

Praktikum

Datenbanken und verteilte Systeme



Jasper Elbeshausen

- elbeshausen@hochgestalten.de

Philipp Danne

- philipp.danne@gmail.com

Manuel Grill

- m.grill1988@gmail.com

Rene Ogniwiek

- reneogniwiek@gmx.net

Zeitraum des Praktikums:

31.08.2015 – 18.09.2015

Abgabedatum:

18.09.2015

Inhaltsverzeichnis

Unser Grundkonzept	3
Praktikumswoche 1	4
Praktikumswoche 2	6
Praktikumswoche 3	9
Datenbankschema	11
Vorgehensweise & Probleme während des Praktikums	11

Unser Grundkonzept

Die Gruppe 7 hat sich bei dem Praktikum „Datenbanken und Verteilte Systeme“ Für die zweite Option entschieden, um mit dem Bagend eine neue unkomplizierte Möglichkeit der Website-Entwicklung mit integrierter, vereinfachter Datenbankanbindung kennenzulernen.

Dabei wurde sich bewusst an ein Grundprojekt mit geringer Komplexität in Form eines Frage-Antwort-Forums nach Art der erfolgreichen StackOverFlow-Seite orientiert, damit die Konzeptionsphase mit der Aufwandseinschätzung transparent und übersichtlich bleibt.

Der Grundgedanke war, dass ein simples Konzept genug Platz für eine komplexe Realisierung bietet - sofern die behandelten Thematiken relativ eingängig bleiben, ist der Aufwand den Funktionsumfang und somit die Komplexität des Projektes zu erweitern leichter zu bewerten.

Die Grundfunktionen eines solchen Forums nach den Vorstellungen der Gruppe 7:

- Stellen von Fragen
- Beantworten von Fragen

Zusätzliche Funktionen wurden in Prioritäten eingeteilt, um weiterhin passierbare Meilensteine positionieren zu können:

- Prio1: Grundfunktionalität herstellen
- Prio2: Implementierung einer Registrierungs- und Loginmöglichkeit
- Prio3: Konzeption und Erstellung einer User Profile Page zur Veränderung gestellter Fragen und gegebener Antworten
- Prio4: Möglichkeit der Bewertung von Fragen und Antworten mithilfe eines Karma-Systems in Anlehnung an das Vorbild StackOverFlow
- Prio5: Programmierung einer Filter/Suchfunktionslogik

Der erste Meilenstein, die Grundfunktionalität des Forums zu gewährleisten sollte bereits in der ersten Woche realisiert werden. Von da an könnte der Funktionsumfang schrittweise erweitert werden.

In der zweiten Woche lag dabei das Hauptaugenmerk auf die Realisierung der Registrierfunktion und der Erstellung einer Profilseite.

In der letzten Woche sollte dann das Grundkonzept einer Bewertung stehen. Nach Möglichkeit sollte die Seite am Ende über eine Such- und Filterfunktion verfügen.

Inwiefern die gesetzten Prioritäten in den einzelnen Wochen realisiert werden konnten, zeigt das Entwicklertagebuch auf den nachfolgenden Seiten.

Entwicklertagebuch

Praktikumswoche 1

Nach Erstellung der Gruppe begann unverzüglich die Konzeptionsphase des Projekts. Bei der Anforderungsaufnahme wurde der Funktionsumfang eines Fragen- und Antwortforums diskutiert, mit dem Ergebnis sich grob an bereits realisierten Lösungen zu orientieren.

Als Vorlage dient dabei die Seite StackOverFlow. Dabei handelt sich um ein Internetforum, auf welchem registrierte Benutzer Fragen zu verschiedenen Aspekten des Themas Programmierung (oder auch andere Computer- oder Technikrelevanten Fragen) stellen können. Andere Nutzer, welche ebenfalls registriert sein müssen, können dann auf diese Fragen ihre Antworten geben. Sowohl die Fragen, als auch die gegebenen Antworten können von Usern bewertet werden, wodurch erreicht wird, dass gute Antworten auf die Fragen weiter oben angezeigt werden und auf der Startseite vorrangig "gute" Fragen auftauchen und Unsinnige schnell weit unten verschwinden. Zusätzlich hat der Fragesteller die Möglichkeit, eine gegebene Antwort als Beste Lösung für sein Problem zu kennzeichnen. Dadurch bekommen die Nutzer Ansehenspunkte. Jeder registrierte User hat zudem ein eigenes Profil mit eigenen Informationen. StackOverFlow hat über Jahre mithilfe ihres immer größer werdenden Nutzerstamms ständig weitere Funktionen hinzugefügt (z.B. eine Jobbörse), in unserem Projekt ging es aber vor allem um die Kernfunktionalität der Seite.

Die Entwicklungsarbeit begann nach einer kurzen Orientierungsphase der Handhabung des bereitgestellten Baqend Systems und der bereitgestellten Technologien wie:

- HTML
- CSS
- JS
- JQuery
- Grunt
- Git

Nach erfolgreichen ersten Schritten im Baqend-System und kurzer Verwirrung ob der endgültigen Arbeitsumgebung (Wo liegt unsere Datenbank, Wo liegt der Entwicklungsserver, Wie kommunizieren diese), die vom Baqend-Team schnell behoben werden konnten, wurden lokale Arbeitsumgebungen eingerichtet, wobei Wissensstandards in der Gruppe zu Versionskontrolle, Editor-Handhabung und auch HTML- bzw. CSS-Konzepten aufgebaut wurden. Diese erstmal gefestigten Wissensstandards halfen uns in späteren Projektphasen zum einen eine saubere Kommunikation zu gewährleisten und zum anderen die Zuständigkeiten innerhalb der Gruppe aufzuteilen.

Nach der Implementierung von Prio1 war jeder Besucher der Seite in der Lage Fragen zu stellen und darauf zu antworten. (siehe Abb. 1 & Abb. 2) Die verfassten

Texte wurden im Hintergrund durch die Datenbank mit Zeitstempel und ID versehen, um innerhalb späterer Prioritäten noch weitere Aktionen mithilfe dieser Werte vornehmen zu können. Um die Validierung der Änderungen im Code leichter zu gestalten, wurden die ersten Dummyfragen und –antworten innerhalb des Backend-Frontends generiert.

The screenshot shows a web form titled "FUBA - the new StackOverflow!!!". It contains two input fields: "Type Your Username here!" and "Your Question Title". Below these is a large text area for the question. A "Submit Your Question!" button is located to the left of the question area. Below the form, two example questions are displayed in a light blue box. The first example is from "WildeHexe43" posted at "Thu Sep 03 2015 15:51:31 GMT+0200" with the title "Oralsexverkehr" and the question "Kann ich von Oralsex schwanger werden?". The second example is from "FairLady13" posted at the same time with the title "Oral- und Analverkehr" and the question "Eine Freundin hat mir abgeraten, mit meinem Schatz Oral- und Analsex gleichzeitig auszuprobieren. Was meint sie damit?".

(Abb. 1 – Questions)

The screenshot shows the answer form for the question "Wie gelange ich mit meinem Computer in den abgesicherten Modus auf Windows 8.1?". It features a "Type Your Username here!" input field and a larger text area for the answer. Below the text area are two buttons: "Daten absenden" and "Submit Your Answer!". The example question and answer from "SchlauerFuchs67" are visible above the form.

(Abb. 2 – Answers)

Der nächste logische Schritt für die Gruppe war es sich als registrierter Nutzer ein- und ausloggen zu können. Die Implementation einer Nutzerdatenbank hatte zur Folge, dass nun mehr Metainformationen zu den Fragen und Antworten zur Verfügung stehen.

Von jetzt an konnte neben Zeitstempel und Titel auch ein Nutzer zugeordnet werden. Die Entwicklung im frühesten Stadium war vor allem durch ihre Modularisierung geprägt. So wurde die Nutzerdatenbank händisch gefüllt bevor die Funktionen des Ein- und Ausloggens bereitgestellt wurden, um die komplexere Funktion der Registrierung danach zu realisieren. Wir arbeiteten in der ersten Woche quasi nach einer **"Easy-Things-First"-Devise**.

Durch diese schrittweise und sehr kleinteilige Implementation der Grundkomponenten wurde das Debugging deutlich vereinfacht und so fiel es nicht schwer, eine saubere Login/Logout-Funktion zu schreiben und daraufhin mit der Registrierung fortzufahren - Für diese war die Woche allerdings zu kurz und so wurde dieser Task als kurzfristiges Ziel für den nächsten Wochenstart festgelegt.

Praktikumswoche 2

Nachdem in der ersten Woche die Grundfunktionalität der Seite stand, ging es in der zweiten Woche vor allem um die Realisierung der Registrierungsfunktion und um das Erstellen einer Profilseite.

Während der Implementation der Registrierungsmöglichkeit traten erste schwerere Probleme auf: Das Konzept der „Promises“ war zuvor niemanden von uns wirklich bekannt. Nach einigen Versuchen und zwischenzeitlicher Anleitung durch das Baqend-Team konnte das Konzept der asynchronen Code-Ausführung verstanden und im Endeffekt extrem häufig angewendet werden. So nutzten wir die **".then(function(result){code here})"**-Konstruktion vermutlich innerhalb jeder Priorität.

Die Registrierungsfunktion findet auf einer weiteren Seite statt. Auf der Startseite wird der Besucher nun unter dem Login-Bereich mit dem Button **"Register now!"** auf die neue Registrierungsseite weitergeleitet. (Siehe Abb. 3)

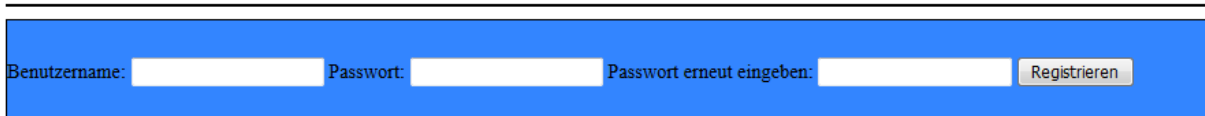
FUBA - the new StackOverflow!!!

The screenshot shows the FUBA website interface. At the top, there is a blue header with the text "Hey Stupid, log in to use our site or else...". Below this, there are input fields for "Benutzername" and "Passwort", followed by an "Einloggen" button. A "Register now!" button is also present. Below the header, there is a list of posts. The first post is by "BigManu" posted at "Wed Sep 16 2015 13:45:16 GMT+0200" with a title "Vergammeltes H?hnchen" and a description "Warum rebelliert Renes Magen so derbe!?". The second post is by "WildeHexe43" posted at "Thu Sep 03 2015 15:51:31 GMT+0200" with a title "Oralverkehr" and a description "Kann ich von Oralsex schwanger werden?".

(Abb. 3 – Register now! Button)

Der Besucher kann auf der neuen Seite sowohl einen beliebigen Benutzernamen anlegen als auch ein eigenes Passwort vergeben und somit die Registrierung erfolgreich abschließen. (Siehe Abb. 4)

FUBA - the new StackOverflow!!!

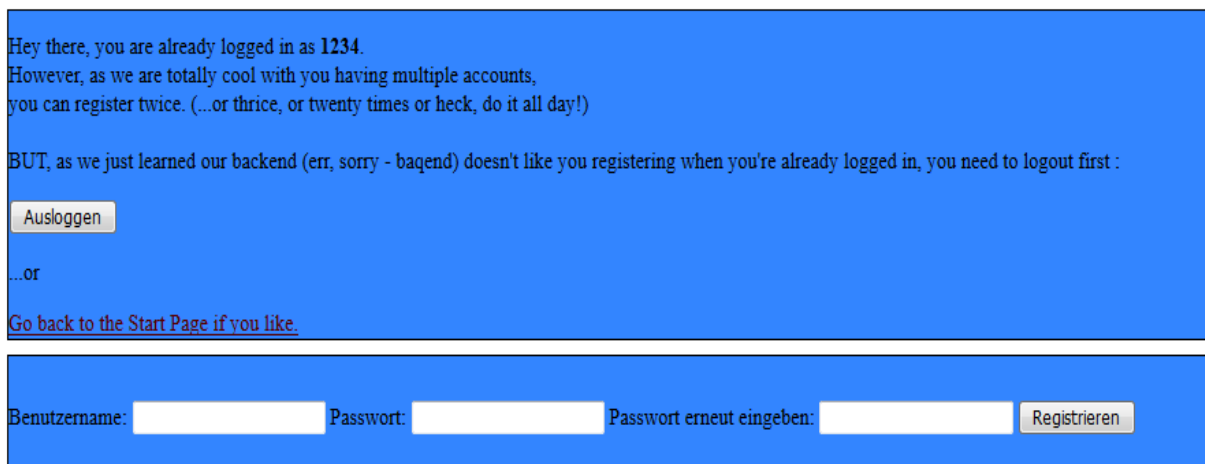


(Abb. 4 – Registerseite)

Hier stießen wir zum ersten Mal auf Fragen, die Funktionalität betreffend, die im Vorfeld noch nicht als solche realisiert wurden. Fraglich zum Beispiel war, ob ein schon registrierter und eingeloggter Nutzer in der Lage sein soll sich erneut zu registrieren - endgültige Antwort: Nein (eine ausführlichere befindet sich auf der /register.html-Seite unserer Webanwendung).

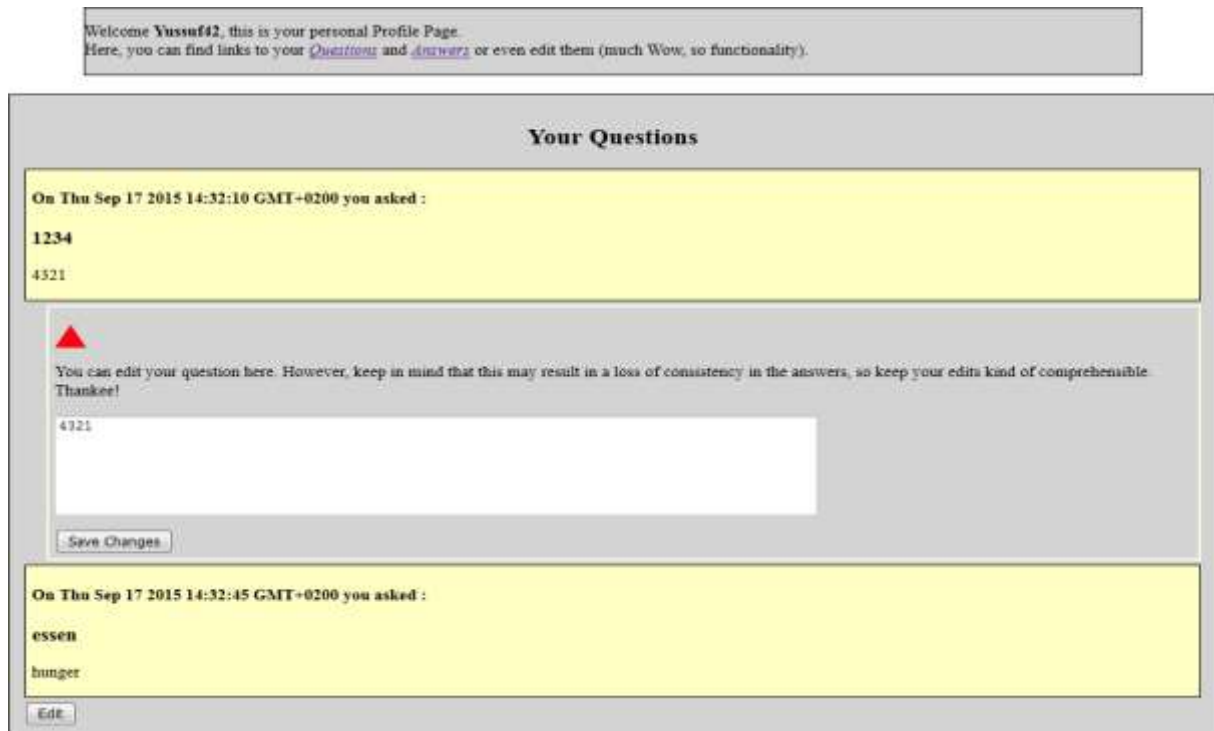
Ab jetzt wird der Nutzer automatisch mit einer persönlichen Begrüßungsmassage im oberen Bereich der Website empfangen. Es besteht die Möglichkeit sich sofort wieder auszuloggen oder zur Startseite zurückzukehren. Der eingeloggte Benutzer kann nun Fragen stellen und auf andere Fragen antworten. (Abb. 5)

FUBA - the new StackOverflow!!!



(Abb. 5 – erfolgreiche Registrierung)

Nachdem die Registrierungsfunktion vollständig implementiert wurde sollte es dem Nutzer im nächsten Schritt möglich gemacht werden, eine personalisierte Profil-Seite aufzurufen. Auf der Profilseite werden alle Fragen und Antworten des angemeldeten Users aufgelistet, welche durch einen implementierten Edit-Button aufgeklappt werden und dann nach Belieben bearbeitet werden können. Entscheidet sich der User dafür doch nichts zu ändern, kann durch das Anklicken des roten Buttons die Frage oder Antwort wieder eingeklappt werden. Wurde etwas geändert, schließt man den Vorgang durch Anklicken des Buttons "Save Changes" ab. (Abb. 6)



(Abb. 6 – Profilseite mit Edit-Funktion)

Durch die Fertigstellung der Priorität 2 & 3 wurde die Funktionalität der Seite erfolgreich ausgebaut. Dabei wurde sich parallel an die Bearbeitung des Designs gesetzt, sodass sich nicht nur die Funktionalität sondern auch das Design auf einem aktuellen Stand befand. (siehe Abb. 7)



(Abb. 7 – Übersicht der Frage Seite und Antworten Seite zum Ende der zweiten Woche)

Zum Ende der zweiten Woche wurde die Funktionalität eingeführt, die Fragen auf der Startseite bewerten zu können. Allerdings besteht aktuell noch das Problem, dass sowohl angemeldete als auch nicht angemeldete Nutzer beliebig oft eine Bewertung für eine Frage abgeben können. Des Weiteren können gegebene Antworten noch nicht bewertet werden. Die vierte Priorität wird also bereits in der zweiten Woche angeschnitten um letztendlich in der letzten Praktikumswoche finalisiert zu werden.

Praktikumswoche 3

Die Editierfunktion wurde nach der Fortschrittspräsentation am Montag erweitert. Sollte eine Frage oder Antwort am Ende nicht editiert werden ist nun ein Button zur Schließung der Editiermaske implementiert worden.

Des Weiteren wurde im Falle einer Bearbeitung eines Textes eine „edited at ...“ flag implementiert, da es vorher nicht möglich war eine geänderte Frage oder Antwort als solche zu identifizieren.

Um sich der Vorlage anzunähern, wurde zusätzlich eine neue „Section“ mit einer Auflistung der Fünf neuesten Fragen erstellt. Dabei stellte sich das vorher gewählte Datumsformat als problematisch heraus.

Das Format: **Tag, Monat tt yyyy hh:mm:ss GMT+0200**

Eine einfache „descending-Anweisung“ mit diesem Format ist nicht ausführbar, da dies kein ausschließlich numerischer wert ist.

Mithilfe einer neuen Datenbank, genannt Counter, konnte jedoch ein Workaround gebaut werden, der gleichzeitig die Problematik der Löschfunktion löst. Letztere war als spätere Priorität geplant, letztendlich jedoch nicht mehr realisiert. Die Counter-Datenbank beinhaltet nun exakt 2 Einträge, die von uns innerhalb des Javascript "hardcoded" via ihre ID angesprochen werden, was aufgrund ihrer Singularität kein Problem darstellt. Einer der Einträge ist hierbei den Q_IDs, also den eindeutigen Identifikationsnummern einer Frage, und der andere den A_IDs, den eindeutigen Identifikationsnummern einer Antwort, zugeordnet. Die Counter werden bei Erstellung neuer Fragen / Antworten inkrementiert um für die nächste Frage / Antwort vorbereitet zu sein.

Dieser Logik nach hat die neueste Frage auch immer die größte Q_ID, wodurch eine Anfrage der neusten fünf Fragen simpel über eine descending (Q_ID)- Anweisung realisiert werden konnte. (Siehe Abb. 8)

FUBA - the new StackOverflow!!!

Hey Guest, please login or register.

Benutzername Passwort

New Questions

▲
3
▼

Yussuf42 posted at Thu Sep 17 2015 14:32:45 GMT+0200

essen

hunger

▲
0
▼

Yussuf42 posted at Thu Sep 17 2015 14:32:10 GMT+0200

1234

4321

(Abb. 8 – New Questions werden angezeigt)

Als weitere letzte Funktionalität wurde durch eine Fußzeile auf der Startseite auf die Impressumseite verlinkt, auf der das Team und ein kleine Danksagung abgebildet wird.

FUBA - the new StackOverflow!!!

Impressum

Team
Philipp Danne
Manuel Grill
René Ognirek
Jasper Eibeshausen

Credits
go to Uni HH and the Team of tutors as well as the very supportive bagend-team of course, however we also would like to thank Astra and the Mensa-team Stellingen for their Support.

Am letzten Tag vor der Abschlusspräsentation wurde sich dafür entschieden keinerlei neue Funktionen mehr zu implementieren. Vielmehr wurde sich darauf konzentriert die letzten Bugs auf der Seite zu beheben und die Funktionalität in allen Bereichen erneut zu überprüfen. Des Weiteren wurde der Quelltext ausführlich kommentiert, die Readme aktualisiert und eine Endversion des Praktikumsberichts realisiert.

Bei letzten Tests fiel noch auf, dass nicht registrierte Nutzer nach wie vor Fragen und Antworten bewerten konnten, sodass hier noch ein kleiner Fix nötig wurde. (siehe Abb. 9)

FUBA - the new StackOverflow!!!

REMEMBER, you need to be logged in to be able to vote for questions or answers.

QUESTION ASKED:

▲

40

▼

BigManu asked at Wed Sep 16 2015 13:45:16 GMT+0200
[Vergammeltes Hühnchen](#)
Warum rebelliert Renes Magen so derbe!?

GIVEN ANSWERS:

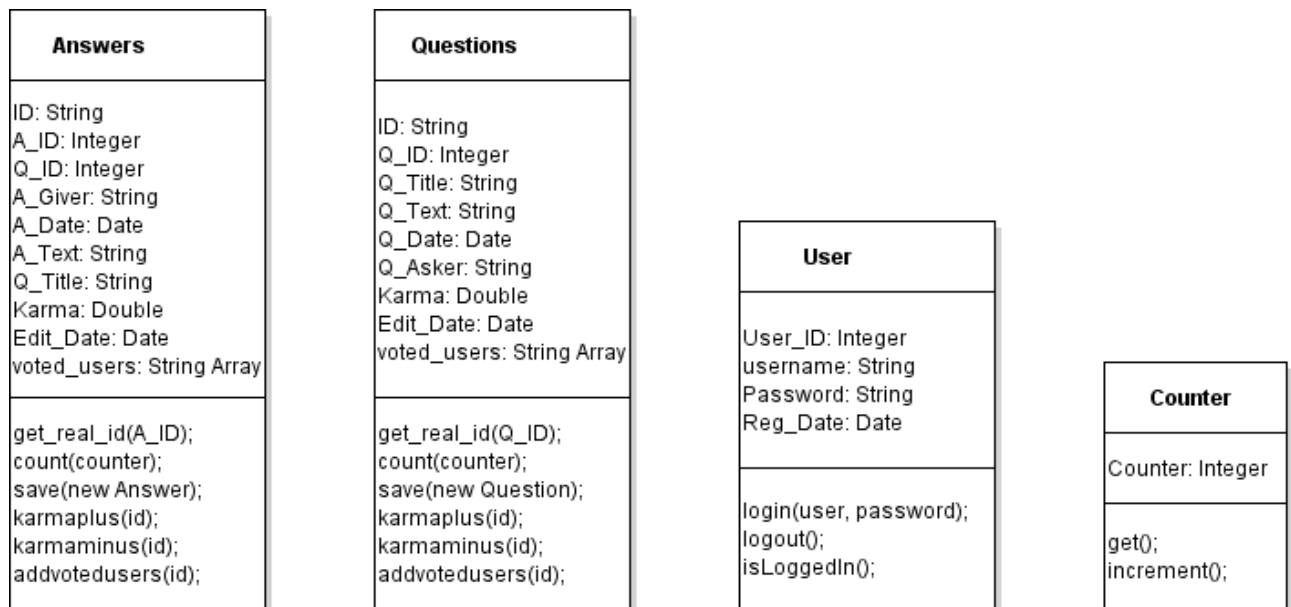
▲

0

▼

Yusuf42 answered at Thu Sep 17 2015 14:32:52 GMT+0200 :
123

Datenbankschema



In Folgendem Diagramm werden die von der Seite verwendeten Datenbanken Tables angegeben.

Dabei sind „Count“ und „Addvoteduser“ besonders hervorzuheben, da sie Prozessunterstützend funktionieren.

Der Table "Counter", hält exakt zwei Dateneinträge mit einem Integer als Zähler. Jeweils ein Eintrag ist für die ID Generierung für Fragen, der andere für die ID Generierung der Antworten zuständig. Wird ein neuer Eintrag in Questions oder Answers erstellt, zählt Count um genau einen höher.

Mit dieser Logik wird in jedem Fall eine eindeutige Zuordnung gewährleistet, auch wenn Fragen oder Antworten gelöscht werden, da dieser beim Löschen eines Datensatzes nicht herunterzählt.

Bei „addvotedusers“, wurde sichergestellt, dass ein Nutzer nur einmal eine Frage oder Antwort bewerten kann.

Das Array „voted_users“ beinhaltet alle „Usernamen“, welche die zugehörige Frage bereits bewertet haben.

So wurde sichergestellt, dass das Ranking der Fragen und Antworten, nach denen der Nützlichste Beitrag oben ist, nicht manipuliert werden kann in dem eine Frage mehrfach gevoted wird.

Vorgehensweise & Probleme während des Praktikums

Grundlegend arbeiteten wir die ganze Zeit mit Sublime als Texteditor und der Git Bash für alle Git-Tasks, wobei zumeist zwei von uns in CSS herumfuhrwerkten und zwei funktional weiterbauten.

Grundsätzliche Probleme traten anfangs bei der Erstellung und "Füllung" von „Handlebars-Templates“ auf, da mit diesen noch niemand gearbeitet hatte und der generelle Aufbau „render-Funktion“, „template-Variable“ und „onReady-Funktion“, die alle miteinander arbeiten und gegenseitig aufeinander abgestimmt sind, nicht intuitiv reproduzierbar waren. Nach Erstellung mehrerer „Subpages“ entwickelte sich dieses jedoch mehr und mehr zum Automatismus.

Weitere Probleme traten dadurch auf, dass die Pool-Rechner doch sehr restriktiv aufgebaut sind, was im Endeffekt dazu führte, dass wir täglich Git neu installierten (anstatt auf eine portable Version umzusteigen, unser Fehler).

Als sich unser Projektordner langsam mit mehr und mehr JS-Dateien füllte, bekamen wir irgendwann Probleme mit der Datenbank-Verbindung. Wir hatten in allen Dateien den DB-Connect, womit Baqend scheinbar überfordert war. Wir umgingen das Problem im Endeffekt dadurch eine eigene connect.js zu erstellen, die sich einzig um den DB-Connect kümmert und fortan auf allen Seiten als erste eigene js geladen wurde.

Weitere Probleme traten im Rahmen unseres Voting-Systems auf, von dem wir relativ klare Vorstellungen hatten:

Generell sollte jede Frage und Antwort einen Zahlenwert als Karma innehaben. Dieses Karma ist nun bei Fragenerstellung als 0 initialisiert. Neben jeder Frage und Antwort sollten nun zwei Pfeile für „Up- und Downvote“, also karma-inkrement und karma-dekrement sein. Durch dieses User-Voting hätten wir dann mithilfe eines korrekten Query an die Datenbank kein Problem die besten Antworten zu einer Frage oben zu zeigen.

Um diese Karma-Werte jedoch aussagekräftig zu machen wurde es nötig jeden User nur einmal zu jeder Frage und jeder Antwort voten zu lassen, was im Endeffekt in dem komplexesten Stück Code den wir produzierten resultierte (s. App.js | 152 ff.).

In der finalen Version haben wir es geschafft jeden User maximal einmal voten zu lassen und ihm zu signalisieren falls er es ein zweites Mal versucht. Unangemeldete User dürfen nun nicht mehr voten.

Bei der Anforderungsaufnahme gelang es der Gruppe 7 nicht immer die Prioritäten in sinnvolle Teilaufgaben zu unterteilen.

So ist es vorgekommen, dass bei der Anforderungserfüllung einzelner Prioritäten weitere wichtige Unteraufgaben auftraten um den diskutierten Prozess modellieren zu können, bspw. war es für die Sektion "New Questions" nötig, die Fragen irgendwie nach Erscheinungsdatum zu sortieren, was im Endeffekt durch den Q_ID-Workaround realisiert wurde (beschrieben in Woche 3).