede „Bewertungsrevision“ aufbewahren. Diese Revisionen könnten dann auch statistisch und visuell aufbereitet werden, um so eine auf Wunsch noch feinkörnigere Reputationinformation zu bieten.

Für den Einsatz der Widgets gibt es vorerst zwei unterschiedliche Varianten und zwar eine plattform- sowie eine projektfokussierte.

Bei der plattformfokussierten Variante liegt der Schwerpunkt darauf, eine ganze Plattform einheitlich um eine Reputationfunktion zu erweitern und ist grundsätzlich nur für TU Wien interne oder nahe Plattformen gedacht, z.B. auf Projektseiten in einem Portfolio. Das Einbauen geschieht entweder über ein in das Frontend einbettbares HTML / Javascript Snippet oder über eine in das Backend integrierbare API.

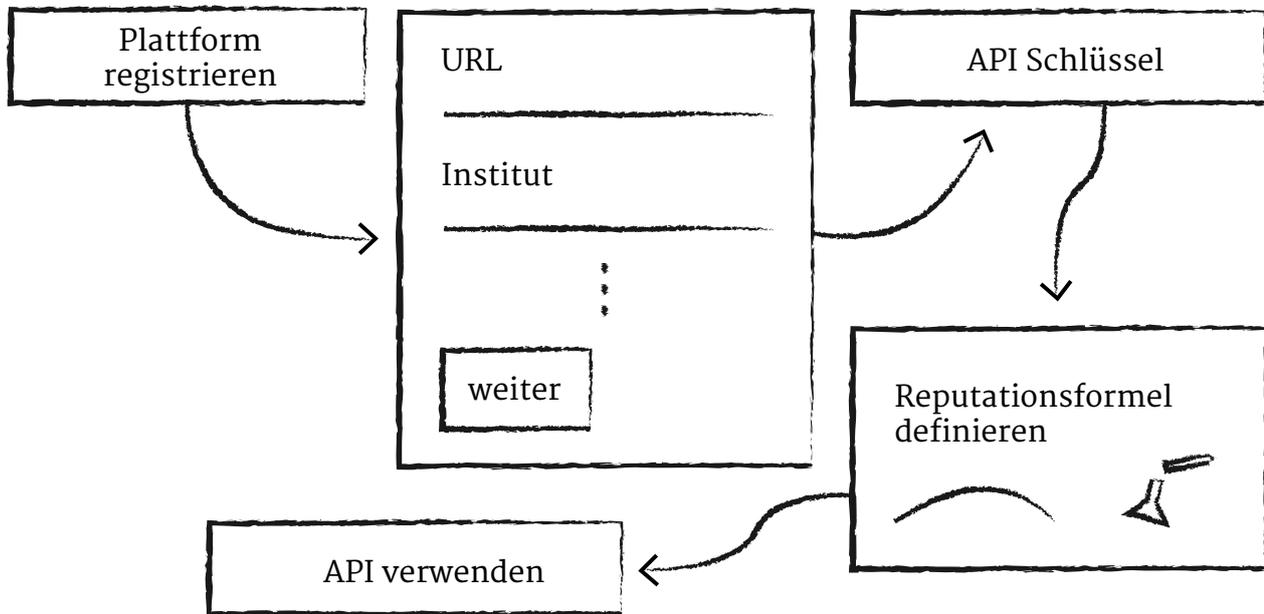
Die Abläufe um die API oder einbettbare Widgets einzusetzen, könnten wie folgt aussehen:

### Einbinden in eine Plattform

#### **Plattform registrieren:**

Bevor man TURep Widgets oder die API in seine Plattform integrieren kann, muss diese registriert werden. Dies geschieht indem man sich mit einem TU Wien Account anmeldet und die folgenden Angaben ausfüllt. Diese Liste an Angaben ist angelehnt an die nötigen Informationen, die man angeben muss, wenn man das TU Wien SSO in seinen Webservice einbinden möchte:

- **URL** - Eine Web URI zur Plattform, in die TURep eingebunden werden soll
- **Institut** - Das Institut der TU Wien, dass die Plattform zur Verfügung stellt
- **Kontaktperson**
- Kurze **Beschreibung** der Plattform



### API Schlüssel generieren:

Sobald man seine Plattform in TUREp eingetragen hat, wird automatisch eine API Schlüssel und ein Geheimnis, beides zufällige alphanumerischen Strings, generiert. Mit dem Schlüssel kann man sich nun mit der TUREp API verbinden und mit dem Geheimnis, die ausgetauschten Inhalte zusätzlich verschlüsseln.

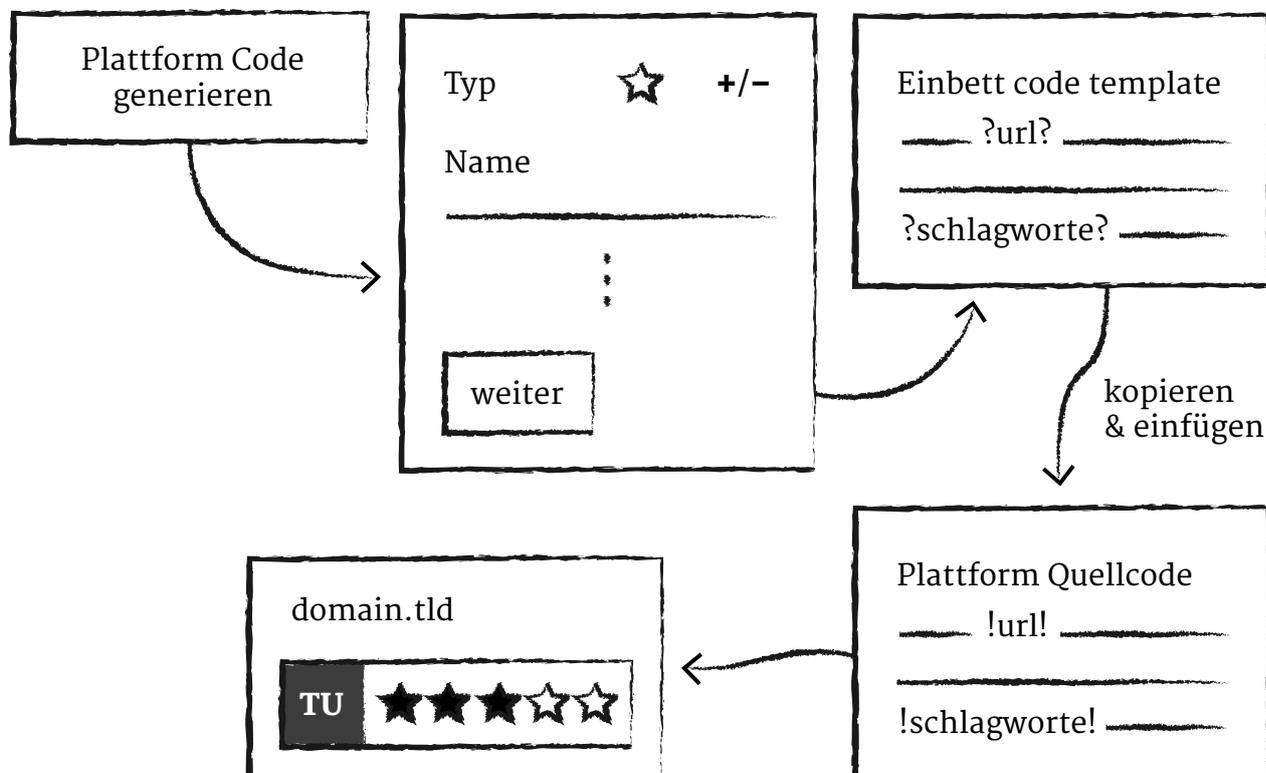
### Reputationformel definieren

Optional kann man für seine Plattform eine spezifische Reputationformel angeben. Innerhalb dieser Formel kann man auf vorberechnete statistische Werte oder auch direkt auf alle Bewertungen der Plattformwidgets zurückgreifen und zu einem oder mehreren Werten zusammenfassen. Dies könnte von der Funktionalität ähnlich wie MapReduce [Dean & Ghemawat, 2004], einem Verfahren zur Verarbeitung von großen Datenmengen, funktionieren, welches beispielsweise in Objektdatenbanken wie CouchDB eingesetzt wird, indem die jeweiligen Schritte für Map & Reduce als Javascript Funktionen definiert werden. Um dies etwas Benutzerfreundlicher zu gestalten, könnte man auch auf grafische Programmierung, in Form von Graphen zurückgreifen.

Wenn man keine eigene Reputationformel definiert, wird die Reputation aus einem Mittel aller erhaltenen N-Stern Bewertungen sowie Likes berechnet. Wobei die Anzahl an Likes ebenfalls in ein Intervall von 0-1 abgebildet werden, indem die Zahl an Likes mit einer Normalverteilung gewichtet wird. Diese Normalverteilung wird mit einem

Mittelwert und Variation aller plattformspezifischen Gesamtlikes erstellt. Somit kann sich der Like Anteil der Reputation anfangs noch laufend ändern, sollte sich aber nach einer gewissen Zeit einpendeln.

### Widget embedcode erstellen:



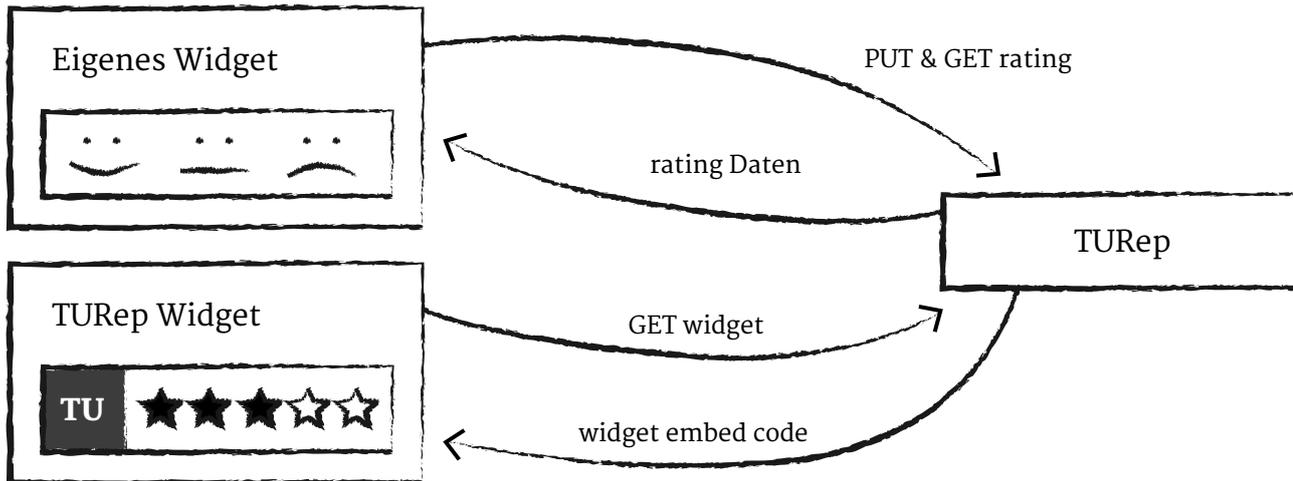
Wenn man die API nicht direkt einsetzen möchte, kann man die TURep Widgets auch über einen Embedcode in seine Plattform integrieren. Hierzu kann man zuerst das Widget auf TURep wie folgt vorkonfigurieren:

- **Widgettyp** (Boolean, N-Stern)
- **Kommentare** (Ja/Nein)
- **Name** - Ein beliebiger einzigartiger Name für dieses Widget. Dies ist nötig um auch mehrere unterschiedliche Widgets für eine Plattform generieren zu können.
- **Schlagworte** (optional) - Grunds Schlagworte die für alle Inhalte der Plattform zutreffend sind

Auf der eigenen Plattform muss man den Embedcode noch um ein paar Parameter ergänzen:

- **Autor/en** - die Objekt ID/s des/r TU Wien Accounts, der/die Bewertung erhalten soll/en
- **URL** - Eine eindeutige URI zu dem zu bewertenden Inhalt.
- **Schlagworte** (optional) - Zu dem zu bewertenden Inhalt passende Schlagworte

### API verwenden:



Die API könnte in Form einer simplen REST/HTTP API implementiert werden und wäre somit relativ einfach in jeder beliebigen Umgebung einzubinden. Auf jeden Fall sollte es für die gängigste Umgebungen wie PHP, Java, Python und Ruby Referenzimplementationen geben, um so das Einbinden zu erleichtern und die Wahrscheinlichkeit zu erhöhen, dass das Service auch eingesetzt wird..

Über die API kann man entweder direkt Bewertungen abschicken oder den Embedcode für ein bereits konfigurierbares Widget abfragen und um die restlichen nötigen Informationen erweitern. Alle von TURep gespeicherten Daten sind unter dem APIKey abgekapselt, so dass man nur die Daten abfragen kann, die man auch selbst abgelegt hat. Die folgenden Aktionen sollten auf jeden Fall möglich sein:

- **PUT rating** (APIKey, URI, Bewerter, Bewertungstyp, Bewertung, Kommentar)

Speichert eine *Bewertung* in TURep. Sollte ein Aufruf mit den selben Parametern noch einmal an TURep gesendet werden, würden die bereits existierenden Information überschrieben werden. Somit wäre es möglich Bewertungen im Nachhinein zu korrigieren. Ob diese Möglichkeit auch an den Benutzer weitergegeben wird, muss bei der jeweiligen Plattform, die auf die API zugreift, implementiert werden.

- **GET rating** (APIKey, URI, Bewerter, Bewertungstyp (optional))

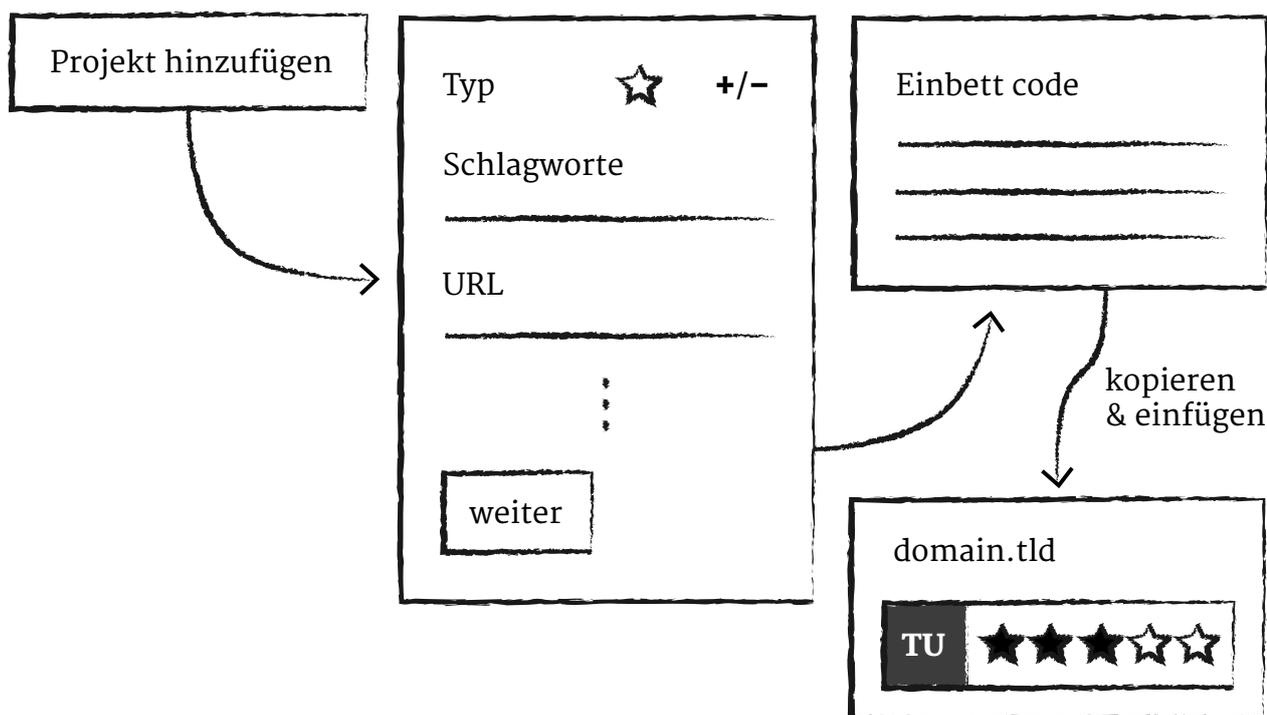
Holt eine beliebige Bewertung von TURep, die zuvor abgelegt wurde.

- **GET totalrating** (*APIKey, URI*)  
Holt die gemittelte Bewertung aller abgelegter Bewertungen unter der *URI*
- **GET widget** (*APIKey, Name, Autor, URI, Schlagwörter*)  
Holt den mit den angegebenen Parametern befüllten Embedcode des Widgets
- **PUT widget** (*APIKey, Name, Typ, Kommentare, Schlagwörter*)  
Erstellt ein neues Widget
- **PUT comment** (*URI, Autor, Kommentar*)  
Speichert ein Kommentar, ohne quantitative Bewertung
- **GET comments** (*URI*)  
Holt alle Kommentare zur *URI*
- **PUT comment\_rating** (*Kommentar ID, Bewertender*)  
Speichert eine Bewertung zum Kommentar
- **GET platform\_reputation** (*Benutzer*)  
Holt die plattformsspezifische Reputation des Benutzers

### **Einbinden in einen beliebigen projektfokussierten Kontext**

Die projektfokussierte Herangehensweise erlaubt es Benutzern, einzelne Projekte oder Arbeiten gezielt mit einem der beiden Widgets zu versehen. So könnte beispielsweise ein Student, eine selbst erstellte Webseite für sein Projekt einfach um das Like Widget oder sogar ein N-Stern Widget erweitern. Mittels der optionalen Kommentarfunktion der Widgets, kann sich der Student dann auch noch Feedback zu seiner Arbeit holen. Bei den Widgets muss es sich auch nicht gezwungenermaßen um einen visuellen Button auf der jeweiligen Seite handeln. Im Grunde würde es genügen, eine simple URL aufzurufen, um eine Bewertung direkt abzugeben oder um auf ein Interface zu gelangen, welches es ermöglicht eine Bewertung abzugeben. Dies könnte man etwa auch so nutzen, dass man die URL als QR Code zur Verfügung stellt und reale Objekte mit diesen versieht. Somit wären beispielsweise Plakate oder architektonische Modelle über TUREp ebenfalls bewertbar.

## Widget Embedcode erstellen:



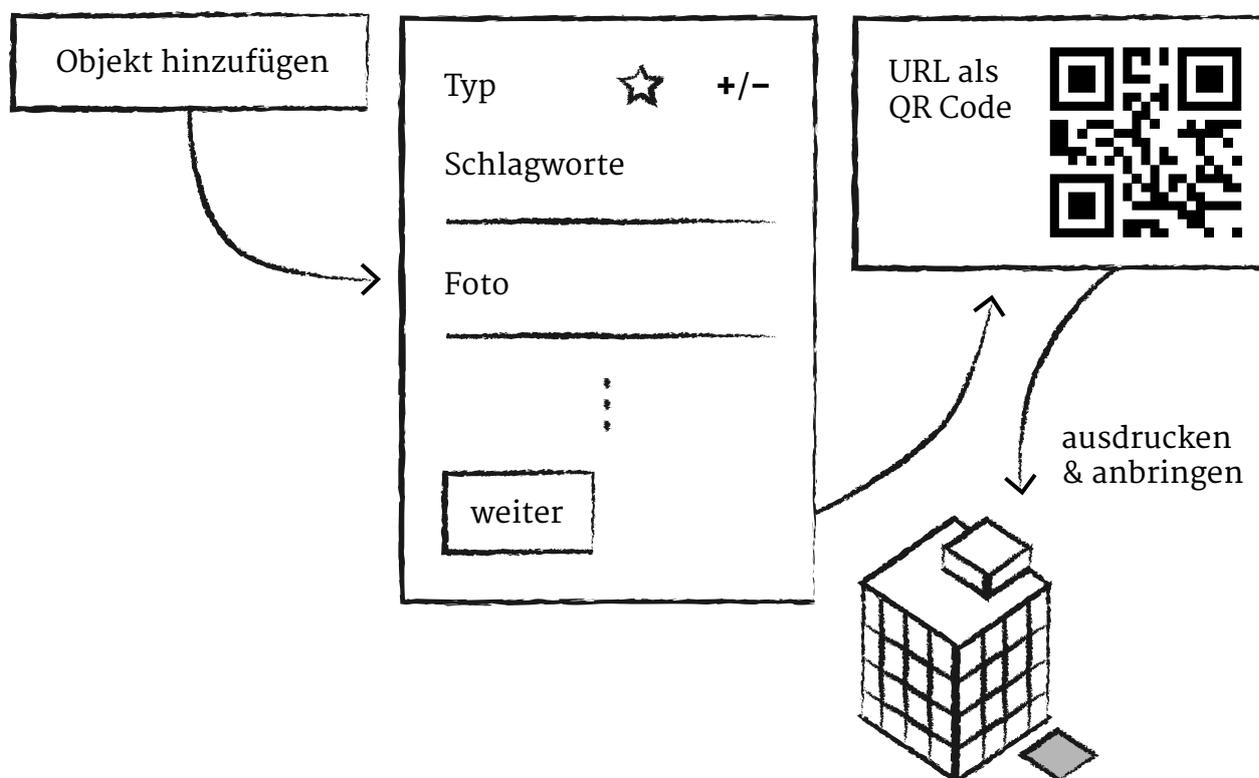
Jede Person, die einen TU Wien Account besitzt, kann bei TURep ein Widget erstellen, um ein beliebiges Objekt bewertbar zu machen. Dazu muss man die folgenden Dinge konfigurieren:

- **Widgettyp** (Boolean, N-Stern)
- **Kommentare** (Ja/Nein)
- **URL** (optional) - Eine eindeutige URI zum zu bewertenden Objekt.
- **Schlagworte** (optional) - Schlagworte, die das zu bewertende Objekt beschreiben.
- **Mitwirkende** (optional) - Namen, Objekt IDs der TU Wien Accounts oder Matrikelnummer von anderen Personen, die an dem zu bewertenden Objekt mitgearbeitet haben.

Daraufhin erhält der Benutzer einen HTML Embedcode, welchen dieser auf einer beliebigen Seite einbauen kann. Wenn der Benutzer keine URL angegeben hat, verwendet das Widget, beim Speichern einer Bewertung die URL der Seite, in die dieses eingebaut wurde. Bei jeder Bewertung wird auch jeweils extra die URL mit gespeichert, unter der das Widget aufgerufen wurde. Somit kann man auch verhindern, dass Benutzer das selbe Widget an mehreren Orten einbauen und so etwa beim Boolean Widget eine höhere Reputationwertung erzielen könnten.

Da sich ein Widget durchaus auch auf TU Wien externen Plattformen befinden kann, wird es nur dann voll dargestellt, wenn der Benutzer bereits im TU Wien SSO angemeldet ist. Sollte dies nicht der Fall sein, wird an Stelle des Widget nur einzelner Button angezeigt, über den der Benutzer zu einem TU Wien SSO Anmeldefenster und nach erfolgter Anmeldung zu einem Bewertungsformular gelangt. Dies kann mittels Javascript, AJAX und Modalfenstern auch innerhalb der selben HTML Seite passieren. Eine Bewertung ohne beim TU Wien SSO eingeloggt zu sein ist auch möglich, indem man das Anmeldefenster ignoriert. Diese Bewertung wird aber schwächer gewichtet und die IP Adresse des Benutzers danach für Bewertungen unter dieser URL blockiert. Diese Blockierung dient dazu, das Erschummeln von Bewertungen zu erschweren, in dem sich ein Benutzer im TU Wien SSO ausloggt und dann die eigene Arbeit mehrmals gut bewertet.

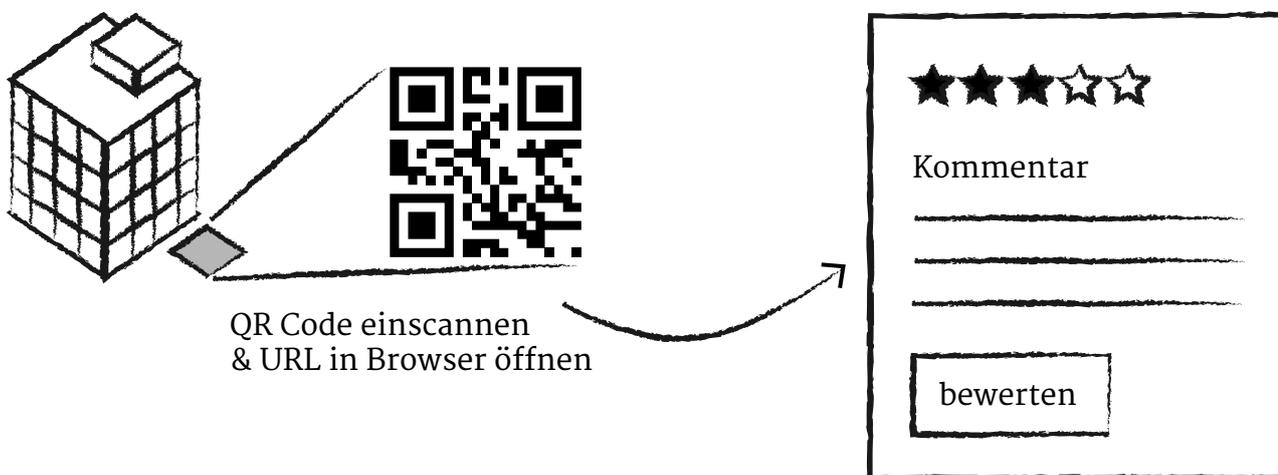
#### QR Code / URL erstellen:



Das Erstellen eines QR Codes oder Links funktioniert ähnlich wie das eines Embedcodes, aber mit dem Unterschied, dass man am Ende keinen HTML Code erhält, sondern eine URL im Reintext oder als QR Code encodiert. Der Benutzer müsste die folgenden Informationen eintragen:

- **Widgettyp** (Boolean, N-Stern)
- **Kommentare** (Ja/Nein)
- **Titel** - Der Titel des zu bewertenden Objekt
- **Bild** (optional) - Ein Bild des zu bewertenden Objekts
- **URL** (optional) - Eine eindeutige URI zum zu bewertenden Objekt.
- **Schlagworte** (optional) - Schlagworte, die das zu bewertende Objekt beschreiben.
- **Mitwirkende** (optional) - Namen, Objekt IDs der TU Wien Accounts oder Matrikelnummern von anderen Personen, die an dem zu bewertenden Objekt mitgearbeitet haben.

Wenn nun ein Benutzer die generierte URL in einem Browser öffnet, gelangt dieser auf eine von TURep generierte Seite, auf der das gewünschte Widget mit den konfigurierten Informationen angezeigt wird. Sollte der Benutzer nicht bereits im TU Wien SSO eingeloggt sein, wird ein Formular zum Anmelden vorgeschaltet. Wie beim regulären direkt eingebunden Widget, kann das Objekt auch bewertet werden, ohne eingeloggt zu sein.



Allgemein zu Widget Embedcode sowie QR Code wäre zu überlegen, ob der Benutzer nicht verpflichtend einen „Beweis“ in Form eines Bildes oder PDF hochladen muss, um somit explizit zu hinterlegen, was denn nun bewertet werden soll. Dies würde vor allem Moderatoren und anderen Benutzern dabei helfen, die Gültigkeit der Widgets oder QR Codes zu verifizieren.

## Kärtchen

Bisher bin ich nur darauf eingegangen, wie Reputationinformationen von TURep selbst erfasst werden können.

TURep soll aber auch die Möglichkeit bieten, die Reputation eines Benutzers von weiteren TU Wien externen Plattformen zusammenzufassen.

Um die gesammelte Reputation eines Studenten oder Professors gut darstellen zu können, schlage ich vor auf ein Kärtchen basiertes Interface zu setzen, visuell ähnlich dem, was bereits im Badge Prototyp eingesetzt wurde und wiederum stark von Google+ und Google Now beeinflusst ist. Bei diesem Interface würden alle zusammengehörende Reputationinformationen innerhalb eines Kärtchens dargestellt werden. Wobei die Zusammengehörigkeit flexibel definierbar sein sollte. Auf jeden Fall sollte es zumindest eine Karte je Plattform geben. Es wäre aber auch interessant Kärtchen zu haben, die Anhand von Schlagworten oder der Art von Beiträgen plattformübergreifend zusammengefasst sind.

Damit externe Benutzer, welche die jeweiligen Plattformen eventuell nicht kennen, die Reputation besser einschätzen können, wäre eine automatisch generierte Zusatzinformation interessant, die dem Benutzer mit einem einzelnen Wert Auskunft darüber gibt, wie gut die Reputation des Studenten denn im Verhältnis zu anderen Benutzern auf dieser Plattform ist. Die Information dazu, wie gut die dargestellte Reputation denn nun in Relation zu anderen Benutzern ist, sollte auf jeden Fall nur die Reputation anderer Studenten der TU Wien berücksichtigen, die auf dieser Plattform aktiv sind. Dies wäre etwa bei einer Plattform wie Stackoverflow durchaus angebracht, da sich ein Student voraussichtlich nie so stark einbringen kann, wie ein professioneller Programmierer mit mehreren Jahren Erfahrung. Sollte sich in der Praxis herausstellen, dass es einfach zu wenige Studenten gibt, welche die selbe Plattform verwenden, könnte auch durch die TURep Betreuer ein gewisser Referenzwert vorgegeben werden.

Neben dieser relativierenden Maßnahme, sollte es auch noch eine zusätzlich einblendbare Information geben, die dem Benutzer kurz erklärt, worum es auf der dargestellten

Plattform geht und wie der Reputationwert zu verstehen ist.

Der Benutzer soll auch steuern können, welche seiner Kärtchen öffentlich einsehbar sind und in welcher Reihenfolge diese angezeigt werden. Es wären auch Kärtchen mit komplett anderen Inhalten denkbar, um so den Benutzer die Möglichkeit zu geben ihr TURep Profil als allgemeines Profil oder gar gesammeltes Portfolio zu verwenden. Dies geht aber definitiv über die Fragestellung dieser Arbeit und eventuell auch die Zielsetzung von TURep hinaus.

Als weitere Möglichkeiten die Reputationdaten zu präsentieren, wurden die folgenden Varianten zwar bedacht, dann aber wieder verworfen:

Eine nach **Schlagworten gruppierte Liste**. Diese wäre so angedacht gewesen, dass die Reputationinformationen gruppiert zu jedem Schlagwort je Plattform angezeigt würden. Das Problem bei dieser Ansicht wäre gewesen, dass man bei vielen Schlagworten sehr schnell den Überblick verloren hätte.

Eine weitere Variante wäre eine **Liste von Objekten** gewesen, die eine Reputation erhalten haben. Beispielsweise eine Liste aller Schlagworte mit Reputationwert, eine Liste aller Badges, eine Liste aller guten Kommentare. Das Problem bei dieser Variante wie auch bei der vorherigen ist, dass die Präsentation relativ unflexibel ist was externe Reputationquellen angeht. Würde man die Reputationquellen nur auf ein fixes Set beschränken, wäre diese Art der Präsentation aber durchaus attraktiv.

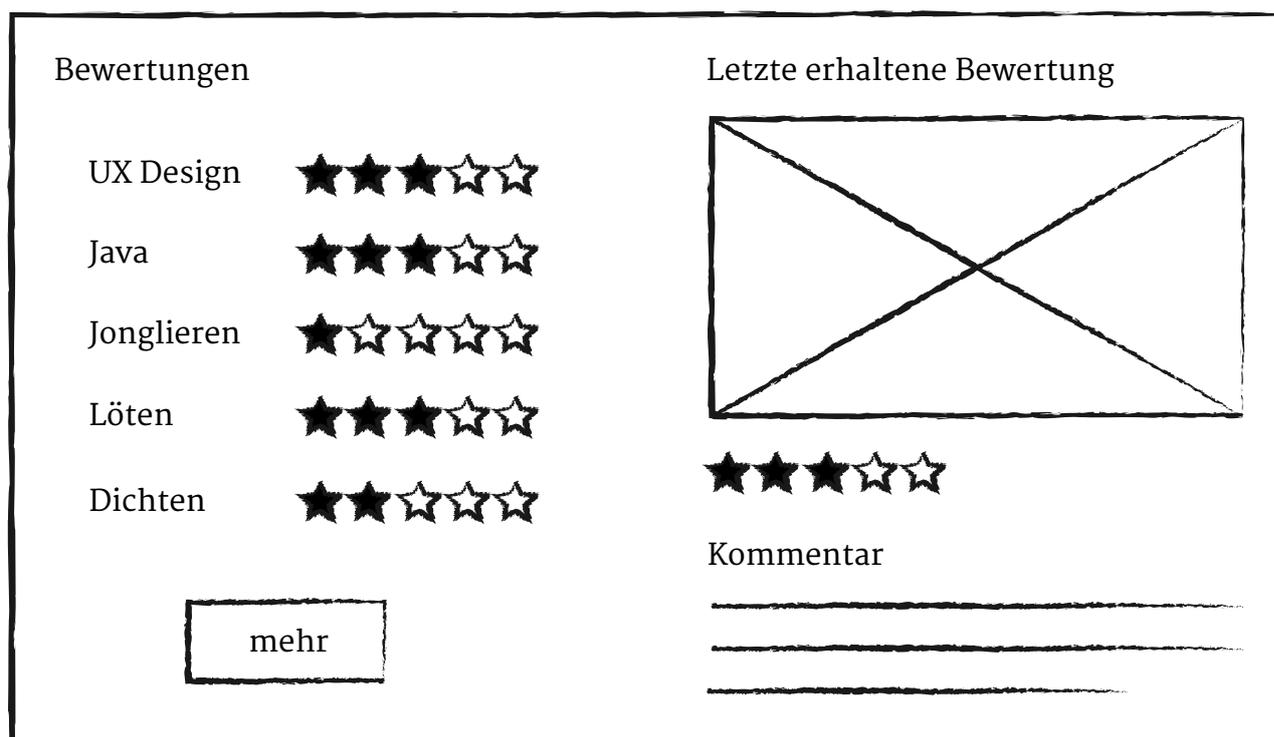
Eine weitere Ansicht die verworfen wurde wäre der **Timeline** von Facebook sehr ähnlich gewesen, indem Sie die erhaltenen Bewertungen des Benutzers auf einer vertikalen Zeitachse aufgetragen hätte. Dies würde zwar grundsätzlich mit externen Quellen funktionieren, es müsste aber immer der aktuelle Stand zentral von TURep gespeichert werden und dann regelmäßig mit der externen Plattform abgeglichen werden um Veränderungen zu ermitteln und eine konsistente Reihung zu garantieren.

Der eigentliche Vorteil der Kärtchen gegenüber der verworfenen Konzepte ist der, dass für jedes Service, das ein-

gebunden werden soll, ein eigenes Kärtchen gestalten und so die Reputationinformation optimal aufbereitet dargestellt werden kann.

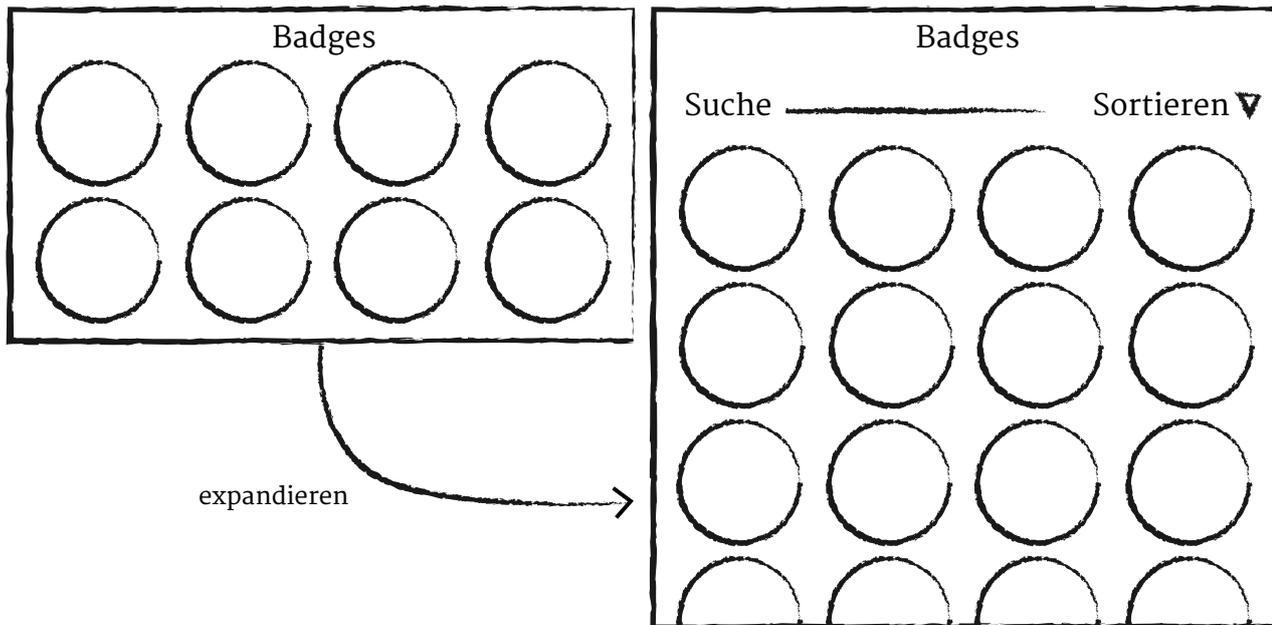
Hier sind ein paar konzeptionelle Beispiele und Mockups für Kärtchen zu den TURep eigenen Widgets, sowie unterschiedlichen Plattformen, die ich einleitend zu dieser Arbeit bereits als Beispiele für Plattformen mit Reputation Systemen beschrieben habe.

### Widgets



Um die, über TURep Widgets erhaltenen, Bewertungen durch ein Kärtchen darzustellen, gäbe es mehrere Möglichkeiten. Die für mich naheliegendsten Variante wäre es, auf eine Gruppierung anhand der Schlagworte zurück zu greifen. Anhand der gemittelten Wertungen bei einem Schlagwort könnte man einfache Darstellungen wie ein Balken-, Stern- oder Tortendiagramme generieren. Platzbedingt müsste man aber die Auswahl an Schlagworten auf einem Kärtchen beschränken. Diese könnte der Benutzer selbst einstellen oder TURep ermittelt die Top X automatisch. Man könnte auch die letzte erhaltene Bewertung anzeigen, um so etwa mehr Fokus auf die eigentlichen Projekte zu lenken.

## Open Badges



Dieses Kärtchen würde es einem Benutzer ermöglichen, seinen Open Badges Backpack darzustellen. Hierzu kann dieser auf externe Plattformen verweisen um die Badgeinformationen dort abzuholen oder auch Badges direkt hochladen. Mit diesem Kärtchen könnte man auch TU Wien interne Badges, vom vorgestellten Badge Prototypen darstellen, insofern bei diesem noch Open Badges Unterstützung implementiert wird.

Das Kärtchen zeigt, in seiner minimierten Form, standardmäßig die zuletzt erhaltenen Badges des Benutzers an. Der Benutzer hat aber auch die Möglichkeit, gewisse Badges zu fixieren, so dass diese immer vorgereicht werden. Dieses Kärtchen verfügt auch über eine maximierte Ansicht, in der man die Badges anhand von eventuell vorhandenen Schlagwörtern oder nach der Ursprungsplattform gruppieren und filtern kann. Wenn man auf einen Badge klickt, bekommt man alle verfügbaren Detailinformationen angezeigt, sowie einen eventuell vorhandenen Link zur ursprünglichen Plattform. Diese Funktionen müssten aber nicht direkt von TURep implementiert sein, sondern könnten von einem TU Wien eigenen Badge System, ähnlich dem bereits beschriebenen Badge Prototypen, zugespült werden.

Da ein TURep Benutzer Badges von sehr vielen unterschiedlichen Quellen sammeln kann, wäre es durchaus sinnvoll eine Möglichkeit zu bieten, mehrere Kärtchen mit

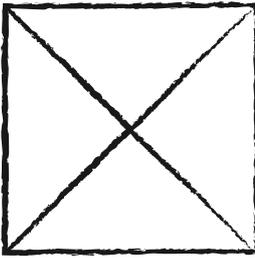
einer gewissen Vorfilterung erstellen zu können. So könnte der Benutzer sein TURep Profil auch nach Themengebieten gruppieren und somit dem Besucher einen besseren Überblick ermöglichen.

### Schlagworte



Die Schlagworte kommen ja bereits bei den Widgets Kärtchen sowie beim Open Badges Kärtchen zum Einsatz. Dennoch wäre es eine Interessante Variante für ein Kärtchen einfach nur alle Schlagworte aufzulisten, mit denen ein Benutzer über Bewertungen bereits in Kontakt gekommen ist. Das könnte etwa in einer mittlerweile schon sehr verbreiteten Form einer Tag Cloud passieren, bei der die Schlagworte anhand des jeweiligen Gewichtes unterschiedliche groß dargestellt werden. Wobei die Gewichte etwa aus der Menge der erhaltenen Badges sowie der Menge an Widgets je Schlagwort errechnet werden könnten.

Stackoverflow



Username  
**12.012**

gold	2
silber	21
bronze	39

[SO Profil ansehen](#)

Top Fragen

- 5 \_\_\_\_\_
- 2 \_\_\_\_\_
- 4 \_\_\_\_\_
- 1 \_\_\_\_\_
- 10 \_\_\_\_\_
- 7 \_\_\_\_\_

Top Antworten

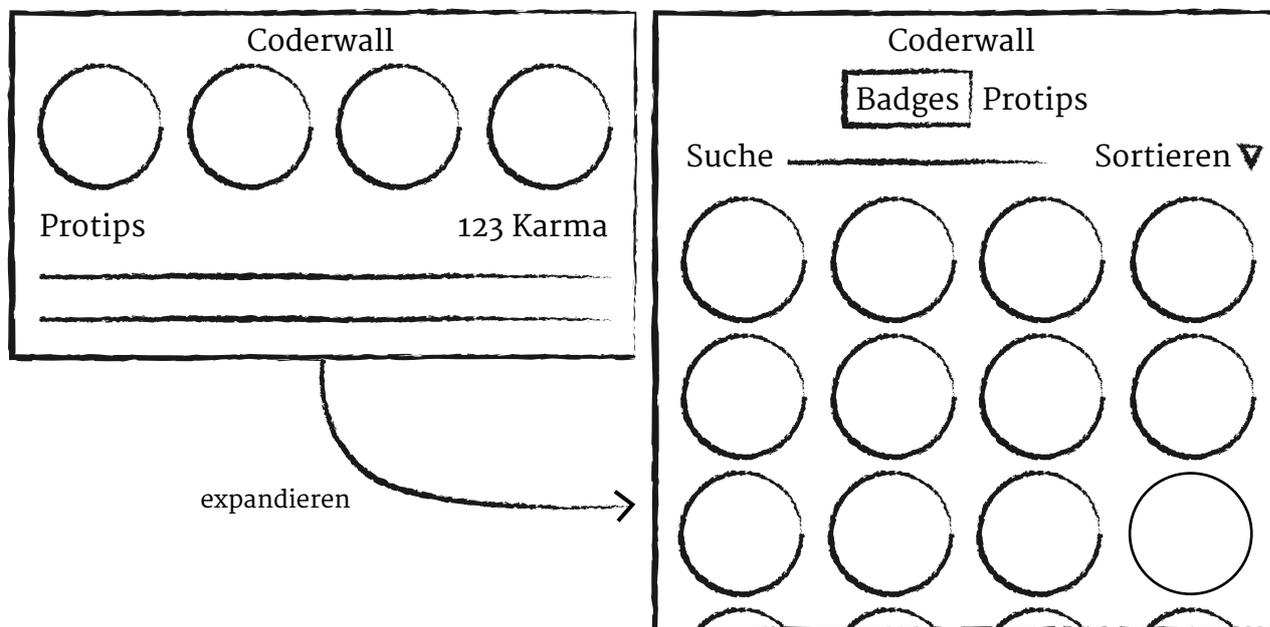
- 5 \_\_\_\_\_
- 2 \_\_\_\_\_
- 4 \_\_\_\_\_
- 1 \_\_\_\_\_
- 10 \_\_\_\_\_

Eine Karte für Stackoverflow wäre recht direkt umsetzbar. Beispielsweise als eine leicht modifizierte Variante des Profils, des hinterlegten Stackoverflow Accounts. Die dargestellten Informationen würden aus der Gesamtreputation sowie der Summen der Gold, Silber & Bronze Badges bestehen. Zusätzlich könnte man noch eine gewisse Anzahl von TU Wien relevanten Badges als Liste anführen. Welche Badges im Kontext der TU Wien relevant sind, könnte man ermitteln, indem man die Badges aller Verknüpfungen zwischen TU Wien Accounts und Stackoverflow Accounts vergleicht und aus der Summe aller Badges und der Summen je Badgetyp Gewichte errechnet.

Die dafür nötigen Daten könnten einfach mittels der Stackexchange API abgerufen werden und von TURep optional zwischengespeichert werden, um nicht von der Stackexchange API wegen zu starkem Traffic ausgesperrt zu werden. Da Stackexchange mehrere Plattformen anbietet und der selbe Account auf mehreren von diesen verwendet werden kann, bietet die API auch eine Filtermöglichkeit an, mit der man explizit nach Inhalten einer Plattform suchen kann. Im Rahmen von TURep könnte man nun auf zwei unterschiedliche Arten damit umgehen. Entweder man bietet eine Karte für jede Stackexchange Plattform an oder man fasst alle in einer Karte zusammen. Erstere Variante wäre

meiner Meinung nach besser geeignet, da diese den Kontext der externen Reputation besser erhält, vor allem wenn der TUREp Benutzer tatsächlich mehrere der Stackexchange Plattformen intensiv verwendet.

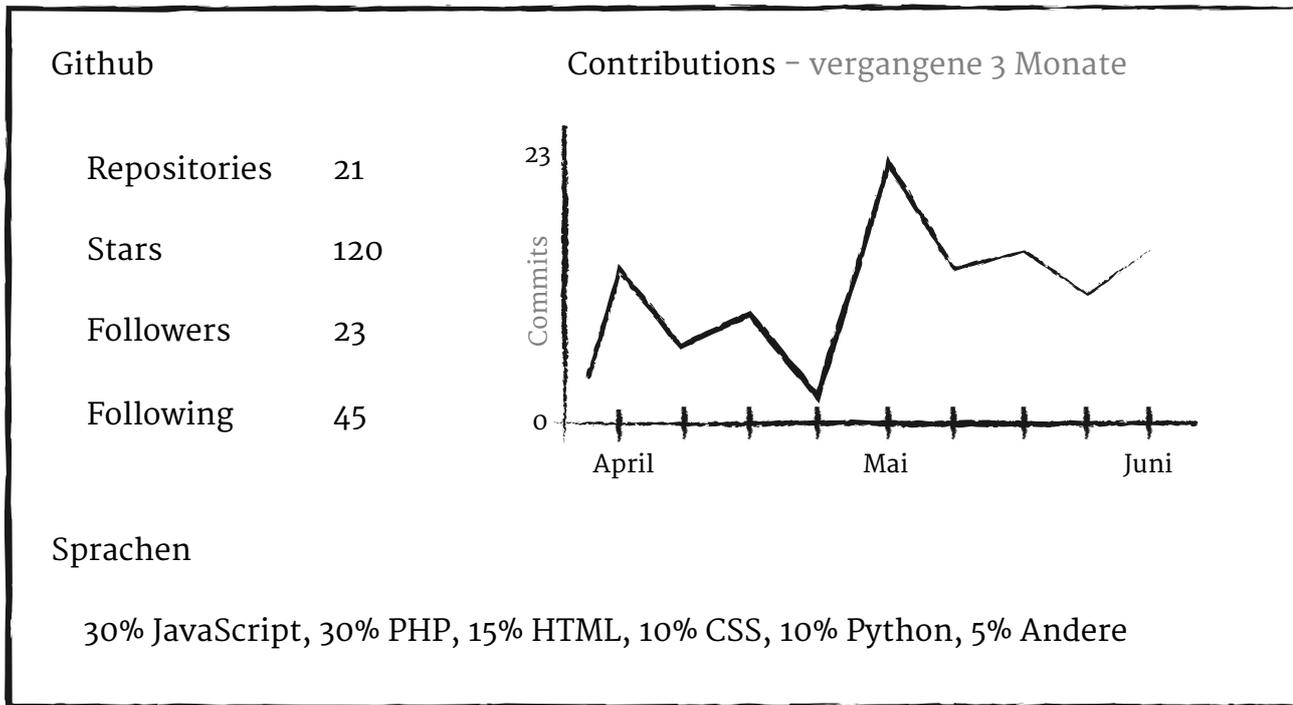
### Coderwall



Um den Badges Aspekt von Coderwall abzubilden, würde eigentlich auch die bereits beschriebene Open Badges Karte ausreichen, nur unterstützt Coderwall zum Zeitpunkt dieser Arbeit Open Badges leider noch nicht. Somit müsste man zu diesem Zweck ein eigenes Kärtchen implementieren. Coderwall bietet aber eine simple API an, mit der man alle Reputationinformation zu einem Benutzer als JSON enkodiertes Objekt abrufen kann.

Zusätzlich zu den Badges würde das Kärtchen auch noch die Menge an Pro-Tips, die der Benutzer veröffentlicht hat, darstellen.

## Github & Bitbucket



Ähnlich wie Coderwall könnte auch TUREp selbst ein Kärtchen anbieten, das Statistikinformationen zu den Repositories eines Benutzers darstellt. Als Beispiel seien an dieser Stelle die derzeit beliebten GIT Repository Hosters Github und Bitbucket angeführt. Vor allem ersterer bietet bereits innerhalb der eigenen Plattform übersichtliche Statistiken zur Aktivität eines Benutzers sowie zu den verwendeten Programmier- und Skriptsprachen. Diese Informationen könnte aber auch TUREp selbst direkt aus den Protokollen und Inhalten der Repositories herleiten.

### Umgang mit externen Plattformen

In diesem Abschnitt möchte ich ein paar Beispiele darlegen, wie TUREp in bereits bestehenden Plattformen der TU Wien sowie auch TU Wien externe Plattformen eingesetzt werden könnte. Ich werde auch auf eine konzeptionelle Plattform für Quests eingehen, die sich im Rahmen der Reputationserfassung im Universitätsalltag anbieten würde.

### TISS

Bei TISS (TU Wien Informations-Systeme und Services) handelt es sich um ein campusweites Informationssystem der TU Wien über das vorwiegend die Verwaltung der Lehrveranstaltungen stattfindet. Die Möglichkeiten für Studenten beschränken sich bei dieser Plattform vor allem auf administrativen Tätigkeiten bezüglich ihres Studiums.

Es werden aber auch die LVA-Bewertungen über TISS abgewickelt und es gibt auch die Möglichkeit, Diskussionen in LVA bezogenen Foren zu führen.

Die LVA-Bewertungen sind Fragebögen, die jedes Semester, von den Studierenden, für jede besuchte Lehrveranstaltung über TISS ausgefüllt werden können. Diese Bewertungen dienen dem einfachen Zweck, dass die LVA abhaltenden Professoren Feedback generieren können. Die Teilnahme an diesen Bewertungen ist aber relativ gering und auch die Sinnhaftigkeit der LVA Bewertungen wird öfters in Frage gestellt, da die vorwiegend quantitative Auswertung, kaum nützlichen Input für die Lehrenden liefert. An dieser Stelle kommt nun TURep ins Spiel. Mit Hilfe von Reputation könnte man die Studierenden dazu motivieren, die Fragebögen ausführlicher auszufüllen, indem Sie dafür etwa einen Badge erhalten oder sich eine TU Wien weite „Uni-Alltag“ Reputation dadurch verbessert. TURep könnte aber auch dazu eingesetzt werden, die Lehrveranstaltung selbst zu bewerten und zwar laufend, das ganze Semester über. Dies würde so ablaufen, dass Studenten jederzeit Feedback in Form von Bewertungen und Kommentaren geben können und diese Informationen von TURep erfasst und für den Professor aufbereitet werden. Dies hätte den Vorteil, dass die Qualität des Feedback gesteigert werden konnte und durch die zeitliche Nähe, dieses auch unmittelbar umgesetzt werden kann und somit, auch dem Feedback gebenden Studenten selbst, noch im laufenden Semester zu Gute kommen könnte.

## **TUWEL**

Bei TUWEL handelt es sich um die e-Learning Plattform der TU Wien. Die Plattform ist eine modifizierte Version des Open Source e-Learning Systems [Moodle]. Obwohl in Moodle selbst bereits eine Unterstützung für Mozillas Open Badges integriert ist, sowie eine Auswahl von anderen Reputationplugins verfügbar ist, werden diese Funktionen in TUWEL aktuell nicht genutzt.

Reputationmechanismen könnten in TUWEL vor allem dazu eingesetzt werden, das Erfüllen von Zusatzaufgaben oder auch das konstruktive Beitragen zu Diskussionen in den in TUWEL integrierten Foren zu belohnen.

Das Einbinden von TURep in TUWEL wäre vor allem auch deswegen interessant, da TUWEL selbst nur mit einem TU Wien Account zugänglich ist und man die Inhalte eines Kurses in TUWEL auch nur dann einsehen kann, wenn man in diesem angemeldet ist. Die Gruppe an Personen, die außergewöhnlichen Leistungen in einem LVA Kontext innerhalb dieser Plattform somit sieht, ist relativ klein, weshalb eine Motivation sich einzubringen eigentlich nur dann gegeben ist, wenn sich dies explizit positiv auf die Note auswirken kann.

Das TURep Like Widget könnte etwa dazu genutzt werden, gute Beiträge zu markieren. Dies muss nicht nur die Reputation des Studierenden verbessern, sondern kann auch den Lehrenden dabei helfen, nützliche Beiträge leichter zu finden und eventuell sogar mit einer besseren Note zu würdigen. Auf jeden Fall könnte man mit diesem einfachen Mitteln die Beteiligung innerhalb der Foren von TUWEL sehr wahrscheinlich verbessern.

Bei Badges wäre es wohl am naheliegendsten die bereits in Moodle integrierte Funktionalität zu verwenden. Dies würde dann zwar nicht direkt über TURep abgewickelt werden, TURep könnte aber, mit einer voll ausgebauten Open Badges Unterstützung inklusive Badge Backbags, diese Badges dann problemlos importieren und darstellen.

### **Online Portfolio**

An der TU Wien wird bereits an der Thematik eines Online Portfolios geforscht. Es handelt sich dabei zwar größtenteils um eine komplexe in sich geschlossene Problemstellung, Reputation kommt aber dennoch zum Einsatz, um Studierenden und Lehrenden die Möglichkeit zu geben, Arbeiten zu bewerten oder Feedback zu geben.

Hier würde TURep ins Spiel kommen und könnte entweder als Reputationsystem komplett eingebunden werden oder mit Hilfe von Kärtchen die eventuell schon vorhandenen Reputationinformationen aus dem Portfolio in darstellen. Letzteres hätte vor allem noch die zusätzliche Option, die Kärtchen so zu erweitern, dass man die Arbeiten selbst auch in TURep darstellen könnte. Sollte es also passieren,

dass man Arbeiten in mehreren TU Wien Portfolio Systemen hat, können diese mit Hilfe von TURep gesammelt dargestellt werden.

### **TU Wien Externe Quellen**

Grundsätzlich sollte TURep eine dezentrale Datenbank sein, bei der nur ein geringer Teil der dargestellten Reputationinformationen direkt von TURep gespeichert werden. Der Großteil der Informationen sollte von anderen, größtenteils TU Wien externen, Plattformen kommen. Dies kann entweder so passieren, dass die TURep Widgets eingesetzt werden oder aber auch TU Wien externe Plattformen eingebunden werden können.

Damit solche Plattformen angebunden werden können, müsste TURep eine SDK zur Verfügung stellen, die es Entwicklern ermöglicht eigenen Kärtchen zu programmieren, welche die Reputationsdaten dieser externen Plattform aufbereiten und darstellen. Ganz abgesehen von den technischen Aspekten dieser API besteht vor allem das Problem des Verifizierens einer Verknüpfung von einem anonymen Account auf einer externen Plattform und dem Personenbezogenen TU Wien Account. Wenn man hier keine adäquaten Mittel einsetzt, um dies zu kontrollieren, wäre es sonst recht einfach, dass ein Student die Reputation von einem, ihm unbekanntem Dritten, *stiehlt* indem er bei TURep vorgibt dieser Benutzer A auf Plattform B zu sein.

Eine Möglichkeit mit diesem Problem umzugehen, wäre es nur Accounts zuzulassen, die mit einem anderen OpenID ähnlichen Service verknüpft sind. So kann über diesen Service per E-Mail verifiziert werden, dass der TURep Benutzer zumindest in der Kontrolle über diesen Account ist.

Falls es die Möglichkeit der Verifikation über einen anderen Authentifizierungsservice nicht gibt, gäbe es noch die Option, Funktionen auszunutzen, die es dem Benutzer ermöglichen, sein eigenes Profil auf der Plattform zu gestalten. Dies könnte dann so ablaufen, dass TURep eine zufällige Zeichenkette generiert, die der TURep Benutzer dann in sein Profil, auf dieser externen Plattform in ein bestimmtes Feld eintragen muss. Wenn er dies getan hat,

überprüft TURep das Profil des Benutzers und akzeptiert die Verknüpfung, wenn er die Zeichenkette dort vorfindet.

Aber selbst diese Mittel können im Grunde nicht verhindern, dass man den Account einer dritten Person verwendet, die sich dazu bereit erklärt, zu kooperieren und die nötigen Schritte durchführt, damit TURep die Verknüpfung akzeptiert. Um das auch noch etwas abfangen zu können, muss auf einen menschlichen Moderator zurückgegriffen werden. So sollte es allen Benutzern möglich sein, Kärtchen anderer Benutzer zu melden, wenn Sie den Verdacht haben, dass deren Kärtchen nicht rechtmäßig zugewiesen ist. Dieser Verdacht müsste dann durch einen Moderator kontrolliert werden. Wenn sich der Verdacht erhärten sollte, kann der Moderator dann diese Verknüpfung wieder aufheben. Strafen wären auch denkbar, um so das Risiko eines Betruges von TURep zu erhöhen. Diese könnten von einer visuellen Markierung, bis hin zu dem Aufheben aller externen Verknüpfungen oder Löschen aller Reputationswerte reichen.

## **Anonymität & Privatsphäre**

Ein wichtiges Thema, welches man bei einer Plattform dieser Art nicht übersehen sollte, ist das Thema der Anonymität und Privatsphäre. Bei ersterem kann man natürlich sofort argumentieren, dass man sich in einem abgegrenzten Universitätskontext befindet und alle Benutzer des Systems echte Personen sind und auch als solche ausgewiesen und erkennbar sind und man somit das Thema Anonymität eigentlich Außen vor lassen könnte. Es gibt aber dennoch einige Argumente, die zumindest für eine teilweise Anonymisierung innerhalb des Systems sprechen.

Ein wichtiger Faktor ist, dass jegliche Form der Reputation nicht unmittelbar über die Matrikelnummer oder eine identifizierende Information zu der Person zugänglich sein sollte, wenn diese das nicht will. Dies wurde etwa beim Badge Prototypen so implementiert, dass man Badges explizit veröffentlichen muss, damit diese von anderen Personen eingesehen werden können. Zusätzlich sind die Listen an Badges nur über eine eigens vom Badge System

vergebene ID erreichbar und man kann somit nicht über die Matrikelnummer auf das Profil schließen.

Ein weiteres Beispiel wäre bei einem Portfoliosystem mit Reputation zu finden. Wenn es etwa die Möglichkeit gibt, bei einer Gruppenarbeit alle Mitwirkenden anzugeben, sollten die einzelnen Personen diese Angabe bestätigen und explizit veröffentlichen müssen. Wenn Sie dies nicht tun, würde der betreffende Mitwirkende zumindest anonymisiert angezeigt werden, damit der Kontext der Gruppenarbeit besser erhalten bleibt. Dies hat den Sinn, dass der Benutzer die Kontrolle darüber behält, mit welchen Projekten er in Verbindung gebracht wird.

Ebenfalls zu überlegen wäre ein Anonymisieren von Studenten untereinander, so fern sich diese gegenseitig bewerten können. Das würde primär dazu dienen um Revanchen zu verhindern. Ich persönlich denke aber, dass das nicht wirklich viel bringen würde, da man im Rahmen einer Lehrveranstaltung meist eine Idee davon haben dürfte, wer welche Arbeiten gemacht hat und diese somit, trotz Anonymisierung der echten Person zuweisen kann. Deswegen sollte das Bewertungssystem selbst so implementiert sein, dass es mit solchen Revancheverhalten umgehen kann.

Ein optionales Anonymisieren wäre aber dennoch angebracht, wenn etwa Professoren, im Rahmen einer Lehrveranstaltung, die Studenten sich gegenseitig bewerten lassen wollen. Diese Bewertungen sollten dann aber in einer temporären, auf die LVA beschränkte, Sandbox ablaufen, die vom Professor „moderiert“ wird. Es wäre also zu diskutieren ob es überhaupt Sinn macht, diese mit Hilfe von TURep abzuwickeln.

## **Opt-Out, Datenexport & Datenschutz**

Ein weiteres wichtiges Thema, das nicht übergangen werden sollte, wenn so viele Personenbezogene Daten verwaltet und aufbereitet werden, ist der Datenschutz, sowie die Möglichkeit wieder aus dem System auszusteigen.

Der Datenschutz sollte primär mit einer sicheren Server Infrastruktur gewährleistet sein. Diese ist auf der TU Wien vermutlich auch weitestgehend vorhanden, da bereits mehrere offizielle Services inklusive der Anmeldung zu LVAs und Benotung bereits über Web Interfaces gehand-

habt werden. Weitaus komplexer wird es aber, wenn ein Benutzer aus einem so stark vernetzten System austreten möchte. Hier stellen sich vor allem zwei Fragen. Wie geht man mit den von dem Benutzer getätigten Bewertungen um? Und in welcher Form ermöglicht man es dem Benutzer, seine Daten zu exportieren.

Die einfachste Antwort auf die erste Frage wäre wohl einfach die Löschung aller vom Benutzer getätigten Bewertungen. Dies hätte aber den Effekt, dass andere Benutzer im System Bewertungen verlieren würden. Eine etwas elegantere Lösung wäre es wohl, den Account nicht komplett zu löschen, sondern nur die personenbezogenen Daten zu entfernen. Dies wäre vor allem sehr einfach wenn man wie in [Müller et al] die Einträge nicht direkt mit der Person verknüpft, sondern diese Verbindung nur mittels eines Identity Providers möglich ist, welcher auch zum Datenschutz positiv beitragen würde. Somit müsste man bei der Abmeldung des Benutzers nur den Eintrag des Identifikation Servers leeren. Informationen dazu, ob es sich um einen Studenten oder Professor gehandelt hat, sollten eventuell erhalten bleiben, da diese auch etwas zum Wert der Reputation beitragen können.

Das exportieren der Daten des Benutzers gestaltet sich noch einmal etwas komplexer. Vor allem, da viele der Informationen ohne den richtigen Kontext kaum noch die selbe Wertigkeit haben. Was sich noch am ehesten exportieren lassen würde, sind etwa Badges, indem man diese in offenes Format wie etwa das von Open Badges [openbadges.org] konvertiert. Bei den Bewertungswidgets müsste man zwei Fälle wohl extra behandeln. Einmal wenn diese auf einer Plattform fix eingebaut sind. Hier könnte das System einfach eine Gesamtreputation für diese Plattform exportieren, bzw. eine Liste aller bewerteten Beiträge. Widgets die man als Benutzer selbst generiert hat, könnten grundsätzlich weiter funktionieren, vorausgesetzt der Account wird nicht komplett gelöscht sondern nur geleert. Dennoch sollten diese aber in diesem Moment eingefroren werden, indem diese etwa in fixe HTML Seiten umgewandelt werden und nicht bei jedem Aufruf neu generiert werden.

## **Über den Tellerrand**

Auch wenn das in dieser Arbeit vorgestellte TUREp System einen starken TU Wien und universitären Bezug hat, kann es dennoch auch in anderen Kontexten eingesetzt werden. Was auf jeden Fall erhalten bleiben sollte ist die Verbindung zu einer echten Person, da das System keine Mechanismen vorgesehen hat um mit Betrugsversuchen mittels mehrerer Accounts zu verhindern oder zu mitigieren.

## **Conclusio & Ausblick**

In dieser Arbeit habe ich mich mit der Fragestellung befasst, wie man ein Reputationsystem im Universitätskontext einsetzen könnte. Hierzu habe ich zuerst Beispiele für Reputationsysteme, sowie Methoden für deren Gestaltung recherchiert und in dieser Arbeit aufgearbeitet. Zusätzlich habe ich Interviews und Workshops mit Stakeholdern wie Professoren und Studenten durchgeführt. Basierend auf den Ergebnissen dieser Recherche, habe ich zwei Plattformen gestaltet. Eine simple Webseite zum Verleihen von Badges, sowie ein konzeptionelles, umfassendes Ökosystem um aus mehreren Quellen die Reputation eines Studenten zentral sammeln und präsentieren zu können.

Während den Recherchen hat sich relativ bald gezeigt, dass es ein gewisses etabliertes Set an Methoden gibt, mit Reputation umzugehen und die Kunst bei der Gestaltung eines Reputationsystems darin liegt, das richtige Rezept für diese Zutaten zu finden. Wie dieses Rezept aussehen muss, kann man meist nur durch mehrere Iterationen annähern. Es gibt aber Richtlinien, an denen man sich anhalten kann, um zumindest den Großteil bereits beim ersten Versuch richtig hinzubekommen. Diese Richtlinien orientieren sich aber daran, welche Art von Community man schaffen oder mit Reputation bereichern möchte. Da es aber von LVA zu LVA unterschiedlich sein kann, welche Form von Kooperation oder Konkurrenz man zwischen den Studenten erreichen möchte, macht es mehr Sinn einfache Komponenten zur Verfügung zu stellen und deren Verwendung den Lehrenden zu überlassen, statt ein fertiges Gesamtsystem zu forcieren. Die Lehrenden müssen aber auch über die korrekte Verwendung dieser Komponenten informiert werden. Dieser Aspekt könnte in zukünftigen Arbeiten beforscht werden, in dem etwa mit Versuchsdurchgängen im echten Lehrbetrieb Best Practices entwickelt werden.

Aus den Interviews und Workshops ergab sich, dass ein Reputationsystem oder ein Badgesystem möglichst keinen Zusatzaufwand bedeuten sollen. Dies konnte mit dem Badge Prototypen leider noch nicht vollständig erreicht werden. Dies liegt aber zum Teil auch daran, dass es noch

ungewohnt war Badges einzusetzen und es, wie bereits erwähnt, kein Katalog an Best Practices oder Beispielen existiert.

Von Studentenseite wurde vor allem mehr Transparenz eingefordert. Auch die Möglichkeit für Studenten selbst Badges oder Bewertungen abzugeben dürfte positiv dazu beitragen, dass sich das Interesse an so einem System erhöht. Beim aktuellen Badge Prototypen war das Interesse verständlicher Weise noch nicht allzu hoch, da Studenten nur mehr oder weniger passive Empfänger der Badges waren.

Ein Thema für weitere Arbeiten wäre, wie das vorgestellte TURep System explizit in bereits existierenden TU Systemen integriert werden kann. In dieser Arbeit bin ich bewusst nur sehr oberflächlich darauf eingegangen, da ich mich primär mit der allgemeinen Fragestellung auseinandersetzen wollte und weniger mit den Implementationsdetails einzelner Plattformen.

Abschließend sei erwähnt, dass diese Arbeit über einen längeren Zeitraum entstanden ist, da es durch verschiedene Gründe vorwiegend beruflicher Natur immer wieder zu etwas längeren Pausen kam. Dadurch war es aber auch möglich, bei manchen Plattformen zu beobachten, wie Reputationsysteme bei selbst sehr großen Plattformen wie Facebook oder Amazon immer noch einem steten Wandel unterliegen und laufend verbessert und angepasst werden. Dasselbe würde auch auf ein so umfassendes System wie TU Rep zutreffen. Wirklich aussagekräftige Erkenntnisse über die Motivation und Akzeptanz eines Reputationsystems im Universitätskontext werden sich auch erst dann gewinnen lassen, wenn ein entsprechendes System implementiert wurde und im laufenden Lehrbetrieb eingesetzt wird. Dennoch ist das Potential da, dass so ein System zur Verbesserung der Lehre beiträgt und eventuell auch zu einer Alternative zum klassischen Notensystem werden könnte.

## Referenzen

[Laat, 2005] P. B. de Laat: Trusting virtual trust, *Ethics and Information Technology* (2005) 7:167–180, Springer, 2006

[Kwan & Ramachandran, 2009] M.Kwan & D. Ramachandran: Trust and Online Reputation Systems, Chapter 11 of J. Golbeck (ed.), *Computing with Social Trust*, Human-Computer Interaction Series, Springer-Verlag London Limited, 2009

[Xiong & Liu, 2004] L. Xiong & L. Liu: PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities, *IEEE Transactions On Knowledge and Data Engineering*, Vol. 16, No. 7, 2004

[Farmer & Glass] R. Farmer & B. Glass: *Building Web Reputation Systems*, O'Reilly, Yahoo! Press, 2010

[Dellarocas, 2010] C. Dellarocas: *Designing Reputation Systems for the Social Web*, Boston University School of Management Research Paper Series, 2010

[Deterding et al, 2011] S. Deterding et al.: *Gamification: Toward a Definition*, CHI 2011

[Koch & Ott, 2012] M.Koch & F. Ott: *Gamification – Steigerung der Nutzungsmotivation durch Spielkonzepte*, Universität der Bundeswehr München, <http://www.soziotech.org/gamification-steigerung-der-nutzungsmotivation-durch-spielkonzepte>, 2012, accessed: 10.2015

[David & Pinch, 2005] S. David & T. Pinch: *Six Degrees of Reputation: The use and abuse of online review and recommendation systems.*, S&TS Working Paper, Cornell University, 2005

[alexa.com] <http://www.alexa.com/topsites/category/Top/Shopping/Auctions>, accessed: 10.2015

[theguardian, 2008] Sellers negative on eBay feedback change, <http://www.theguardian.com/technology/2008/feb/21/eBay.consumeraffairs>, accessed: 10.2015

[stackoverflow reputation] StackOverflow, What is reputation? How do I earn (and lose) it?, <http://stackoverflow.com/help/whats-reputation>, accessed: 10.2015

[stackoverflow bounty] StackOverflow, What is a bounty? How can I start one?, <http://stackoverflow.com/help/bounty>, accessed: 10.2015

[reddit voting] Reddit, voting, <https://www.reddit.com/wiki/voting>, accessed: 10.2015

[youtube.blogspot, 2009] Shiva Rajaraman, Five Stars Dominate Ratings, Official YouTube Blog, <http://youtube-global.blogspot.co.at/2009/09/five-stars-dominate-ratings.html>, accessed: 10.2015

[Yahoo Reputation] Yahoo: Reputation, Yahoo Design Pattern Library, <https://developer.yahoo.com/ypatterns/social/people/reputation/>, accessed: 10.2015

[Thom et al, 2011] J. Thom et al.: Removing Gamification from an Enterprise SNS, CSCW 2012, ACM, 2011

[Abramovich et al.] S. Abramovich, C. Schunn & R. Higashi: Are badges useful in education?: it depends upon the type of badge and expertise of learner, Association for Educational Communications and Technology, Springer, 2013

[Fitz-Walter et al, 2011] Z. Fitz-Walter et al.: Orientation Passport: Using gamification to engage university students, OZCHI ,11, 2011

[philbit] P. Rogers, SVG Patterns Gallery, <http://philbit.com/svgpatterns/>, accessed: 10.2015

[iconfinder.com] Iconfinder, <https://www.iconfinder.com/>, accessed: 10.2015

[jquery.autocomplete] T. Kirda, Ajax Autocomplete for jQuery, <https://github.com/devbridge/jQuery-Autocomplete>, accessed: 10.2015

[Haaranen et al] Lassi Haaranen et al: How (not) to Introduce Badges to Online Exercises, Department of Computer Science and Engineering Aalto University, School of Science Espoo, Finland, 2015

[openbadges.org] Mozilla: Open Badges, <http://openbadges.org>, accessed: 10.2015

[WAI] World Wide Web Consortium: Web Accessibility Initiative (WAI), <http://www.w3.org/WAI/>, accessed: 10.2015

[Yeung et al, 2009] C. A. Yeung et al.: Decentralization: The Future of Online Social Networking, W3C Workshop on the Future of Social Networking, 2009

[VoWi] VorlesungsWiki, Fachschaft f. Informatik, TU Vienna, <http://vowi.fsinf.at>, accessed: 10.2015

[couchDB] Apache CouchDB, Apache Software Foundation, <http://couchdb.apache.org/>, accessed: 10.2015

[mongoDB] MongoDB, MongoDB Inc., <https://www.mongodb.org/>, accessed: 10.2015

[diaspora] diaspora\*, <https://diasporafoundation.org/>, accessed: 10.2015

[newgrounds.com] Newgrounds, <http://www.newgrounds.com>, accessed: 10.2015

[Dean & Ghemawat, 2004] J. Dean & S. Ghemawat, Google Inc.: MapReduce: Simplified Data Processing on Large Clusters, OSDI, 2004

[Müller et al, 2008] W. Müller et al.: Sybil Proof Anonymous Reputation Management, SecureComm 2008, ACM, 2008

[Moodle] Moodle, <https://moodle.org/>, accessed: 07.2016

[classbadges.com] Classbadges, <http://classbadges.com>,  
accessed: 07.2016