

# **Ein kooperatives mathematisches Spiel mit kollaborativer Spielmechanik**

## **Projektarbeit**

Web-Development II

In Kooperation mit Penn State University

Vorgelegt von Team 3:

Andreas Glaser

Andreas Khouri

Christa Strauß

Dzianis Brysiuk

Noah Lehmann

Tina Amann

Studiengang:

Informatik

4. Semester

**Dozent:**

**Prof. Dr. Jürgen Heym**

## Inhaltsverzeichnis

1 Einleitung.....	7
2 Beschreibung des Projektes.....	7
Einleitung in die Thematik.....	7
Erklärung des Projektumfangs.....	8
3 Projektrandbedingungen.....	10
Rollenverteilung der Beteiligten.....	10
Vorgehensmodell.....	12
Vorgegebene Hard-/ Software.....	12
Entwicklertools.....	13
4 Erkundung der Frameworks.....	15
HumHub.....	15
Entscheidungsmatrix.....	15
Erklärung der Vorgehensweise.....	15
Bewertungskriterien.....	16
Bewertung.....	17
Entscheidung.....	17
Markdown.....	18
Syntax.....	18
Nutzungsregeln.....	19
AngularJS.....	20
Analyse der Funktionen von AngularJS.....	20
1. Services.....	20
2. Filter.....	22
Bootstrap.....	23
Yii.....	23
Yii2 Basic Installation.....	23
Yii2 Advanced Installation.....	24
5 Entwicklung des Prototypen.....	25
Regelsätze.....	25
Regelsatz 1.....	25
Punktevergabe.....	25
Multiple-Choice.....	26
Frage-Antwort.....	27
Kollaboration.....	29
Punkteverhältnis.....	30
Amateur (Level 1).....	30
Fortgeschritten (Level 2).....	31

<i>Profi (Level 3).....</i>	31
Regelsatz 2.....	33
Warum die KI hilft bzw. Hilferufe simuliert.....	33
Hilferufe.....	33
Wo die KI hilft.....	34
Wann die KI hilft.....	34
Wie die KI hilft beim Kauf von Hinweisen.....	34
Hilfestellung bei einem Rundungsfehler.....	36
Hilfestellung bei einer richtigen Antwort.....	36
Hilfestellung bei keiner richtigen Antwort.....	36
Wie die KI beim Antworten auf Hilferufe hilft.....	37
Wie die KI Hilferufe generiert.....	37
Weitere Beispiele für käufliche Hinweise.....	37
Regelsatz 3.....	40
Beschreibung des Problems.....	40
Was unterscheidet gute von schlechten Spielern?.....	40
Aufbau eines Fragezyklus.....	41
Berechnung von Richtwerten zur Bewertung von Spielern.....	42
Beispiel.....	44
Aufteilung der Website.....	45
User-Frontend.....	45
Admin-Frontend und Admin-Backend.....	47
Session.....	49
Screendesign.....	50
Aufgaben.....	50
Fortschrittsanzeige.....	51
Frage.....	52
Antworten.....	53
Einbindung der Corporate Identity des Kunden.....	54
User Stories.....	54
Registrierung Lehrer.....	55
Registrierung Schüler.....	55
US Login.....	56
US Schüler beantwortet Aufgabe (Multiple Choice).....	56
US Schüler beantwortet Aufgabe (Frage-Antwort-Aufgabe).....	57
US Schüler braucht Hilfe.....	58
US Schüler nutzt das Wiki.....	59
US Schüler hilft Mitspieler (Kollaboration).....	59
US Lehrer erstellt ein Klassenzimmer.....	59
US Lehrer möchte Statistiken sehen.....	60
US Lehrer verwaltet Schülerkonten.....	60
US Logout.....	61

Einbinden des Spiels.....	61
Auswahlmöglichkeiten.....	61
Wahl des Spielkonzepts.....	64
Serverseitige Funktionen.....	65
Nutzer-Verifizierung.....	65
Passwortsicherheit.....	66
Aufbau der Datenbank.....	68
Administrator-Tabelle.....	68
Lehrer-Tabelle.....	69
User-Tabelle.....	69
Session-Tabelle.....	71
Aufgaben-Tabelle.....	73
Datenbank Inhalte laden.....	73
Benutzer Anlegen.....	73
Login.....	74
Anlegen einer Session.....	74
Anmelden eines Users in einer Session.....	75
Übersicht über alle User.....	75
Bearbeiten von Aufgaben.....	75
Aufbau und Speicherung der Aufgaben.....	76
Aufgabentypen.....	76
Aufgabenarten.....	77
Generieren der Aufgaben.....	77
Speicherung der Aufgaben in der Datenbank.....	78
Abrufen der Aufgaben aus der Datenbank.....	79
Nachbearbeitung der Aufgaben.....	80
Wer führt die Überprüfung durch?.....	80
Wie geht der Tester vor?.....	80
Wie geht der Lehrer vor?.....	80
Wie geht das Programm vor?.....	81
Warum ist die Überprüfung nötig?.....	81
Datenschutz.....	81
Datenschutzregelungen.....	82
Einwilligungsgrenze.....	82
Datenschutz.....	82
Transparenz.....	82
Verfahrensverzeichnis.....	83
Einwilligung.....	83
Nachfrage.....	83
Datenvernichtung.....	84
User.....	84
Administrator.....	84

Datenschutzerklärung.....	85
Allgemeiner Hinweis und Pflichtinformationen.....	85
Mindestalter.....	85
Widerruf Ihrer Einwilligung zur Datenverarbeitung.....	85
Recht auf Beschwerde bei der zuständigen Aufsichtsbehörde.....	86
Recht auf Datenübertragbarkeit.....	86
Recht auf Auskunft, Berichtigung, Sperrung, Löschung.....	86
SSL- bzw. TLS-Verschlüsselung.....	86
Impressum.....	87
Betreiber / Kontakt.....	87
6 Backend Acceptance Tests.....	88
„Login“.....	88
„Language Change“.....	90
„Teacher Site“.....	90
„Header Links“.....	92
„View Teacher“.....	93
„Update Teacher“.....	94
„Create Teacher“.....	97
„Logout“.....	99
7 Verweise und Quellen.....	100
8 Anhang.....	102
Sprintplanungen.....	103
Erster Sprint.....	103
Aufgaben.....	103
Rückblick.....	104
Zweiter Sprint.....	106
Aufgaben.....	106
Rückblick.....	107
Dritter Sprint.....	108
Aufgaben.....	108
Rückblick.....	109
Vierter Sprint.....	110
Aufgaben.....	110
Rückblick.....	111
Fünfter Sprint.....	112
Aufgaben.....	112
Rückblick.....	113
Sechster Sprint.....	114
Aufgaben.....	114
Rückblick.....	115
Termine und zeitliche Abläufe.....	116
Meetings und Team-Treffen.....	116

Zusammenarbeitsregeln.....	119
Backup- / Restore-Vorgang.....	119
Erstellen eines Backups.....	119
Wiederherstellen eines Backups.....	120
Backup-Strategie.....	121
GitHub Nutzungsregeln.....	123
Repository.....	123
Git-Admin.....	123
Pull.....	123
Commit.....	123
Push.....	124
Konflikte.....	124
Branches.....	124
Mergen.....	125
9 Erklärung zur Selbstarbeit.....	126

## 1 Einleitung

Dieser Projektplan beschreibt die grundlegenden Eigenschaften dieses Projektes. Er geht auf die Aufgabe an sich ein, die Absprechen die mit dem Auftraggeber gemacht wurden und beschreibt im Abschluss unsere Herangehensweise an das Projekt. Genaue Daten zu Meilensteinen und Meetings werden nicht hier festgehalten, da das Team sich in Absprache mit dem Auftraggeber im Vorhinein auf ein agiles Vorgehen geeinigt hat.

## 2 Beschreibung des Projektes

In diesem Kapitel wird beschrieben, wie die Randbedingungen des Projektes aufgebaut sind. Der Ursprung und das Ziel des Projektes werden hier erläutert.

### Einleitung in die Thematik

Die Grundlegende Idee hinter diesem Projekt liegt in einem Problem, das aktuell die Hochschulen weltweit, aber vor Allem die in den Vereinigten Staaten von Amerika beschäftigt. Viele junge Menschen, die an weiterführende Schulen wie Universitäten gehen, haben große Mühe damit, sich in mathematischen Gebieten ihrer Studienfächer dem Tempo der Lehre anzupassen. Leider fängt dieser Rückstand dieser Lücken oft schon sehr früh an, entweder bei der Einführung der Mathematik an den Grundschulen oder an den weiterführenden Kursen in der High School oder ähnlichen Einrichtungen.

Um dieses Problem anzugehen, haben sich die Penn-State-University aus Pennsylvania, USA, und die Hochschule für angewandte Wissenschaften aus Hof,

Deutschland, zu einer Zusammenarbeit entschieden. Die führenden Professoren haben sich schnell dazu entschieden, dieses Problem mit der Entwicklung eines Spieles anzugehen. Die Idee ist es, junge Menschen spielerisch dazu zu bewegen, sich in ein Gebiet einzuarbeiten, welches ihnen oft nur wenig Spaß macht. Zusätzlich wird Wert darauf gelegt, dass Schüler sich untereinander helfen und einige Probleme auch zusammen angehen. Im weiteren wird weiter darauf eingegangen, wie die genaue Aufgabe dieses Projektes definiert ist.

## **Erklärung des Projektumfangs**

Wie oben beschrieben, beschäftigt sich dieses Projekt mit der Entwicklung eines Lernprogrammes, welches es Schülern ermöglicht, sich spielerisch und unter Zusammenarbeit in die Thematik der Mathematik einzuarbeiten. Das Projekt hat mit der Absprache einiger Studenten der Hochschule Hof bei den Kollegen aus der Penn-State-University angefangen. Dort wurden erste Ideen entwickelt, die beschreiben, wie man ein Spiel aufbauen kann, welches mathematischen Inhalt vermittelt. Die Studenten aus der Hochschule in Hof wurden darauf hin beauftragt, sich einer dieser Ideen zu widmen und einen Prototypen zu entwickeln, der es ermöglicht, das Projekt zu legitimieren und die Funktion der Idee des spielerisch-kollaborativen Lernens zu beweisen. Dieses Projekt beschäftigt sich also damit, eine erste Version des Mathe-Spiels zu implementieren. Hier wird jedoch nicht nur auf den spielerischen Inhalt geachtet, sondern der Fokus liegt durchaus zu großen Teilen auf dem kollaborativen Aspekt des Prototyps. Der Prototyp soll umfassende Ideen und Analysen zu dem Konzept des Mathe-Spiels enthalten, die Arbeiten der Studenten sollten umfassend dokumentiert werden, um späteren Teams die Möglichkeit zu geben, auf den Arbeiten des Prototyps aufzubauen. Des Weiteren soll am Ende des Projektes ein fertiger und lauffähiger Software Prototyp stehen, der die Kernfunktionen und Ideen des aktuellen

Projektteams widerspiegelt. Details zu den Ideen müssen nicht komplett implementiert werden, dazu ist der Projektzeitraum zu klein.

### 3 Projektrandbedingungen

Es wird das Team vorgestellt und auf die Mittel, die das Team nutzt, eingegangen. Darunter fallen die Rollenverteilung im Projekt, die Vorgehensweise und die genutzten Entwicklertools.

#### Rollenverteilung der Beteiligten

Das Projekt in diesem Umfang ist im Allgemeine auf 7 beteiligte begrenzt. Allen voran ist hier der Auftraggeber Prof. Dr. Jürgen Heym zu erwähnen, er ist verantwortlich für die Idee und Zusammenarbeit mit der Penn-State-University. Es obliegt ihm, den Auftrag für das Entwicklerteam zu definieren und im Laufe des Projektes erforderliche Änderungen im Umfang der Aufgaben durchzuführen.

Des weiteren wurde von Prof. Dr. Heym ein Teamleiter ernannt, der zuständig für den Ablauf des Projektes ist. Diese Rolle wurde Noah Lehmann zugeteilt, da er der Gruppe angehört hat, die bei der Absprache mit der Penn-State-University anwesend waren. Seine Aufgabe besteht also primär darin, dem Rest des Teams den Hintergrund des Projektes zu vermitteln und den Ablauf der Arbeit zu beobachten. Um dies zu ermöglichen fungiert er bei Fragen oder Unklarheiten als Vermittler zwischen dem Entwicklerteam und dem Auftraggeber. Zudem wurde ihm im ersten Teammeeting unter Absprache des Entwicklerteams die Aufgabe zugeteilt, die Dokumentation des Prototypen zu verfassen und primär zu verantworten.

Bei dem Teammeeting wurde Dzianis Brysiuk die aus der Vorgehensweise resultierende Aufgabe des Scrummasters übertragen. Seine Aufgabe besteht darin, dem Teamleiter bei der Koordinierung der Aufgaben zu helfen. Zudem ist es seine Aufgabe, nicht nur allgemeine Projektaufgaben zu übernehmen, sondern dem Rest des Teams primär bei ihren Aufgaben zu helfen. Sollten also irgendwo Unklarheiten

entstehen oder Probleme verursacht werden, so steht der Scrummaster dem Team unterstützend zur Verfügung.

Andreas Glaser wird Scrummaster und Teamleiter beratend zur Seite stehen, da er derjenige ist, der im Team die meiste Erfahrung mit dem Vorgehensmodell Scrum ist. Zusätzlich übernimmt er die Aufgabe eines Entwicklers.

Christa Strauß hat die Verantwortung für das Testen der Codeabschnitte übernommen. Zudem arbeitet sie auch als Entwicklerin.

Tina Amann hat die Aufgabe des Entwickelns von Skripten auf dem Server übertragen bekommen, auch sie wird nebenbei als Entwicklerin fungieren.

Andreas Khouri übernimmt das Sichern des Zustands des Web-Servers, auf dem die Applikation laufen wird. Er ist dafür zuständig, einen Backup Plan zu erstellen, außerdem ist es im Verlauf des Projektes seine Aufgabe, die Backups dann zu erstellen und die Speicherabbilder zu sichern. Im Falle eines Problems muss er dann auch die Abbilder wiederherstellen.

#### Zusammenfassende Tabelle der Rollenverteilung:

Name des Beteiligten	Aufgabenzuteilung
Prof. Dr. Jürgen Heym	Auftraggeber
Andreas Glaser	Entwickler, Beratung bei Fragen zu Scrum
Andreas Khouri	Verantwortlicher für Backups/ Recovery
Christa Strauß	Entwicklerin, Verantwortliche für Tests
Dzianis Brysiuk	Scrum Master
Noah Lehmann	Teamleiter, Verantwortlich für Dokumentation
Tina Amann	Entwicklerin, Verantwortlich für Serverseitige Skripts

## Vorgehensmodell

Nach Absprachen mit dem Auftraggeber hat sich das Projektteam für das agile Vorgehensmodell Scrum entschieden. Dieses ist aus kurzen Sprints aufgebaut, in denen vom Auftraggeber priorisierte Aufgaben abgearbeitet werden. Diese Aufgaben werden regelmäßig mit den Verantwortlichen evaluiert und abschließend neu priorisiert. Dies birgt den Vorteil, dass der Fokus in den Prototypen immer neu gelegt werden kann. Sollte sich im Laufe eines Sprints herausstellen, dass das Team seine Arbeitszeit auf Dinge fokussiert, die in der Hierarchie nicht ganz oben stehen, so kann dies schnell behoben werden. Die Schere des Verständnisses zwischen Auftraggeber und Team wird so immer gering gehalten. Ein weiterer Vorteil ist, dass das Team bei gescheiterten Ansätzen, was bei einer neuen Entwicklung oft vorkommt, diesen Rückstand schnell verarbeiten kann und die Umstellung auf einen neuen Ansatz schnell und ohne großen Planungsaufwand geschehen kann.

## Vorgegebene Hard-/ Software

Als Grundlage für deren Arbeit wurde dem Entwicklerteam von Auftraggeber Prof. Dr. Jürgen Heym ein Serverzugang auf eine virtuelle Maschine zugeteilt. Diese virtuelle Maschine läuft unter der Virtualisierungssoftware VMWare (von VMWare Inc.). Installiert ist ein Linux Server, der auf der Debian Distribution Ubuntu 16.04.4 basiert. Zugang zu diesem gibt es auf 2 Wege, per Secure Shell im Hochschulnetz einwählen oder über das selbe Netzwerk die Weboberfläche der VMWare zugreifen. Nachinstalliert wurde auf dem Server zusätzlich ein Apache Web-Server der Version 2.4.18 und ein MySQL Datenbank-Server der Version 5.7.20-0. Auf diesem hinterlegt ist eine HumHub Version, die auf dem Server nachinstalliert werden muss. Über das HumHub Modul soll im Projekt der Teil der Kollaboration durchgeführt werden, da dieses die wichtigsten Funktionen, wie Foren ähnliche Spaces, Umfragen und

Aufgaben in Form einer Webbasierten Seite schon zur Verfügung stellt. Die Rechtfertigung der Nutzung von HumHub gegenüber anderer Alternativen ist von den Nachforschungen, die von einem Team studentischer Hilfskräfte vor dem Projektstart durchgeführt wurden, bestätigt.

Zum Projektstart wurden vom Auftraggeber zusätzlich noch weitere Anforderungen an die Entwicklung, spezieller an die Programmierung, gestellt. Nach Absprache mit Prof. Dr. Heym nutzt das Team zur Entwicklung nun die Programmiersprache PHP in der Version 7.0.22 für die Serverseitige Programmierung. Zusätzlich wird serverseitig das PHP-Framework Yii2 verwendet. Clientseitig wird für die Oberflächenprogrammierung das Framework Bootstrap verwendet, die genaue Version dazu wird erst im Verlauf des Projektes festgelegt. Das Content Delivery Management wird über das Framework AngularJS programmiert.

## Entwicklertools

Das Entwicklerteam hat sich nach Absprache im ersten Teammeeting auf folgende Entwicklertools geeinigt; Für die Koordinierung der agilen Vorgehensweise wird die Projektmanagementsoftware Trello genutzt. Diese ist insbesondere deshalb so nützlich und wichtig für das Projekt, da sie eine ständige Erstellung von Aufgaben unterstützt, die unterschiedlich verteilt und priorisiert werden können. Die Teammitglieder werden außerdem benachrichtigt, wenn für sie eine neue Aufgabe erstellt wurde. Als Kollaborative Plattform, zur Kommunikation und Verbindung mit dem Auftraggeber wird die soziale Plattform SeeYou der Hochschule Hof verwendet. Dort wurde für das Team ein Space erstellt, in dem sie untereinander kommunizieren können, sowie wichtige Mitteilungen an die anderen Mitglieder weiterleiten können. Außerdem existiert dort auch ein weiterer Space, der für alle Entwicklerteams des Gesamtprojektes nutzbar ist. Dort können alle Mitglieder Fragen an den Auftraggeber

stellen. Dieser hat zusätzlich die Möglichkeit, neue Anforderungen oder generelle Mitteilungen an die Entwicklerteams weiterzuleiten.

## 4 Erkundung der Frameworks

In diesem Kapitel werden die im Projekt zu nutzenden Frameworks untersucht. Es wird festgehalten, inwieweit die einzelnen Module, die uns in den Frameworks zur Verfügung stehen, uns im Projekt weiterhelfen, um die Idee der Kollaboration im Spiel und die Programmierung des Prototypen umzusetzen.

### HumHub

Wie im vorhergehenden Kapitel beschrieben, beruht die Kollaboration in dem Prototypen dieses Projektes auf der Open-Source Software HumHub. Diese Software bietet dem Team eine soziale Plattform, die bereits über Nutzerbehandlung, Login und andere soziale Komponenten verfügt. Außerdem baut HumHub auf dem PHP-Framework Yii auf, das auch zum Umfang des Projektes gehört und auf das später noch einmal genauer eingegangen wird. Im folgenden wird die Plattform HumHub kurz analysiert und es wird auf die wichtigsten Funktionen eingegangen.

### Entscheidungsmatrix

In diesem Kapitel wird auf die einzelnen Module des Systems HumHub eingegangen. Diese werden analysiert und von den Teammitgliedern bewertet. Es soll außerdem auf die Bewertungsmethoden und die Gewichtungen der einzelnen Punkte eingegangen werden. Zum Schluss wird entschieden, welche Module genutzt werden, diese Entscheidung wird dann noch erläutert.

### Erklärung der Vorgehensweise

Es werden für das Projekt relevante Kriterien gesammelt, die dann so objektiv wie möglich nach ihrem Einfluss auf die Entwicklungsarbeit priorisiert werden. Danach werden diese Kriterien mit jedem Baustein abgeglichen und es wird darüber entschieden, wie gut ein Modul dieses Kriterium erfüllt. Alle Entscheidungen über die Priorität und das Erfüllen der Kriterien wurden im Team besprochen.

## **Bewertungskriterien**

Die Priorität eines Kriteriums wird in drei Stufen untergliedert, Hoch (3), Mittel (2) und Niedrig (1). Der Wert, mit dem ein Kriterium bewertet wurde, wird dann mit dem Wert des Einflusses multipliziert.

Der Einfluss eines Kriteriums auf das Projekt wird ebenfalls in drei Stufen unterteilt, Hoch (5), Mittel (3) und Niedrig (1).

Das Produkt aus Priorität und Einfluss bildet den Richtwert eines Kriteriums für ein Modul. Alle Kriterien werden danach aufsummiert, das Modul mit dem höchsten Gesamtwert ist demnach am geeignetsten für die Nutzung im Rahmen von HumHub.

Kriterien	Priorität	Einfluss	Polls	Einfluss	Tasks	Einfluss	Streams
Spaßfaktor	2	5	10	3	6	5	10
Umsetzungsmöglichkeit	3	5	15	1	3	1	3
Komplexität (für Entwickler)	3	5	15	3	9	1	3
Benutzer-freundlichkeit	2	5	10	3	6	5	10
Übersichtlichkeit	2	5	10	3	6	5	10
Arbeitsaufwand (für Entwickler)	1	5	5	3	3	1	1
Lernfaktor	3	1	3	5	9	5	15
Fehleranfälligkeit in Programmierung	1	3	3	3	3	3	3
Fehleranfälligkeit bei Nutzung	3	1	3	5	15	5	15
Summe			74		60		70

## **Bewertung**

Bestes Modul	Mittleres Modul	Schlechtestes Modul
--------------	-----------------	---------------------

Farben kennzeichnen das Abschneiden der Module.

## **Entscheidung**

Obwohl die „Polls“ am besten mit 42 Punkten abgeschnitten haben und die „Tasks“ am schlechtesten, entscheiden wir uns alle Drei (Polls, Tasks und Streams) zu verwenden.

„Polls“ sind am besten für Multiple-Choice-Fragen geeignet, wir möchten aber nicht alle Fragen als Multiple-Choice machen. „Tasks“ sind für Fragen mit Antworten gut geeignet und „Streams“ werden wir für Background-Information verwenden. Somit entsteht ein Abwechslungsreiches Spiel.

## Markdown

In diesem Kapitel wird auf die Syntax eingegangen, die in dem zu nutzenden System HumHub genutzt werden kann, um Texte zu formatieren. Außerdem wird kurz erläutert, welche Elemente genutzt und welche vermieden werden sollen.

### Syntax

Formatierung	Eingegebener Text	Zielform
Fett	<b>**Fettschrift**</b>	<b>Fettschrift</b>
Kursiv	<i>*Kursivschrift*</i>	<i>Kursivschrift</i>
Aufzählungslisten	<ul style="list-style-type: none"> <li>* Aufzählung 1</li> <li>(Leerzeichen nach dem Stern nicht Vergessen)</li> <li>* Aufzählung 2</li> </ul>	<ul style="list-style-type: none"> <li>• Aufzählung 1</li> <li>• Aufzählung 2</li> </ul>
Nummerierte Listen	<ol style="list-style-type: none"> <li>1. Schritt 1</li> <li>2. Schritt 2</li> </ol>	<ol style="list-style-type: none"> <li>1. Schritt 1</li> <li>2. Schritt 2</li> </ol>
Verschachtelte Listen	<ul style="list-style-type: none"> <li>* Erste Ebene</li> <li>* zweite Ebene mit 2 Leerzeichen vor dem Stern/Zahl</li> </ul>	<ul style="list-style-type: none"> <li>• Erste Ebene</li> <li>◦ Unterebene</li> </ul>
Überschriften	# Überschriftebene 1 ## Überschriftebene 2 ### Überschriftebene 3 #### ....	<b>Überschriftebene 1</b> <b>Überschriftebene 2</b> <b>Überschriftebene 3</b> ...
Blockzitate	> Blockzitate müssen mit einer Leerzeile beginnen und enden	Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonu.
Inlinecode	Beispiel für `Inlinecode`.	<code>Inlinecode</code>
Codeblocks	```	Codeblock

	Beispiel für Codeblock ```	
Bilder	![alt] ( <a href="http://sampleurl.de/logo.png">http://sampleurl.de/logo.png</a> )	 <i>Abbildung 1:</i> <i>Markdown</i>
Links	[Anzeigetext für Link] ( <a href="http://www.google.de">http://www.google.de</a> )	<a href="#">Anzeigetext für Link</a>
Bilder mit Link	![Alt](imageurl)(link)	 <i>Abbildung 2:</i> <i>Markdown</i> Mit Link
Horizontale Linie	---	---

## Nutzungsregeln

Um die Wichtigkeit und Übersichtlichkeit von Beiträgen auf HumHub zu erhöhen, kann die Markdown Sprache verwendet werden. Befehle die verwendet werden können werden in der angeführten Syntax Tabelle aufgelistet.

## AngularJS

Im folgenden Kapitel werden die Funktionen des Frameworks AngularJS analysiert und zum weiteren Nutzen für das Projekt dokumentiert.

### Analyse der Funktionen von AngularJS

Im Folgenden werden nun einige Funktionen und Filter von AngularJS aufgezählt und erläutert, welche wir in unserem Projekt verwenden können. Dies stellt den momentanen Stand dar, kann sich jedoch jederzeit verändern.

#### 1. Services

In AngularJS ist ein Service eine Funktion oder ein Objekt, das für eine AngularJS-Anwendung verfügbar und auf diese beschränkt ist.

Service	Analyse
<b>\$exceptionHandler:</b> <i>Behandelt Exceptions, indem er diese an die \$log.error weiterleitet.</i>	Dieser Service ist nützlich, wenn beispielsweise ein timeout oder ein anderer Fehler auftritt. Durch den \$exceptionHandler wird gewährleistet, können wir die Fehler in einen Log auf dem Server schreiben, was uns den Umgang mit diesen erleichtern wird.
<b>\$filter:</b>	Siehe 2. Filter.
<b>\$http:</b> <i>Der \$http Dienst ist ein zentraler AngularJS-Dienst, der die Kommunikation mit HTTP-Servern entweder über das XMLHttpRequest-Objekt des Browsers oder über JSONP ermöglicht.</i>	Da wir speziell in unserem Projekt viel mit Servern arbeiten, benötigen wir diesen Service um zwischen den Servern Kommunikation zu gewährleisten.

<b>\$location:</b> <i>Analysiert URL in der Adressleiste des Browsers und stellt diese für die Anwendung zur Verfügung. Mit diesem Dienst können Änderungen an der URL übernommen und angezeigt werden.</i>	Mit Hilfe des \$location können wir in unserem Projekt die unschöne und unübersichtliche in eine kurze, prägnante und verständliche URL umwandeln und festlegen. Dies hilft zu einer schöneren und ansprechenderen Seite. Außerdem kann man sich so den Link zu der Seite schnell merken und auf sie zugreifen.
<b>\$log:</b> <i>Einfacher Service für die Protokollierung. Schreibt Nachricht sicher in die Konsole des Browsers.</i>	Der \$log-Service ist für unser Projekt sehr interessant, da dieser uns ermöglicht, leichter debuggen zu können.
<b>\$parse:</b> <i>Konvertiert AngularJS-Expressions in eine Funktion.</i>	Der parse-Service kann in uns beim Programmieren einige Arbeit ersparen. Dadurch, dass er automatisch in Funktionen konvertiert, müssen wir das nicht manuell machen und sparen Zeit, welche wir wieder in andere Arbeiten stecken können.
<b>\$q:</b> <i>Führt Funktionen asynchron aus und deren Rückgabewerte können nach der Verarbeitung verwendet werden.</i>	Dieser Service findet eine Verwendung bei zum Beispiel dem Collaborations-Stream. Während der User auf eine Antwort von anderen Spielern wartet, kann er in der Zwischenzeit wieder andere Aufgaben bearbeiten und weiter lernen.
<b>\$timeout:</b> <i>window.setTimeout in AngularJS. Alle Exceptions die bei der Verarbeitung der zu verzögerten Funktion auftreten, werden an den \$exceptionHandler delegiert.</i>	Kann in unserem Projekt für eine höhere Sicherheit der Daten sorgen, indem der User beispielsweise nach einer bestimmten Zeit ausgeloggt wird.

## 2. Filter

In AngularJS ist ein Filter eine Formatierungsmöglichkeit für Daten.

Filter	Analyse
<b>Filter:</b> <i>Wählt eine Teilmenge von Elementen aus einem Array aus und gibt diese als ein neues Array zurück.</i>	<p>Dieser Filter kann genutzt werden, wenn man nur einen Teil von in einem Array gespeicherte Daten braucht und mit diesen weiterarbeiten möchte.</p> <p>Beispielsweise kann das beim Einteilen der Mathematik-Aufgaben nützlich sein.</p>
<b>Number:</b> <i>Formatiert eine Zahl zu einem Text.</i>	<p>Number kann sehr sinnvoll sein, um aus Zahlen einen Text zu formatieren. Der Number-Filter ist speziell in unserem Projekt sinnvoll für zum Beispiel Textaufgaben beziehungsweise für die Frage-Antwort-Aufgaben. Hier wäre es nämlich von Vorteil, wenn man die Aufgaben direkt in Text umwandeln kann, um sie besser und leichter zu verarbeiten.</p>
<b>Date:</b> <i>Formatiert ein Datum in ein gewünschtes String-Format um.</i>	<p>Dieser Filter eignet sich sehr gut um Daten zum Beispiel für die Spielstände der Nutzer zu speichern und dieses als String anstatt als Date weiterzubearbeiten. Strings lassen sich in dem meisten Fällen leichter bearbeiten, als ein Date-Format.</p>
<b>Json:</b> <i>Konvertiert JavaScript-Objekte in JSON-Strings um.</i>	<p>JSON wird benötigt, um mit dem HTTP-Server zu interagieren.</p>
<b>Lowercase &amp; Uppercase:</b> <i>Konvertieren Zeichenfolgen in Klein- bzw. Großbuchstaben.</i>	<p>Diese Filter sind sinnvoll um Zeichenfolgen in eine einheitliche Konvertierung zu wandeln. Dies ist hilfreich, wenn zum Beispiel Strings verglichen werden müssen. Dies wäre bei Frage-Antwort-Aufgaben anzuwenden, um eben die eingegeben Antworten mit den korrekten</p>

	<p>Lösungen zu vergleichen.</p> <p>In unserem Fall haben wir uns für <b>lowercase</b> entschieden, da diese Variante definitiv verbreiteter ist und im Englischen die Kleinschreibung die Norm darstellt. Dadurch muss dann im besten Fall nicht viel umgewandelt werden, um die Eingaben vergleichen zu können.</p>
<b>OrderBy:</b> <i>Gibt sortiertes Array zurück.</i>	<p>Der OrderBy-Filter wird gebraucht, um Dinge nach einer bestimmten Reihenfolge zu sortieren. Hierbei wäre es denkbar, diesen Filter unter anderem bei der Auswertung des Votings im Stream einzusetzen. Man würde somit direkt ohne großen Aufwand die Antworten nach den Votes sortiert zurückbekommen.</p>

## Bootstrap

Im Projektverlauf wurde festgelegt, dass Oberflächen nur mit dem Framework Bootstrap implementiert werden sollen. Demnach wird die aufwändigste Oberfläche, welche nicht vom Framework Yii2 übernommen werden kann, auch mit Bootstrap aufgebaut. Diese Oberfläche ist das User-Frontend, über das in späteren Kapiteln<sup>1</sup> noch genaueres beschrieben wird.

## Yii

In diesem Kapitel wird die Nutzung des PHP-Frameworks Yii erläutert. Es wird auf die Installation und die genutzten Features eingegangen.

---

<sup>1</sup> Siehe Kapitel Screendesign, User Stories

## Yii2 Basic Installation

- 1. Ordner namens „basic“ auf dem Server erstellen**
- 2. yii2 mit Composer installieren:**

Befehl:

```
composer create-project --prefer-dist yiisoft/yii2-
app-basic yii-application
```

- 3. Initialisierung**

Befehl:

```
/usr/bin/php var/www/html/basic/yii-application/init
```

## Yii2 Advanced Installation

- 1. Ordner namens „advanced“ auf dem Server erstellen**
- 2. yii2 mit Composer installieren:**

Befehl:

```
composer create-project --prefer-dist yiisoft/yii2-
app-advanced yii-application
```

- 3. Initialisierung**

Befehl:

```
/usr/bin/php var/www/html/advanced/yii-application/init
```

## 5 Entwicklung des Prototypen

Im folgenden Kapitel wird die Entwicklung des Prototypen dokumentiert. Es werden alle Überlegungen, die in die Spielidee und in das Konzept der Kollaboration einfließen, aufgenommen. Dazu gehören unter Anderem die Regelsätze für die Nutzung der künstlichen Intelligenz und der Vergabe von Punkten, um im Spiel voran zu kommen, Designs der Spiele- und Kollaborationsseiten und User Stories, die die Interaktion mit dem Prototypen verdeutlichen.

### Regelsätze

Hier wird auf die vom Auftraggeber Prof. Dr. Jürgen Heym definierten Regelsätze für das kollaborative Spiel eingegangen. Es werden Alle 3 Regelsätze nochmals formuliert und anschließend mögliche Lösungen präsentiert.

#### Regelsatz 1

Verteilung von Wissenspunkten und Kollaborationspunkten und dessen Verhältnis.

#### Punktevergabe

In dem Spiel werden zwei Arten von Punkten vergeben, zum einen Kollaborationspunkte und zum anderen Wissenspunkte. Kollaborationspunkte erreicht der Nutzer, indem er Fragen beantwortet, die möglicherweise von anderen Spielern gestellt wurden und ihnen beim Lösen von Aufgaben helfen. Wissenspunkte sammelt der User durch das richtige Beantworten von Mathe-Aufgaben, welche diesem im normalen Spielablauf gestellt werden. Wie der Name schon sagt, verdeutlichen Wissenspunkte, dass man sich nach deren Erreichen erfolgreich Wissen angeeignet hat.

Nachkommend werden die einzelnen Arten an Punkte zu gelangen aufgezählt und näher erklärt.

### **Multiple-Choice**

Multiple-Choice-Fragen sind Fragen, bei denen insgesamt vier Antwortmöglichkeiten vorgegeben sind und eine gewisse Anzahl von diesen zur Lösung der gestellten Aufgabe ausgewählt werden müssen. Dabei ist es vorgegeben, wie viele Antworten benötigt werden. Der Fall von vier richtigen Antwortmöglichkeiten wird jedoch ausgeschlossen, da dies sonst zu keinen Lernerfolg führen würde.

Die Punktevergabe setzt wie folgt zusammen:

Hat der Spieler keine einzige richtige Antwort, bekommt er keinen Punkt.

Gibt es eine einzige richtige Antwort und er wählt diese aus, erhält er drei Wissenspunkte, da die Wahrscheinlichkeit, dass er genau diese anklickt gleich 1/4 ist und somit relativ gering. Ist die gewählte Antwort halb richtig, erhält er nur einen Punkt.

Dies wird mithilfe eines Beispiels erklärt. (siehe Abb.1)

Frage: Wie lauten die Nullstellen der folgenden Gleichung? $y = (35-x_1)(x_2+(-2))$ <input type="radio"/> 35, 2 <input checked="" type="radio"/> 35, (-2) <input type="radio"/> 7, 3 <input type="radio"/> 88, -15
---

*Abbildung 3: Bsp. halb richtige Antwort*

Die folgende Antwort ist zur Hälfte richtig, da die erste Nullstelle 35 ist, die zweite aber 2 und nicht -2. Deshalb erhält der User einen Punkt.

Sind zwei Antworten richtig und der Nutzer sucht diese aus, sammelt er zwei Punkte, da eine 50/50 Chance besteht. Entscheidet er sich jedoch nur für eine der Richtigen, kriegt er einen Wissenspunkt.

Zuletzt besteht die Möglichkeit von drei richtigen Antworten. Wählt der Spieler alle drei Richtigen, bekommt er einen Punkt, da in diesem Fall zwei seiner Antworten immer richtig sind und die Wahrscheinlichkeit richtig zu liegen bei 3/4 liegt, was sehr hoch ist. Der Nutzer könnte hierbei somit raten und würde dennoch mindestens zwei richtige Antworten ankreuzen. Der Lerneffekt ist dabei sehr gering, weshalb er hier auch nur einen Wissenspunkt verdient. Jedoch wird die Möglichkeit von drei Richtigen nicht ausgeschlossen, da sich der Spieler allein durch das Lesen der Frage und der Antworten schon ein wenig Wissen aneignet, welches er durch weiteres Beantworten von Fragen vertiefen und festigen wird. Ist bei diesem Fall eine Antwort des Nutzers falsch, bekommt er keinen Punkt, da wie oben schon beschrieben, zwei Antworten automatisch immer korrekt sind.

### **Frage-Antwort**

Bei Frage-Antwort-Aufgaben wird dem User eine Frage gestellt, welche er mit eigenen Worten beantworten muss. Dabei kann er in ein vorgegebenes Textfeld seine Lösung eingeben, welche dann ausgewertet wird. Bei dieser Art von Fragen sammelt der Spieler erneut Wissenspunkte. Diese Art der Aufgaben sind für den User komplexer zu lösen, wie schon in der Entscheidungsmatrix festgehalten, weshalb ihn hier mehr Punkte erwarten als bei Multiple-Choice-Fragen.

<p>Frage: Welchen Wert hat x?</p> $x + 10 = 20$ <p>Nutzer-Eingabe: <b>10</b></p>	<p>Frage: Was ist die Lösungsmenge?</p> $x+7a = 19a-25 \quad (a,x \in \mathbb{Q})$ <p>Nutzer-Eingabe: L = {12a - 25}</p>
--	--

Abbildung 4: Bsp. leichte Aufgabe

Abbildung 5: Bsp. schwere Aufgabe

Die Frage-Antwort-Aufgaben werden in zwei Kategorien unterteilt, zum einen gibt es schwere (siehe Abb.2) beziehungsweise umfangreichere Aufgaben, zum anderen leichtere und nicht so komplex (siehe Abb.3) zu beantwortende Fragen. Außerdem gibt es Aufgaben mit mehreren möglichen Antworten, welche extra zu behandeln sind. Es wird bei der Aufgabe angemerkt, ob sie eine der schweren oder der Leichten ist, damit der Nutzer weiß, wie viele Punkte ihn erwarten.

Wie immer gilt, gibt der Benutzer eine falsche oder ungültige Antwort, werden ihm keine Punkte gut geschrieben.

Beantwortet der Spieler eine der komplizierteren Fragen korrekt, so erhält er fünf Punkte.

Löst er eines der simpleren Probleme richtig, so sammelt er vier Wissenspunkte.

Bei den Aufgaben mit mehreren Antwortmöglichkeiten werden zwei Arten zusammengefasst. Die Erste beinhaltet das Aufzählen mehrerer Antworten auf eine Frage, die Andere das Darstellen eines mehrschrittigen Rechenweges. Bei beiden gibt es pro richtige Antwort beziehungsweise Rechenschritt vier Punkte. Dies wird darauf begründet, dass die Wahrscheinlichkeit, dass man die Antwort wie bei Multiple-Choice erraten kann, ohne sie zu wissen, gleich null ist. Das Wissen des Spielers wird bei dieser Fragen-Art wirklich auf den Prüfstand gestellt, da er diese Vorkenntnisse und ohne Hilfe nicht beantworten kann. Trifft er keine korrekte Lösung, bekommt er keinen Punkt.

Dies wird nun an einem Beispiel verdeutlicht. (siehe Abb.)

Frage: Was ist die Lösungsmenge? Gib auch Lösungsweg mit an. $x+7a = 19a-25 \quad (a,x \in \mathbb{Q})$
Nutzer-Eingabe: $x+7a = 19a-25 / -7a$ $x = 12a-25$ $L = \{12a - 25\}$

*Abbildung 6: Bsp. Fragen-Antwort-Aufgabe mit Rechenweg*

Im oberen Beispiel wird der Lösungsweg verlangt. Das bedeutet die wichtigsten Schritte werden gewertet. Der Nutzer bekommt somit pro Rechenschritt vier Punkte, in diesem Beispiel wären dies acht Punkte.

## **Kollaboration**

Bei einer Kollaboration sammelt man, wie der Name schon sagt, die sogenannten Kollaborationspunkte. Unter einer Kollaboration in diesem Spiel versteht man, die gegenseitige Unterstützung und Zusammenarbeit der Nutzer. Hat ein Spieler beispielsweise ein Problem seine Aufgabe zu lösen oder generell eine Frage zu dem Themengebiet, kann er diese in einem für die Nutzer regenerierten Chat stellen. Auf diese Unterhaltung kann jeder Spieler des selben Levels zugreifen. Anderen Spielern wird die Frage dann angezeigt und können diese, wenn sie das nötige Wissen dazu haben, im Chat beantworten. Beim Beantworten der Fragen anderer Nutzer, können Spieler ihre Kollaborationspunkte sammeln.

Auf eine Frage ist es mehreren Spielern möglich zu antworten. Für die Antworten können dann ebenfalls die Nutzer voten und die drei Personen mit den meisten Upvotes, also mit den besten Erklärungen, erhalten ihre Punkte. Das Voting nach dem Beantworten der Probleme durch die anderen Nutzer dauert stetig zehn Minuten,

danach werden die Punkte verteilt. Antworten weniger als drei Personen oder sogar keine, werden diese Lösungen „bepunktet“ oder die künstliche Intelligenz schreitet ein und liefert eine Hilfestellung auf die Frage.

Die Lösung mit den meisten Upvotes erhält fünf Kollaborationspunkte, da diese Art der Beantwortung ungefähr mit Frage-Antwort-Aufgaben gleichzusetzen ist. Der User muss beides mal eine eigene Antwort beziehungsweise Hilfe formulieren und die Frage sollte jedes Mal korrekt beantwortet werden. Der einzige Unterschied ist, dass der Nutzer Kollaborationspunkte anstatt der Wissenspunkte erhält.

Der zweite Platz bekommt vier Punkte und der dritte drei, was sich eben auf die absteigende Platzierungen zurückführen lässt.

### **Punkteverhältnis**

Das Basiskonzept des Spiels basiert auf drei Schwierigkeitsstufen: Amateur, Fortgeschritten und Profi.

Die erste Stufe und somit der Einstieg in das jeweils gewählte Themengebiet ist als leichtes Anfängerlevel angedacht. In diesem werden die Nutzer hauptsächlich damit beschäftigt das Thema kennen zu lernen und dieses grob zu verstehen, damit sie in der Lage sind leichte Aufgaben in diesem Bereich zu lösen.

Das darauffolgende Level zielt auf die fortgeschritteneren Kenntnisse ab und vertieft das Wissen des ersten Levels. Hierbei wird auf die jeweils bereits erreichten Grundkenntnisse aufgebaut. Die Aufgaben werden zunehmend schwerer und fordern den User somit immer mehr.

Hat der Spieler dann das erwartete Niveau erreicht, kommt er in eine noch höhere Anforderungsklasse, welche für die Profis auf dem Gebiet gedacht ist. Hier werden besonders anspruchsvolle Aufgaben gestellt, wodurch die erworbenen Kenntnisse des Spielers auf den Prüfstand gestellt und bis ins kleinste Detail verbessert werden. Schließt der Benutzer diese Stufe ab, hat er sehr gute Kenntnisse in dem Themenbereich und kann sich nun anderen Problematiken der Algebra zuwenden.

Im Folgenden werden nun die Verhältnisse der Kollaborations- und Wissenspunkte der jeweiligen Stufen erläutert und begründet.

### ***Amateur (Level 1)***

Das Punkteverhältnis in der Anfängerstufe beträgt 25:75. Dies bedeutet, dass der Nutzer 25 Prozent seiner Punkte durch Kollaboration und 75 Prozent durch Wissen sammeln muss, damit er dieses Level abschließen kann.

Die Auswahl dieses Verhältnisses beruht darauf, dass in dieser Stufe zunächst der Fokus darauf gelegt wird, sich Wissen und Grundkenntnisse anzueignen. Deswegen sollen deutlich mehr Wissenspunkte als Kollaborationspunkte erreicht werden. Ein weiterer Grund ist, dass die Aufgaben dieses Levels noch relativ leicht und in der Norm schnell zu lösen sind, weshalb man in kurzer Zeit im Vergleich mit einem höheren eher Wissenspunkte sammeln kann. Es ist außerdem viel schwerer in dieser Niveaustufe Kollaborationspunkte zu sammeln, da die Aufgaben für viele Nutzer ohne Hilfe zu bewältigen sind und somit die Kollaborationsfunktion nicht sehr oft in Gebrauch genommen wird. Jedoch wird trotzdem noch Wert auf diese gelegt, da es immer Personen gibt, die sich selbst bei simpleren Problemen schwer tun und Hilfe benötigen.

### ***Fortgeschritten (Level 2)***

Folgendem Level wurde das Verhältnis 50:50 zugeordnet.

In dieser Stufe gleichen sich die Kollaborations- und Wissenspunkte aus, da hier beide Aspekte gleich wichtig sind und benötigt werden. Es muss sowohl das Wissen, das zuvor angeeignet wurde, vertieft als auch Neues erlernt werden.

Die Nutzer, die sich mit den nun mittelschweren Aufgaben etwas schwer tun, haben die Möglichkeit sich durch die Zusammenarbeit mit anderen helfen zu lassen, wodurch die guten Spieler wiederum Kollaborationspunkte sammeln können. Dennoch reißt die Motivation sich neues Wissen anzueignen nicht ab, da Wissenspunkte erreicht werden müssen. Das Level ist somit ausgewogen und sowohl die guten als auch die schlechten Spieler werden gefördert und können dadurch gemeinsam das Ziel der Fortgeschrittenen-Stufe erreichen.

### ***Profi (Level 3)***

In der Profi-Stufe besteht zwischen den Kollaborations- und Wissenspunkten das Verhältnis 65:35.

In dem höchsten Level werden die Fragen auf ein hohes Schwierigkeitsniveau gehoben, somit werden deutlich mehr Nutzer Probleme haben, die Fragen und Aufgaben zu lösen. Durch gegenseitiges Helfen wird das bereits erworbene Wissen angewendet und vor allem vertieft, was besonders das Ziel in dieser Stufe ist. Deshalb ist vor allem die Kollaborationsfunktion, bei der sich die Nutzer gegenseitig unterstützen und helfen können, in diesem Level von hoher Notwendigkeit, was die hohe Gewichtung der Kollaborationspunkte erklärt. Die Wissenspunkte fallen somit etwas in den Hintergrund, dürfen aber nicht komplett vergessen werden, da der Nutzer stets Motivation benötigt, um weiterhin zu lernen und sich Wissen anzueignen.

## Regelsatz 2

Wann greift eine künstliche Intelligenz bei der Kollaboration ein und wie das realisiert werden kann.

### ***Warum die KI hilft bzw. Hilferufe simuliert***

Das Lernsystem beruht auf der Kollaboration der Nutzer, sind allerdings keine Nutzer in der Lage einem hilfebedürftigen Nutzer zu helfen, muss das System diese Aufgabe übernehmen. Hierfür wird eine KI entwickelt, die auf die Hilferufe der Nutzer reagieren und passend antworten kann.

Weiterhin sollen die Nutzer in der Lage sein Kollaborationspunkte zu sammeln, auch wenn kein anderer Nutzer Hilfe benötigt oder die Probleme der anderen Nutzer zu anspruchsvoll sind. Deshalb generiert die KI automatisch Hilfeafrufe an Spieler, falls nicht genug natürliche Hilferufe gestellt werden, und gibt wiederum Antworten auf die eigenen Fragen, falls keine Antwort von mindestens einem Spieler erfolgt ist. Diese Antworten werden ebenfalls bewertet, um die KI-Handlungen nicht von spielergesteuerten Handlungen unterscheidbar zu machen. Dies hat auch zur Folge, dass das System belebt wirkt und die Nutzer leichter motiviert sind selbst Hilferufe zu generieren und auf die anderer zu antworten.

### ***Hilferufe***

Hilferufe sind ein separater Chat, der von einem Nutzer oder der KI generiert werden kann. Je nachdem wie lange der Hilferuf bereits besteht, können verschiedene Gruppen in dem Chat antworten, näheres hierzu in Regelsatz 3. Jeder Spieler, der auf einen Hilferuf reagiert hat, bekommt 50 Sekunden nachdem der Hilferuf generiert worden ist, alle abgegebenen Hilfestellungen für fünf Minuten angezeigt, sollte die Gruppe der Nutzer nicht in einem gemeinsamen Zeitraum arbeiten, sondern über den Verlauf einer Woche verteilt, bietet sich hierfür an diese Zeit auf einen Tag zu

erhöhen. Die Hilferufe und die dazugehörigen Antworten werden nach einer gewissen Zeit gelöscht, um zu verhindern dass die Nutzer lediglich alte Hilferufe durchforsten anstatt selber Hilferufe zu generieren

### ***Wo die KI hilft***

Die KI hilft, wenn ein Nutzer einen Hilferuf generiert hat und kein anderer Nutzer ihm helfen konnte. Außerdem stellt die KI generierte Hinweise zu allen Aufgaben zum Kauf zur Verfügung.

### ***Wann die KI hilft***

Es wird im folgenden kurz dargestellt, wann die künstliche Intelligenz eingreift und weshalb.

### **Beim Kauf eines Hinweises**

Um dem Nutzer die Antwort nicht vorwegzunehmen, soll die KI erst einschreiten, wenn dieser um Hilfe bittet. Dies teilt der Nutzer mit, indem er auf den „Hilfe“-Button klickt, er wird daraufhin gefragt ob er einen Hinweis kaufen möchte.

### **Nach dem Generieren eines Hilferufs**

Damit die anderen Nutzer eine Möglichkeit haben, Hilfestellung zu leisten und so erfolgreich Kollaboration zu betreiben, soll die KI 50 Sekunden warten bevor sie dem hilfebedürftigen Nutzer hilft. Diese Zeit ist bewusst lang gewählt um auch Nutzer, die das Zehnfingersystem noch nicht beherrschen, nicht zu benachteiligen. Sie darf aber auch nicht zu lange ausfallen, da der hilfebedürftige Nutzer sonst versucht ist, sich mit neuen Problemen zu beschäftigen.

## **Wie die KI hilft beim Kauf von Hinweisen**

Die KI soll dem Nutzer anfangs nur Hinweise in die richtige Richtung geben, soll aber bei Bedarf den Nutzer durch die Aufgabe leiten können. Deshalb sind die Hinweise in kleine Teile zerlegt und müssen vom Nutzer einzeln gekauft werden, hierdurch können schlechtere Nutzer durch die Aufgabe geleitet werden und bessere müssen nur die Hinweise kaufen, die sie auch benötigen.

## **Wenn der Nutzer keine Lösung hat**

Hat der Nutzer keinen Lösungsansatz finden können, kann ihm die KI eine Reihe von, für diese Aufgabe generierten, Hinweisen geben.

Beispiel:

Aufgabe:

$$0 = -20x^2 + 320$$

kleineres  $x = -4$  ; größeres  $x = 4$

Hinweise:

1. Versuche von beiden Seiten der Gleichung 320 zu subtrahieren.
2. Versuche beide Seiten der Gleichung durch 20 zu dividieren.
3. Versuche beide Seiten der Gleichung durch -1 zu dividieren.
4. Versuche von beiden Seiten der Gleichung die Wurzel zu ziehen.

- **Möglichkeit 1:**

Die Hinweise müssen in der richtigen Reihenfolge kommen und vom Nutzer mit den Wissenspunkten, die er für die Aufgabe erhalten würde, bezahlt werden. Ein Hinweis soll einen Wissenspunkt kosten, der Nutzer kann diese Punkte zurück erhalten, indem er die Aufgabe, eine Woche später, korrekt löst.

- **Möglichkeit 2:**

Die Hinweise müssen in der richtigen Reihenfolge kommen und vom Nutzer mit Kollaborationspunkten bezahlt werden. Ein Hinweis soll einen Kollaborationspunkt kosten, der Nutzer kann diese Punkte zurück erhalten, indem er auf Hilferufe anderer Nutzer oder der KI reagiert.

### **Wenn der Nutzer eine falsche Lösung hat**

Ist eine Antwort, auf die Aufgabe des Nutzers vorhanden, so kann die KI eine seinem Fortschritt angepasste Hilfestellung bieten.

Aufgabe:

$$0 = -10x^2 + 320$$

Ergebnis auf zwei Nachkommastellen gerundet.

kleineres  $x = -5,66$  ; größeres  $x = 5,66$

### **Hilfestellung bei einem Rundungsfehler**

Antwort des Nutzers: kleineres  $x = -5,65$  ; größeres  $x = 5,65$

Hilfestellung: Ab 5 wird aufgerundet, nur 4 oder kleiner wird abgerundet.

### **Hilfestellung bei einer richtigen Antwort**

Antwort des Nutzers: kleineres  $x = -2,66$  ; größeres  $x = 5,66$

Hilfestellung: Das kleinere  $x$  ist nicht korrekt, das größere hingegen ist korrekt.  
(Diese Hilfestellung ist für den Nutzer kostenlos, bei Bedarf kann er die selben Hinweise kaufen, die er bekommen würde, wenn er keine Antwort gegeben hätte.)

## **Hilfestellung bei keiner richtigen Antwort**

Antwort des Nutzers: kleineres  $x = -2,66$  ; größeres  $x = 2,66$

Hilfestellung: Beide  $x$  sind nicht korrekt. (Diese Hilfestellung ist für den Nutzer kostenlos, bei Bedarf kann er die selben Hinweise kaufen, die er bekommen würde, wenn er keine Antwort gegeben hätte.)

## **Wie die KI beim Antworten auf Hilferufe hilft**

Um Hilfe leisten zu können, muss die KI zunächst verstehen, wo das Problem des Nutzers liegt. Dies kann zum einen durch Machine Learning aber auch durch das Ausfiltern von Schlüsselwörtern realisiert werden. Machine Learning kann verwendet werden um die Antworten der Nutzer, die auf Hilferufe getätigten wurden zu analysieren, die KI soll lernen, was den Nutzern bei einem bestimmten Problem geholfen hatte und daraus eine Antwort generieren. Das Ausfiltern von Schlüsselwörtern soll ermöglichen, auf passende Kapitel des Lernstoffes zu verweisen, sodass der Nutzer durch Nachlesen des Stoffes das Problem selbstständig lösen kann.

## **Wie die KI Hilferufe generiert**

Die KI kann alte archivierte, vorgefertigte oder automatische Hilferufe generieren. Die archivierten Hilferufe sind alte Hilferufe von Nutzern die von Nutzern beantwortet werden konnten, die vorgefertigten sind von einer betreuenden Lehrkraft oder einem Administrator zu erstellen. Automatische Hilferufe müssen von der KI so generiert werden, dass sie dem Niveau der Nutzer entsprechen. Sie bewertet die Antworten der Nutzer auf Basis ihrer Qualität. Wenn Nutzer wenig Kollaborationspunkte haben, generiert die KI für diesen Nutzer gezielt Hilferufe.

## Weitere Beispiele für käufliche Hinweise

Im folgenden werden einige Beispiele gezeigt, wie die künstliche Intelligenz den Anwendern bei Bedarf Hinweise geben könnte.

### Beispiel 1

Aufgabe:

$$27c = 22c + 5x^2$$

kleineres  $x = \dots$ ; größeres  $x = \dots$

Hinweise:

1. Versuche alle Terme mit  $x$  und alle Terme ohne  $x$  auf verschiedene Seiten zu bringen.
2. Versuche auf beiden Seiten der Gleichung  $22c$  zu subtrahieren.
3. Versuche beide Seiten der Gleichung durch  $5$  zu teilen.
4. Versuche auf beiden Seiten der Gleichung die Wurzel zu ziehen.

### Beispiel 2

Aufgabe:

$$-x + 4a + 14 = 16a$$

$x = \dots$

Hinweise:

1. Versuche alle Terme mit  $x$  und alle Terme ohne  $x$  auf verschiedene Seiten zu bringen.
2. Versuche auf beiden Seiten der Gleichung  $4a$  zu subtrahieren.
3. Versuche auf beiden Seiten der Gleichung  $14$  zu subtrahieren.
4. Versuche beide Seiten der Gleichung durch  $-1$  zu teilen.

### Beispiel 3

Aufgabe:

$$3y + 9a - 6c = 27b$$

$$y =$$

Hinweise:

1. Versuche alle Terme mit y und alle Terme ohne y auf verschiedene Seiten zu bringen.
2. Versuche auf beiden Seiten der Gleichung 9a zu subtrahieren.
3. Versuche auf beiden Seiten der Gleichung 6c zu addieren.
4. Versuche beide Seiten der Gleichung durch 3 zu teilen.

## **Regelsatz 3**

Wie wird sichergestellt, dass alle Spieler, unabhängig von Ihren Fortschritten oder Kenntnissen in der Materie, gleichberechtigt Kollaborationspunkte sammeln können.

### ***Beschreibung des Problems***

Das Problem bei kollaborativem Lernen ist, dass das Niveau, auf dem sich die Schüler befinden, stets unterschiedlich ist. Es wird also immer der Fall sein, dass es einem Anwender leichter fällt, im Spiel voran zu kommen, als es einem schwächeren Schüler fällt. Das Grundkonzept des Spieles ist es ja, laut Vorgabe des Auftraggebers Prof. Dr. Heym, dass Spieler sowohl durch Fortschritte im Inhalt der Mathematik, aber auch durch Mithilfe für andere Schüler bei Problemen Punkte bekommen, wodurch sie im Spiel voranschreiten (Dieses Konzept wird in Regelsatz 1 beschrieben). Leider fällt es den schwächeren Schülern deutlich schwerer, anderen Mitspielern zu helfen, denn entweder haben sie selbst noch Probleme mit der Materie oder ein besserer Schüler hat ihnen die Möglichkeit der Mithilfe durch schnelleres Antworten vorweggenommen. Der folgende Abschnitt beschäftigt sich genau mit diesem Problem und erörtert mögliche Herangehensweisen, um das gleichmäßige Voranschreiten aller Schüler zu gewährleisten.

### ***Was unterscheidet gute von schlechten Spielern?***

Um zu differenzieren, ob die aktuelle Lage im Spiel ausgewogen und fair ist, muss man sich erst Gedanken darüber machen, was einen guten Spieler von einem schlechten Spieler unterscheidet. Im allgemeinen soll hier von einem schlechten Schüler gesprochen werden, wenn dieser weniger als 60% der bisher erreichbaren Wissenpunkte im Spiel erlangt hat. Ein gute Spieler hat demnach mehr als 60%.

Wissenpunkte werden im Laufe der Spiele unterschiedlich vergeben, man unterscheidet, ob ein Spieler eine mathematische Aufgabe komplett selbstständig

gelöst hat, er einen Tipp oder Mithilfe eines anderen Schülers in Anspruch nehmen musste oder ob er die Aufgabe gar nicht beantworten konnte.

### **Aufbau eines Fragezyklus**

Der Prozess einer Frage im kollaborativen Umfeld beginnt damit, dass ein Spieler mit seiner Aufgabe nicht voran kommt, er benötigt demnach Hilfe und stellt seine Frage öffentlich an alle anderen Mitspieler. Um das oben genannte Problem zu vermeiden wird der weitere Ablauf zeitlich unterteilt:

Sobald die Frage gestellt wurde, wird ein Timer gestartet. Dieser Timer erlaubt es in den ersten 20 Sekunden der Fragestellung nur den schlechteren Schülern, die Frage zu beantworten, damit sie etwas Vorlaufzeit gegenüber den guten Spielern haben, um eine geeignete Antwort zu verfassen. Sollte in diesen 20 Sekunden keiner die Frage beantwortet haben, so wird die Frage für alle Spieler freigegeben, außer für Spieler, die in der Vergangenheit zu oft und zu schnell auf Fragen geantwortet haben, um allen anderen die Möglichkeit zu nehmen, Kollaborationspunkte zu sammeln. Jetzt haben alle 15 Sekunden Zeit, die Frage zu beantworten. Sollten im Vorfeld Spieler aufgefallen sein, die zu oft und zu schnell Fragen beantworten, dann bekommen diese im Anschluss an die vergangenen 35 Sekunden nochmals 15 Sekunden Zeit, um die Frage zu lösen. Nach diesen 50 Sekunden wird die Frage an die künstliche Intelligenz weitergeleitet, mit der sich Regesatz 2 beschäftigt.

Es sollte noch mit angefügt werden, dass die Frage ab dem Zeitpunkt, in dem sie gestellt wurde, für alle Spieler sichtbar ist, aber immer nur für die oben genannten Zielgruppen beantwortbar ist. Außerdem wird der Timer zum stehen gebracht, sobald ein Spieler anfängt, eine Antwort einzutippen. Somit soll die Unterbrechung durch die künstliche Intelligenz unterbunden werden.

Zum Schluss werden dem Fragestellenden Spieler die Antworten angezeigt. Er hat dann die Möglichkeit, über 3 Icons auszuwählen, wie sehr ihm die Antwort des anderen Spieler geholfen hat. Es werden drei Smileys angezeigt, ein grüner, der

signalisiert, dass die Antwort sehr hilfreich war, ein gelber, der klarstellt, dass die Antwort durchaus weiter geholfen hat, aber dennoch zu Wünschen übrig lies, und einen roten Smiley, der klarstellt, dass die Antwort der Frage nicht zu genüge getan hat.



Abbildung 7: Smileys zur Bewertung der Hilfe

Die Kollaborationspunktevergabe ist in Regelsatz 1 klar beschrieben. Aufbauend auf diesen Werten bekommt ein Spieler für eine hilfreiche Antwort 100% der möglichen Antworten, eine weniger hilfreiche Antwort bekommt aber nur 50% ihrer möglichen Punkte. Eine schlechte Antwort soll aber nicht ganz leer ausgehen, denn der Aufwand, zu antworten soll trotzdem beachtet werden. Deshalb bekommt eine schlechte Antwort 10% ihrer Punkte. Ausnutzen dieser 10% durch sinnfreie Antworten ist allerdings nicht möglich, da dies durch die Bewertung der Spieler, die unnötig antworten, abgefangen wird.

### **Berechnung von Richtwerten zur Bewertung von Spielern**

Zur Berechnung der Verhältnisse zwischen Antworten bei Spielern werden nur abgeschlossene Kapitel genutzt. In Regelsatz 1 wird die Voraussetzung an Kollaborationspunkten für ein Voranschreiten in das nächste Kapitel beschrieben. Somit wird ein Minimum an Punkten festgehalten. Das Maximum orientiert sich an dem Spieler, der in dem Kapitel die meisten Kollaborationspunkte gesammelt hat und der das Kapitel beendet hat. Aus den Beiden Extremwerten wird nun der Durchschnitt

errechnet. Wenn ein Spieler mit seinen Kollaborationspunkten in abgeschlossenen Kapiteln 25% über dem Durchschnitt liegt, so ist dieser Spieler ein Kandidat für solche, die potentiell Fragen zu oft beantworten, um anderen Spielern die Möglichkeit zur Hilfe zu nehmen. Um die Spieler, die den anderen das Kollaborieren erschweren, eindeutig zu identifizieren, werden in den oben errechneten Wert zwei weitere mit eingenommen.

Der erste der beiden Werte setzt sich zusammen aus folgenden Einflüssen:

- Anzahl der Fragen, auf die von aktiven Spielern, nicht von der künstlichen Intelligenz, geantwortet wurde.
- Anzahl der Fragen, die der zu betrachtende Spieler selbst beantwortet hat.
- Gesamtanzahl der teilnehmenden Spieler im System.

Der zweite der beiden zusätzlichen Werte betrachtet die Antwortzeit, die der verdächtige Spieler gebraucht hat, um seine Antworten im Durchschnitt zu geben.

Beispiel Formel für das Errechnen des Wertes, der einen Spieler als übermäßigen Beantworter identifiziert:

- |          |  |
|----------|--|
| <i>a</i> | = Anzahl der von Spielern beantworteten Fragen                                     |
| <i>b</i> | = Anzahl der vom zu betrachtenden Spieler beantworteten Fragen                     |
| <i>c</i> | = Anzahl der Spieler, die am System teilhaben                                      |
| <i>t</i> | = Zeit, die der zu betrachtende Spieler im Schnitt für eine Antwort braucht (in s) |

$$\text{wenn } \frac{a}{c} \leq b * \sqrt{\frac{15s}{t}} , \text{ dann ist der Spieler zu eifrig.}$$

## Beispiel

### Aufgabe:

Schreibe die Gleichung, die du erhältst, wenn du in

$$2x+a=b, \text{ mit } (a,b,x \in \mathbb{Q})$$

2 statt a und 10 statt b setzt und bestimme deren Lösungsmenge.

### Frage:

Schüler weiß nicht, was  $a, b, x \in \mathbb{Q}$  bedeutet.

„Was bedeutet das Q in der Aufgabe oben, komme da nicht weiter...“

### Ablauf:

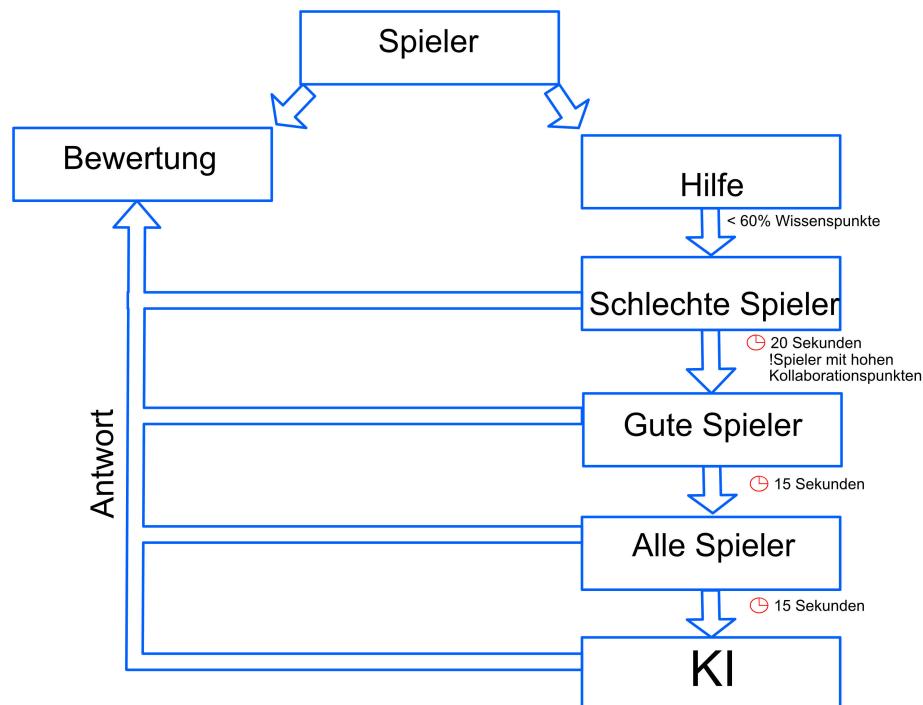


Abbildung 8: Ablauf der Fairness-Regelung

## Aufteilung der Website

Im folgenden Kapitel wird beschrieben, wie die Zugriffe auf die Website aufgeteilt werden. Im groben wird die Website für zwei Arten von Benutzern zugänglich sein, einmal für reguläre Benutzer, die die Website zum Lernen nutzen und einmal für Administratoren, die Lerngruppen organisieren. Alle Benutzergruppen und deren Sicht auf die Website wird im Folgenden erläutert.

### User-Frontend

Wie bereits angedeutet ist eine der Benutzergruppen die der lernenden Schüler. Sie haben keine administrativen Rechte und bekommen deshalb nur die Teile der Website zu sehen, die sie auch nutzen können. Deshalb wird ihnen nur das User-Frontend gezeigt.

Folgende Schritte können von den Usern unternommen werden:

#### 1. Login

Im User-Frontend hat der Benutzer die Möglichkeit sich einzuloggen oder zu registrieren. Nach dem erfolgreichen Login kann er über die Navigation zu anderen Inhalten springen.

#### 2. Account erstellen

Wenn ein Benutzer keinen gültigen Account hat und sich registrieren möchte, muss er für die Registrierung einen Token<sup>2</sup> eingeben um fortzufahren. Hat der Benutzer einen gültigen Token, kann er sich registrieren, in dem er einen neuen Benutzernamen<sup>3</sup> und ein Password eingibt.

---

<sup>2</sup> Ein Token kann nur von einem Administrator erstellt werden, später beschrieben

<sup>3</sup> Benutzername ist nicht das richtige Name des Benutzers

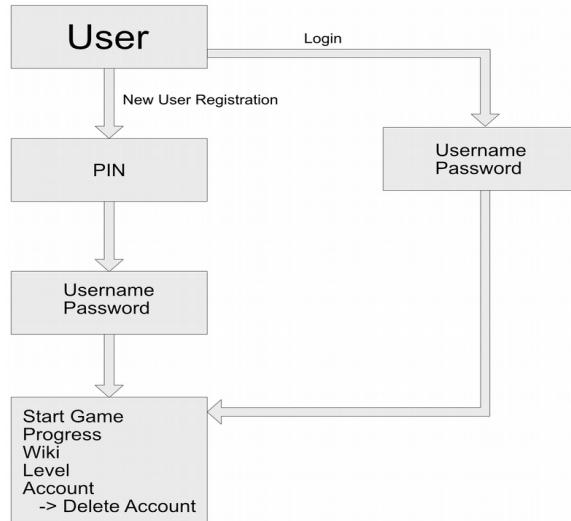


Abbildung 9: Flussdiagramm Account erstellen

### 3. Frontend-Seite

Auf der User-Frontend-Seite kann der Benutzer ein Spiel Starten bzw. ein vorhandenes Spiel fortfahren, seinen Fortschritt anschauen oder in den Wikis etwas über die Aufgabenthemen nachlesen. Im Fortschritt<sup>4</sup> sieht der Benutzer seinen aktuellen Level<sup>5</sup> und alle Aufgabenthemen.

Neben den Aufgabenthemen steht zusätzlich noch, ob das jeweilige Thema bereits gelöst, nicht vollständig gelöst oder noch nicht angefangen wurde.

Beim Spielstart kann der Benutzer einen Charakter<sup>6</sup> auswählen. Die Charaktere können vom Benutzer noch farblich angepasst werden.

<sup>4</sup> Siehe Wireframes

<sup>5</sup> Wie im 1. Regelsatz beschrieben

<sup>6</sup> Die einzelnen Charaktere werden im Screendesign vorgestellt

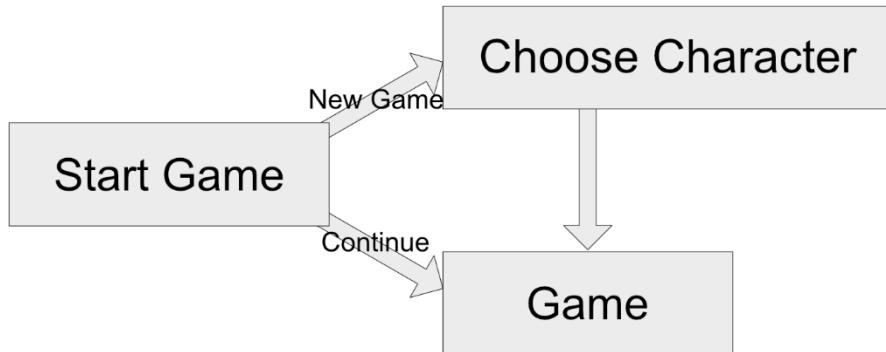


Abbildung 10: Flussdiagramm Spielstart

## Admin-Frontend und Admin-Backend

Im Admin-Frontend kann sich ein Lehrer einloggen oder neu registrieren. Nach dem Login kann auch er auf verschiedene Funktionen zugreifen, die aber über die des normalen Nutzers hinaus gehen.

Im folgenden werden die Schritte erläutert, die ein Lehrer nach Zugriff auf die Website durchführen kann:

### 1. Login

Wenn ein Lehrer bereits einen Account hat, kann er sich mit seinem Benutzernamen<sup>7</sup> und Password einloggen.

### 2. Account erstellen

Wenn ein Lehrer keinen gültigen Account hat und sich registrieren möchte, muss er seinen vollständigen Namen, Benutzernamen, seine E-Mail-Adresse und ein Password eingeben.

---

<sup>7</sup> Der Benutzername ist nicht sein echter Name

### 3. Admin-Frontend-Seite

Nachdem sich ein Lehrer eingeloggt hat kann er eine Session<sup>8</sup> erstellen. Dafür muss er einen Session-Namen vergeben, Themen auswählen und die Anzahl der Aufgaben festlegen. Die Aufgaben werden durch einen Aufgabengenerator generiert<sup>9</sup>. Nachdem eine Session erstellt wurde, wird diese durch einen Token zugänglich gemacht. Mit diesem Token können sich Benutzer in dieser Session einschreiben. Der Zugang zu einer Session kann von dem Ersteller abgeschaltet werden. Der Lehrer kann auch zwischen bereits erstellten Sessions wechseln.

In einer von ihm erstellten Session kann der Lehrer eine Benutzerliste sehen. Bei einer Benutzerliste kann der Lehrer Benutzer löschen oder von Benutzern verloren gegangene Passwörter auf das „Default-Password“ (=0000) zurücksetzen.

Der Lehrer hat auch die Möglichkeit in die Statistik der Session einzusehen. In der Statistik sieht er die Anzahl der Spieler sowie den Bearbeitungsfortschritt eines Benutzers bei jedem Thema (in %).

Im Admin-Backend können sich die Administratoren der Website einloggen um die Lehrer zu verwalten. Sie können sich eine Liste der Lehrer ausgeben lassen, die aktuell registriert sind und entweder neue Lehrer hinzufügen oder vorhandene löschen.

---

8 Siehe Kapitel Session

9 Siehe Kapitel Aufgaben

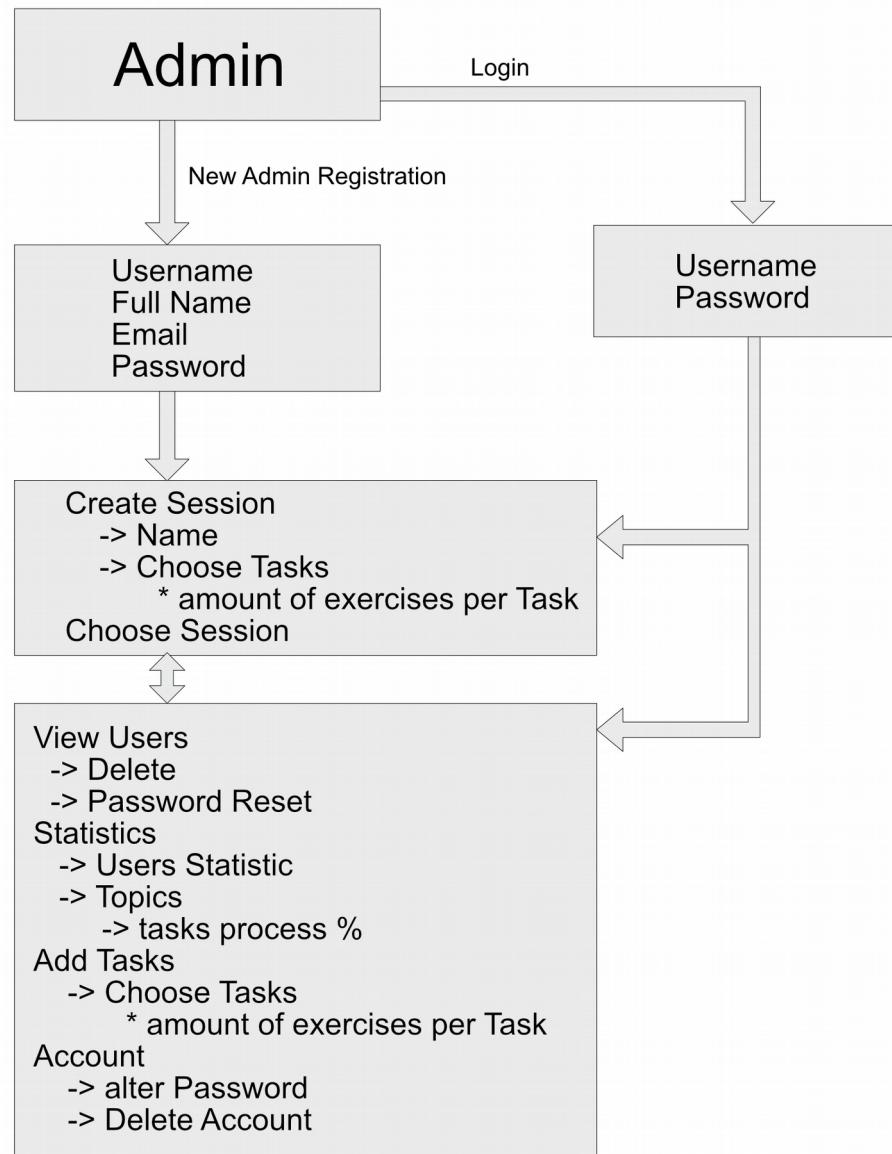


Abbildung 11: Flussdiagramm Administrator-Frontend

## Session

Eine Session ist eine Sammlung von Aufgaben und Themenbereichen, die durch einen eindeutigen Token vor dem Zugang von unberechtigten Benutzern geschützt ist. Der Administrator kann diesen Token an die Nutzer weitergeben, die an der Session teilhaben sollen. Die Benutzer können sich dann damit in der Session einschreiben.

Die Session bietet dem Administrator die Möglichkeit, den Umfang und Schwierigkeitsgrad der Themengebiete für die Zielgruppe festzulegen und gegebenenfalls zu variieren. So kann zum Beispiel ein Lehrer eine Session nutzen, um den Schülern der Klasse ein einheitliches und an den Lernfortschritt angepasstes Lernerlebnis durch das kollaborative Mathe-Spiel zu bieten.

## Screendesign

In diesem Kapitel werden die erarbeiteten Elemente der kollaborativen Plattform erläutert. Diese wurden bildlich festgehalten und können so besser veranschaulicht werden. Die ersten Designs bauen auf der kollaborativen Software von HumHub auf.

## Aufgaben

Die Aufgabenseite für das User-Frontend wird wie folgt aufgebaut:

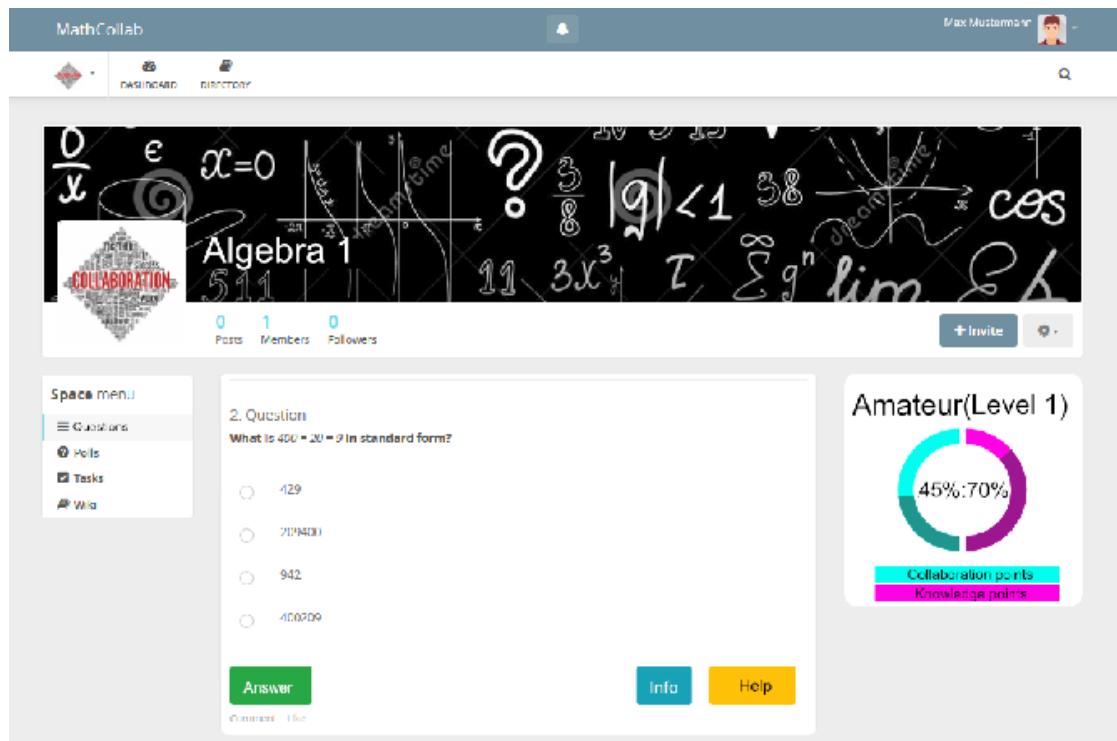


Abbildung 12: Wireframes (Aufgaben)

Wie auf der Abbildung zu sehen ist, wird dem Nutzer die Aufgabe nicht isoliert, sondern im Umfeld von anderen Elementen, wie einer Statusanzeige und der Navigation angezeigt.

In dem Fragenelement, welches zentral auf dem Bildschirm angezeigt wird, ist als Überschrift die eigentliche Aufgabenstellung zu sehen, darunter wird entweder eine Auswahl an Antworten oder ein Eingabefeld zu sehen sein. Am Ende des Elementes sind 2 Gruppen von Buttons zu sehen, die linke besteht aus dem „Next“-Button, welcher beim betätigen die Antwort des Nutzer evaluiert und dann sofortiges Feedback gibt. Unter dem Button erscheint ein Feld, welches entweder „Richtig“ oder „Falsch“ ausgibt. Die gewählte Antwort des Nutzers wird dann auch noch einmal markiert.

Die Button-Gruppe rechts unten wird die Funktionen der künstlichen Intelligenz auslösen. Der Info Button gibt dem Nutzer eine Auswahl an Links, die zu dem Thema verwandten Wikis führen. Der Hilfe Button hingegen gibt dem Nutzer die Möglichkeit, entweder die Hilfe der

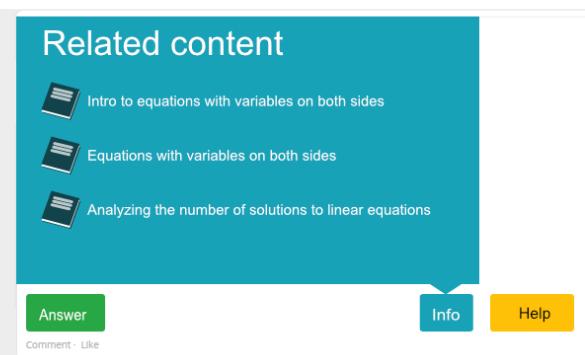


Abbildung 13: Wireframes (Info)

künstlichen Intelligenz durch Nutzung eines Tipps in Anspruch zu nehmen, oder aber,

durch einen Hilferuf, sein Problem in Form einer Frage an die Allgemeinheit zu richten. Falls der Nutzer einen Tipp möchte, so wird er aufgefordert, diesen quasi zu bezahlen, wie es in den Regelsätzen beschrieben ist.

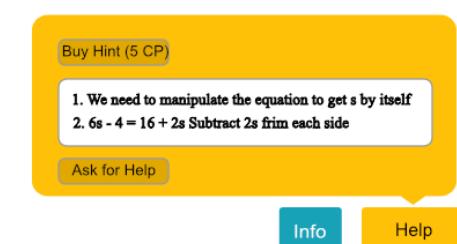


Abbildung 14: Wireframes (Hilfe Button)

Buttons die nächste Antwort angezeigt.

Wenn der Nutzer seine Antwort abgegeben hat, dann wird im nach Bestätigung des „Next“-

## Fortschrittsanzeige

Dem Nutzer wird im User-Frontend zu jederzeit sein Spiel- und Lernfortschritt angezeigt.

Oben in diesem Element wird das aktuelle Niveau des Nutzers und das Level, in dem sich der Nutzer befindet, angezeigt. Das Niveau kann, wie in den Regelsätzen beschrieben, entweder Amateur, Intermediate, oder Pro sein.

### Amateur(Level 1)

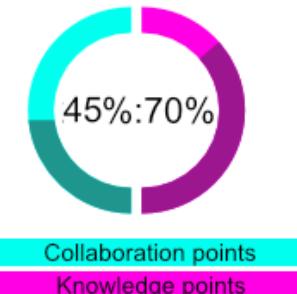


Abbildung 15: Wireframes

Das Diagramm unter dem Level zeigt in der Mitte das(Fortschritt)

Verhältnis der Punkte an, welches der Nutzer zum nächsten Level benötigt. Die Halbkreise um das Verhältnis zeigen den Fortschritt des Nutzers beim Sammeln der Punkte an. Ganz unten im Element findet sich noch eine Legende, welche die Farben der Art von Punkten in der Fortschrittsanzeige zuordnet, dies kann (siehe Regelsätze) entweder Kollaboration oder Wissen sein.

## Frage

Wählt ein Nutzer bei einer Aufgabe oder im allgemeinen den Punkt „Frage stellen“, so wird ihm dieses Element angezeigt. Falls, durch die Auswahl in einer Aufgabe, der Zusammenhang klar ist, so ist das Thema der Frage schon eingetragen. Der Nutzer hat aber die Möglichkeit, dieses zu verändern, falls es die Frage nicht genau genug beschreibt. Dies muss der Nutzer auch tun, falls der Zusammenhang nicht eindeutig ist.

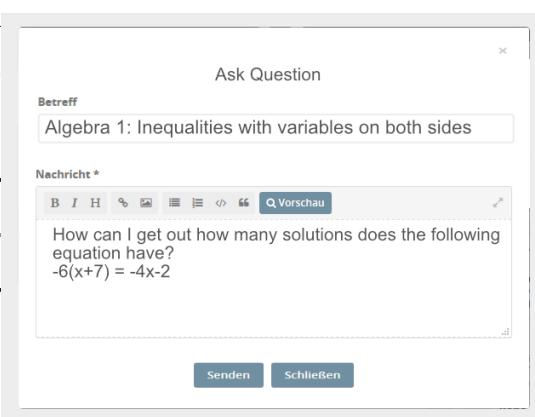
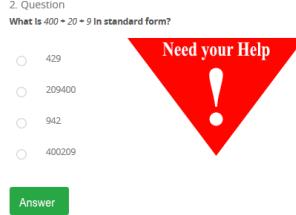


Abbildung 16: Wireframes (Frage)

Im zweiten Textfeld hat der Nutzer dann die Möglichkeit, seine eigentliche Frage zu stellen. Diese sendet er dann durch einen bestätigenden Button ab. Danach wird die Frage in der Reihenfolge, wie sie in den Regelsätzen beschrieben ist, angezeigt. Jeder Nutzer, der gerade online ist, bekommt dann auf seinem Bildschirm unübersehbar eine Nachricht gesendet, dass gerade eine Frage gestellt wurde.



*Abbildung 17: Wireframes (Hilfe benötigt)*

## Antworten

Sieht ein Nutzer, dass eine Frage gestellt wurde, so hat er die Möglichkeit, diese zum gegebenen Zeitpunkt, je nach persönlichem Level (siehe Regelsätze), auch zu beantworten. Hierfür bekommt der Anwender dieses Element angezeigt.

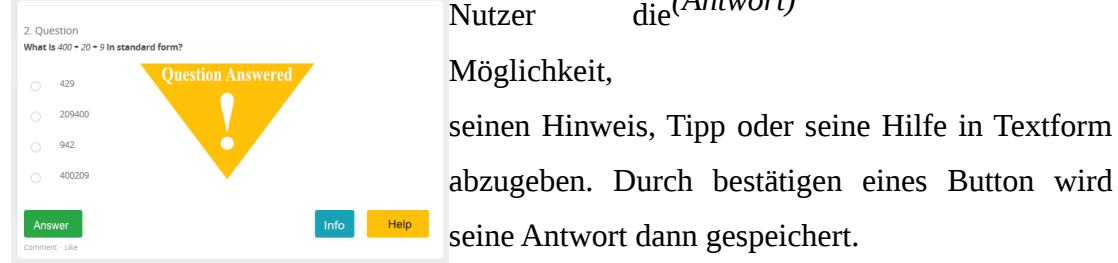
Oben im Element steht die Kategorie und die gestellte Frage. Im Textfeld darunter bekommt der

*Answer Question*



Nachricht \*

Senden Schließen



*Abbildung 19: Wireframes (Frage beantwortet)*

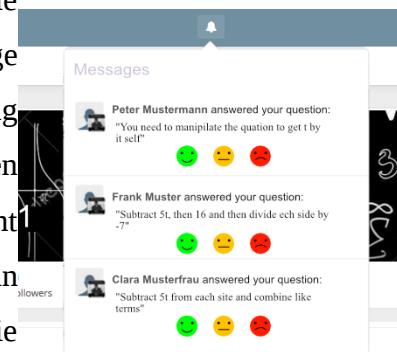
Nutzer die *(Antwort)*

Möglichkeit, seinen Hinweis, Tipp oder seine Hilfe in Textform abzugeben. Durch bestätigen eines Button wird seine Antwort dann gespeichert.

Sobald dies geschehen ist, kann derjenige, der die Frage gestellt hat, dies auf seinem Bildschirm sehen, insofern er noch online ist.

Sollte das nicht der Fall sein, so kann er beim nächsten Login in seinen Benachrichtigungen nachsehen, ob seine Fragen beantwortet wurden. Diese wird durch eine kleine Glocke am oberen Rand des Bildschirms symbolisiert. Leuchtet dort eine rote Zahl auf, so zeigt diese die Anzahl der nicht gelesenen Benachrichtigungen des Nutzers an.

Klickt der Nutzer auf diese Glocke, kann er die Benachrichtigungen einsehen. Wurde eine Frage beantwortet, kann er die zugehörige Benachrichtigung anklicken, worauf sich ein Element mit den drei ersten Antworten zu der Frage öffnet. Zu jeder Antwort steht auch der Verfasser der Hilfestellung. Der Nutzer hat nun die Möglichkeit, die Hilfe zu bewerten, damit die Helfenden ihre Kollaborationspunkte dementsprechend erhalten (siehe Regelsätze). Die Bewertung geschieht durch drei Smileys, welche den Zufriedenheitsgrad des Nutzers mit der Antwort widerspiegeln. Die Smileys können einfach angeklickt werden, worauf die Reaktion des Nutzers auf die Antwort gespeichert wird.



<sup>9</sup> In standard form?

*Abbildung 20: Wireframes (Benachrichtigung)*

## Einbindung der Corporate Identity des Kunden

Im folgenden Kapitel werden die Möglichkeiten betrachtet, die Corporate Identity eines Lehrinstituts, dass die Website nutzt, in die Weboberfläche einzubinden.

Einige Schulen würden den sicherlich den Vorteil genießen, ihr eigenes Design in eine Weboberfläche zu integrieren, die sie in ihrem Unterricht nutzen. Folgende Möglichkeiten haben sich im Rahmen einer Evaluation des Teams als sinnvoll ergeben:

- **Hochladen des Logos des Instituts**

Es soll im Lehrer Frontend möglich sein, ein Logo hochzuladen, welches dann im Userfrontend links oben im Bild angezeigt werden soll. Dies kann auch als Icon neben dem Titel der Website im Tab-Reiter sein.

- **Konfigurieren einer Standard-Grundfarbe**

Im Lehrer-Frontend sollte eine Grundfarbe und eine sekundäre Farbe auswählbar sein. Diese müssen dann lediglich in der CSS-Datei der Weboberfläche geändert werden, damit die Schüler die neuen Farben im Userfrontend sehen können.

- **Einfügen eines Impressums des Instituts**

Als letzter Schritt kann im Lehrer Frontend noch ein Impressum oder zumindest Kontaktdaten der Lehrkraft eingegeben werden. So können die Schüler schnell sehen, an wen sie sich bei Verständnis-Problemen wenden können, sobald die kollaborative Hilfe des Spiels nicht mehr weiter hilft.

**Anmerkung:** Diese Funktionen sind im Rahmen eines fertigen Produktes sinnvoll.

Jedoch werden sie nicht im Prototyp umgesetzt, da hierzu die Zeit fehlt und diese Funktionen eine relativ geringe Priorität haben.

## User Stories

Im folgenden Kapitel werden einige User-Stories aufgelistet. Sie sollen den Ablauf der Hauptfunktionen im Spiel erläutern und ein erstes Bild darüber schaffen, wie der fertige Prototyp aufgebaut sein wird. Da die Userstories jedoch vor der Implementation erstellt werden, können diverse Abweichungen von den Anforderungen auftreten.

### 1. Registrierung Schüler

Ein Schüler registriert sich, um mit seinen Klassenkameraden an den Lernspiel teilnehmen zu können, um sein Schulwissen spielerisch zu verbessern.

#### Akzeptanzkriterien:

1. Das System nimmt einen Token entgegen und überprüft diesen, sobald der „check token“-Button gedrückt wurde.
  - a) Falls der Token gültig ist wird mit der Registrierung fortgefahrene.
  - b) Falls der Token gültig ist, wird dem Schüler die Fehlermeldung “your token does not exist, please check it“ ausgegeben.
2. Bei korrektem Token erscheint das Feld für die Eingabe von Passwort, Nutzernamen und Charakterauswahl, sowie der “check-in“ Button.
3. Der Schüler gibt Passwort und Nutzernname ein, sucht den Charakter aus und klickt den “check-in“-Button.
4. Das System überprüft die Daten.
  - a) Falls der Schüler nicht alle Felder ausfüllt, wird die Meldung „please fill in all fields“ ausgegeben.
  - b) Falls der Schüler alle Felder ausgefüllt hat, öffnet das System das Spiel, also die „home“ Seite.

## **2. US Login Lehrer**

Ein Lehrer möchte sich einloggen, um neue Klassenzimmer und die zugehörigen Schüler und Aufgaben für diese zu erstellen oder diese zu beobachten.

Die Eingabe besteht aus dem Nutzernamen und Passwort des Nutzers.

### **Akzeptanzkriterien**

- 1) Nachdem der “Log-in-Teacher“-Button gedrückt wurde überprüft das System, ob der eingegebene Nutzernname bereits registriert ist.
  - a) Falls der eingegebene Nutzername nicht registriert und somit nicht existent ist, wird die Meldung “username does not exist“ ausgegeben.
  - b) Falls der Nutzername existiert, wird mit der Verarbeitung fortgefahrene.
- 2) Das System überprüft, ob das eingegebene Passwort mit dem Nutzernamen übereinstimmt.
  - a) Falls das Passwort nicht mit dem Nutzernamen übereinstimmt, gibt das System hierfür die Fehlermeldung “incorrect password“ aus.
  - b) Sind beide Eingaben existent bzw. korrekt und der Login somit erfolgreich, kann der Nutzer nun fortfahren.
- 3) Das System stellt der Webseite die notwendigen Daten bereit, damit der Lehrer seine Klassenräume verwalten und sich über die Fortschritte seiner Schüler informieren kann. Anschließend öffnet es die „Home“ Seite“.

### **3. US Login Schüler**

Der Schüler will sich in das Spiel einloggen, um zu lernen und sein Wissen zu erweitern. Die Eingabe besteht aus dem Nutzernamen und Passwort des Nutzers.

#### **Akzeptanzkriterien**

- 1) Nachdem der “Log-in-Student“-Button gedrückt wurde überprüft das System, ob der eingegebene Nutzernname bereits registriert ist.
  - a) Falls der eingegebene Nutzername nicht registriert und somit nicht existent ist, wird die Meldung “username does not exist“ ausgegeben.
  - b) Falls der Nutzername existiert, wird mit der Verarbeitung fortgefahren.
- 2) Das System überprüft, ob das eingegebene Passwort mit dem Nutzernamen übereinstimmt.
  - c) Falls das Passwort nicht mit dem Nutzernamen übereinstimmt, gibt das System hierfür die Fehlermeldung “incorrect password“ aus.
  - d) Sind beide Eingaben existent bzw. korrekt und der Login somit erfolgreich, kann der Nutzer nun fortfahren.
- 3) Das System bereitet das Spiel auf den individuellen Stand des nun eingeloggten Schülers vor und öffnet dieses.

## **4. US Schüler beantwortet Aufgabe (Multiple Choice)**

Der Spieler wählt eine oder mehrere Antworten aus, um Wissenspunkte zu sammeln.  
(siehe Abbildung 12)

### **Akzeptanzkriterien**

1. Das System prüft, wie viele richtige Antwortmöglichkeiten bei der beantworteten Frage benötigt sind und merkt sich diese, sowie die konkret angekreuzten Kästchen.
2. Der Spieler kreuzt seine Antworten an und drückt den “next-“Button.
3. Das System vergleicht die vom Nutzer angekreuzten Antworten mit den tatsächlich richtigen Antwortmöglichkeiten und merkt sich die vom Nutzer in der Aufgabe erreichte Punktzahl.
4. Das System markiert die richtigen und falschen Antworten und der User bekommt verschiedene Meldungen ausgegeben.
  - a) Unter den „Check-boxen“:
    1. Bei einer richtigen Antwort die Meldung “Correct (Selected)“.
    2. Bei einer falschen Antwort die Meldung “Incorrect“.
    3. Bei der nicht angeklickter Antwort die Meldung “Correct“.
  - b) Unter dem “Next“-Button:
    1. Bei richtig beantworteter Frage die Meldung “You got it“.
    2. Bei nicht richtig beantworteter Frage die Meldung “Sorry, Try again later“.
5. Das System addiert die sich gemerkte Punktzahl auf die bereits bisher erreichten Wissenspunkte des Spielers, speichert diese und aktualisiert die Anzeige.

## **5. US Schüler beantwortet Aufgabe (Frage-Antwort-Aufgabe)**

Der Spieler antwortet auf eine Frage-Antwort-Aufgabe mit einem Text, um Wissenspunkte zu sammeln. (siehe Abbildung 16)

### **Akzeptanzkriterien**

1. Der Schüler gibt seine Antwort ein und drückt den „send“-Button.
2. Der eingegebene Text wird vom System eingelesen.
3. Das System wertet die vom User eingegebene Antwort aus und merkt sich die erworbenen Punktzahl, je nachdem wie fehlerlastig die Antwort ist.
4. Der User bekommt eine Meldung ausgegeben.
  - a) Bei richtig beantworteter Frage die Meldung “You got it”.
  - b) Bei nicht richtig beantworteter Frage die Meldung “Sorry, Try again later”.
  - c) Bei nicht komplett richtig beantworteter Frage die Meldung “you got it almost right”.
5. Das System addiert die sich gemerkte Punktzahl auf die bereits bisher erreichten Wissenspunkte des Spielers, speichert diese und aktualisiert die Anzeige.

## **6. US Schüler braucht Hilfe**

Der Spieler kommt bei einer Aufgabe nicht weiter und betätigt deswegen den “Help“-Button.

### **Akzeptanzkriterien**

Das System öffnet ein kleines Fenster, in welchem der User sich einen Tipp kaufen oder bei seinen Mitschülern um Hilfe fragen kann (siehe Abbildung 14).

1. Der Spieler betätigt den “Buy Hint“ Button:
  - a) Das System subtrahiert von der Gesamtpunktzahl der Kollaborationspunkte des Spielers eine bestimmte Anzahl von Punkten zur Bezahlung des Tipps.
  - b) Das System sucht mit der KI eine Hilfestellung passend zur Frage und zeigt diese dem Spieler in dem kleinen Fenster an.
  
2. Der Spieler betätigt den “Ask for Help“ Button
  - a) Das kleine Fenster schließt sich und der Spieler kommt auf eine neue Seite.
  - b) Der Spieler stellt seine Frage und das System stellt Hilfe bereit.
    1. Falls ein anderer Mitspieler innerhalb von 50 Sekunden antwortet, bekommt der User eine Mitteilung “Question Answered“ (siehe Abbildung 19) und es wird an der Benachrichtigungsglocke die Anzahl der beantworteten Fragen angezeigt, wenn der Spieler dies anklickt, kann die Antworten der Mitspieler bewerten (siehe Abbildung 20).
    2. Falls kein Spieler antwortet, greift die KI ein und wählt eine passende Antwort zu der Frage aus.

## **7. US Schüler nutzt das Wiki**

Der Schüler bemerkt, dass sein Wissensstand bei einer bestimmten Frage nicht ausreicht. Er möchte aber nicht direkt um Hilfe beten, sondern sich mithilfe eines Wikis weiter informieren, um so die Aufgabe zu lösen.

### **Akzeptanzkriterien:**

1. Der Anwender betätigt den “Info-Button“ (siehe Abbildung 13).
2. Das System öffnet ein kleines Fenster mit kurzen Beschreibungen der Wiki-Einträge, des passenden Themas.
3. Der Anwender klickt eine dieser Möglichkeiten an.
4. Das kleine Fenster schließt sich und das System stellt den Artikel bereitet.

## **8. US Schüler hilft Mitspieler (Kollaboration)**

Der Anwender hilft einem Mitspieler eine Aufgabe zu beantworten, um Kollaborationspunkte zu sammeln.

### **Akzeptanzkriterien**

1. Der Anwender bekommt einen Hinweis, dass ein Mitspieler Hilfe braucht durch die Notiz “Need your Help” (siehe Abbildung 17).
  1. Falls der Anwender auf den Button klickt
    1. Der Schüler wird auf eine andere Seite verwiesen.
    2. Der Schüler gibt seine Antwort auf die Frage ein und bestätigt diese mit dem „ok“-Button.
    3. Der Schüler gelangt zu seinem vorherigen Spielstand.
    4. Das System verteilt Kollaborationspunkte an die am besten bewerteten Plätze und addiert diese somit auf die Kollaborationskonten der Beantworter und aktualisiert die Anzeige.
  2. Falls der Schüler woanders als auf den Button klickt, schließt sich nur der Button und er kann fortfahren.

## **9. US Lehrer erstellt ein Klassenzimmer**

Der Lehrer möchte einer neuen Klasse die Möglichkeit geben am Spiel teilzunehmen.

### **Akzeptanzkriterien**

1. Der Lehrer klickt den “create Session“-Button im Menü, wodurch sich ein neues Fenster öffnet.
2. Der Lehrer benennt die Session, sucht sich aus, wie viele Aufgaben es geben soll und gibt einen Rahmen für die Schwierigkeit der Level an, anschließend drückt er den “ok“-Button.
3. Das System überprüft, ob alle Felder ausgefüllt sind.
  - a) Falls nicht alle Felder ausgefüllt worden sind wird die Meldung “please fill in all fields“ ausgegeben.
  - b) Falls alle Felder ausgefüllt wurden, werden die Daten gespeichert und der Lehrer gelangt zurück zum vorherigen Bildschirm.

## **10. US Lehrer möchte Statistiken sehen**

Der Lehrer ist daran interessiert, wie gut sich seine Schüler schlagen, um so seinen Unterricht an die Schwierigkeiten dieser anzupassen.

### **Akzeptanzkriterien**

Der Lehrer verwendet den “Statistics“ Button um ins Statistics Menü zu gelangen.

1. Der Lehrer betätigt den “Topics“ Button, dadurch generiert das System eine Statistik, über die korrekt beantworteten Fragen der Schüler zu einem einzelnen Thema.
2. Der Lehrer betätigt den “Users Statistic“ Button, um die Leistung eines einzelnen Schülers beobachten zu können.

## **11. US Lehrer verwaltet Schülerkonten**

Der Lehrer möchte die Konten seine Schützlinge verwalten, um bei einen vergessenen Passwort dem Schüler wieder die Möglichkeit zu geben weiter zu spielen oder inaktive Accounts zu löschen.

### **Akzeptanzkriterien**

Der Lehrer betätigt den “view Users“ -Button, um ins Verwaltungsmenü zu kommen.

1. Falls der Lehrer nun den “delete“-Button auf einen Schüler anwendet, wird der Account dieses Mitspielers vom System gelöscht.
2. Falls der Lehrer den “password reset“ Button verwendet, hat der Schüler beim nächsten Login-Versuch die Möglichkeit ein neues Passwort einzugeben. Das neue Passwort wird vom System gespeichert und das alte gelöscht.

## **12. US Logout**

Der Spieler möchte sich aus dem Spiel ausloggen, nachdem er sein Tagesziel erreicht hat.

### **Akzeptanzkriterien**

1. Der Anwender betätigt die Logout-Funktion.
2. Das System zeigt dem Spieler eine Logout-Nachricht “logout successful“ an und verabschiedet sich somit ihn somit.

## Einbinden des Spiels

Um den Lernfaktor für Kinder zu vergrößern und vor Allem, um den Spaß an der Mathematik in ihnen zu wecken, beinhaltet das Projekt neben seinem kollaborativen Teil einen spielerischen Teil. Im folgenden wird erläutert, welche Möglichkeiten zu Auswahl standen und für welche sich entschieden wurde.

## Auswahlmöglichkeiten

Die amerikanische Partnerschule, Penn-State-University, hat durch ihre Studenten und unter ihrem Dozenten Joe Oaks eine Liste an möglichen Spielideen entwickelt, welche sich ganz besonders dafür eignen, in eine Umgebung eingebunden zu werden, in der mathematische Grundlagen vermittelt werden. Die folgende Liste bietet einen groben Überblick<sup>10</sup> über die Spielideen und ihrer Vor- und Nachteile.

- **Nexus**

Das Design des Spiels ist eine „Cyber-Punk-World“, in dem der Hauptdarsteller, Harold Chase, für den Diebstahl von Firmen internen Informationen verhaftet wird.. Die Storyline verfolgt Harold Chase, während er versucht durch mathematische Rätsel herauszufinden, wer ihn hinter Gitter gebracht hat.

Pro	Contra
Einfache Einbindung mathematischer Aufgaben	Aufwändige Storyline, die noch erstellt werden muss. Ein Rätsel zu erstellen ist oft sehr aufwendig und muss gut durchdacht werden
	Eventuell hohe Komplexität der Story, die jüngere Schüler auch überfordern kann

<sup>10</sup> Die Überblicke sind Übersetzungen aus den original Dokumenten, welche im Anhang als Quellen mitgegeben werden.

- **Run Craig Run**

Das Spiel basiert auf dem Charakter Craig, der ein Hürdenläufer in seiner Universitäts-Mannschaft ist. Craig ist sehr begabt auf seinem sportlichen Gebiet, jedoch weniger in Mathe, weshalb seine Noten schlechter werden. Er entscheidet sich also, die Mathematik mit dem Hürdenlauf zu verbinden, um nicht aus der Mannschaft zu fliegen und findet langsam Spaß daran. Mit jeder Aufgabe, die Craig löst, läuft er schneller, springt höher und erzielt bessere Ergebnisse in Wettkämpfen.

Pro	Contra
Einfache Story	Kein klarer Zusammenhang zwischen Mathe und Hürdenlauf erkennbar, scheint sehr unwahrscheinlich
Einfache Schwierigkeits-Steigerung bei wachsendem Wettkampfdruck des Hürdenläufers	Story ist wenig ansprechend für die Breite der Zielgruppe, Hürdenlauf ist ein Thema, für das sich nicht viele Kinder begeistern können

- **Math Life**

Eine Story ist in diesem Spiel nicht vorhanden. Es wird jedem Spieler ein Charakter zugewiesen, welcher einen Job hat, in dem er Mini-Spiele lösen muss, um Geld zu verdienen. Diese Minispiele werden mathematische Rätsel sein. Mit dem verdienten Geld kann man den Lifestyle seines Charakters verbessern und somit vor anderen Mitspielern angeben.

Pro	Contra
Einfache Einbindung mathematischer Aufgaben	Zielloses Konzept, nach hinten offen
Hohe Modularität, einfaches Einfügen neuer Funktionen und Minispiele, da kein echter Zusammenhang in Form einer Story besteht	Charaktere verändern gefällt nicht jedem und zählt nicht zu den spannendsten Geschichten, die die Mathematik interessanter machen

	Fokus soll auf die Mathematik gelegt werden, nicht auf den Aufbau eines sozialen Aushängeschildes
--	---

- **Algebra Pirates**

Die Story basiert auf einer Piraten Crew, die versucht einen Schatz zu finden, welcher aber nur über eine Karte gefunden werden kann, die mit mathematischen Rätseln zu entschlüsseln ist. Die Spieler lösen ein Rätsel nach dem anderen und finden so immer neue Hinweise auf neuen Inseln. Die Inseln stellen neue Themengebiete dar.

Pro	Contra
Einfache Einbindung mathematischer Aufgaben durch Rätsel in der Karte	Schwieriger Ansatz, das Spiel in die kollaborative Umgebung zu integrieren
Einfache Anbindung neuer Themenbereiche durch hinzufügen einer neuen Insel	
Klare Struktur in der Idee erkennbar	
Interessantes Spielthema für eine breite Masse des Zielpublikums	

- **Math Royale**

In dem Spiel geht es um Action, man ist ein Charakter, der versucht, Gegner im Kampf zu besiegen, indem er mathematische Aufgaben löst. Mehrere richtige Aufgaben hintereinander bringen dem Gegner mehr Schaden. Die Gegner besitzen Namen aus der Mathematik, z.B. „Unit Circle Wizard“ oder „Square Root Warrior.“ Besiegt man den Gegner, so hat man einen Themenbereich erfolgreich abgeschlossen.

Pro	Contra
Einfache Einbindung mathematischer Aufgaben, ein Schlag erfordert eine richtige Antwort	Actionreiches Spiel, was eventuell Gewaltverherrlichend interpretiert werden kann
Hohe Modularität, neue Themenbereiche werden durch neue Endgegner eingefügt	Schwer in das kollaborative Umfeld einzubinden
Einfaches Konzept, bis auf die Gegner keine große optische Designarbeit notwendig	
Actionreiche Story, begeistert ein breites Publikum im jungen Alter	

## Wahl des Spielkonzepts

Das Team hat sich nach Absprache für die Spielidee „Math Royale<sup>11</sup>“ entschieden. Grund hierfür war die einfache Umsetzung der Spielidee, man kann neben einer einfachen Multiple Choice Frage ein zweites Element anzeigen, welches die kämpfenden Charaktere anzeigt. Diese müssen sich nur bewegen, wenn sie einen Schlag austauschen, was nur dann passiert, wenn der Anwender eine Antwort bestätigt. Auch die gute Ausarbeitung mit vielen Details, die in einer finalen Version der Website angewendet werden können, ist ausschlaggebend für die Wahl. Im Gegensatz zu vielen anderen Spielideen war diese am präzisesten formuliert.

Eine Alternative bietet auch noch das Spiel „Algebra Pirates“, welches auch gut durchdacht ist und viele Möglichkeiten bildet, Mathematik einzubringen. Das Team hat sich aber gegen „Algebra Pirates“ entschieden, was aber eine völlig subjektive Entscheidung ist, da die beiden Ideen „Algebra Pirates“ und „Math Royale“ objektiv betrachtet die selben Vor- und Nachteile mit sich bringen.

Der große Nachteil an beiden Optionen ist die geringe Kollaboration in der Spielidee an sich. Deshalb wird der kollaborative Faktor in der Website getrennt vom Spiel

<sup>11</sup> Spielidee übernommen von Kyle Rimler, Kelby Chen, William Donovan, Emily Bonilla, Anthony Lee und Viktoria Fytsyk, Studenten der Penn-State-University, Abington, Pennsylvania

betrachtet, wer eine Frage hat oder Hilfe braucht, der muss dies außerhalb des Spiels klären. Die Kämpfe finden immer dann statt, wenn ein Spieler sich in der Lage fühlt, ein Thema zu beenden.

## **Serverseitige Funktionen**

In diesem Kapitel werden alle Funktionen des Servers beschrieben. Dies sind grundlegend alle Aufgaben, die für den Client zu rechnen aufwändig sind, daher die Website langsamer machen, aber auch die Aufgaben, die aus Gründen der Sicherheit nicht außerhalb des Servers erledigt werden dürfen.

### **Nutzer-Verifizierung**

Im Folgenden werden die Funktionen aufgelistet, welche der Server im Rahmen der Nutzerverwaltung übernimmt.

- Überprüfung der Daten bei Login-Versuch eines Administrators / Users

Der Server muss bei einem Login-Versuch die relevanten Daten, also Benutzername und Passwort, mit den in der Datenbank gespeicherten Daten abgleichen. Wichtig ist hier vor Allem, dass die Sicherheit bei der Speicherung und Verifizierung der Passwörter gewährleistet ist.
- Laden der Daten eines erfolgreich eingeloggten Administrators / Users

Ist der Datenabgleich beim Login erfolgreich, so müssen, je nach Navigation des Nutzers, verschiedene Inhalte aus der Datenbank geladen werden.
- Anlegen und Speichern eines neuen Administrators / Users

Legt ein Nutzer einen neuen Account ein, so müssen diese Daten in der Datenbank abgelegt werden. Der wichtigste Punkt ist hier die sichere Speicherung der Passwörter, was später beschrieben wird.

- Reset eines User-Passworts durch einen Administrator

Vergisst ein User sein Passwort, so kann der Administrator einer Session, an der der User teilnimmt, das vergessene Passwort auf das Default-Passwort zurücksetzen, damit der User sich wieder einloggen kann.

- Reset eines Administratoren-Passworts bei Verlust

Verliert oder vergisst eine Administrator sein Passwort, so muss er die Möglichkeit haben, dies zurückzusetzen, um sich wieder einzuloggen. Dieser Vorgang wird später beschrieben.

- Änderung eines Passworts eines Administrators / Users

Sind Nutzer eingeloggt, so sollen sie immer die Möglichkeit haben, ihr Passwort zu ändern.

## **Passwortsicherheit**

Grundlegend ist die Voraussetzung für ein sicheres Login der Nutzer, dass die Passwörter verschlüsselt verschickt und gespeichert werden. Viele Websites nutzen daher Hashing-Verfahren, wie z.B. „sha256“, welche für ihre gute Verschlüsselung und schnelle Algorithmen bekannt sind. Dies ist jedoch ein Sicherheitsfehler, da schnelle Algorithmen die Möglichkeit der Brute-Force Angriffe auf Passwörter ermöglichen. Deshalb nutzt der Prototyp den „bcrypt“-Algorithmus, gepaart mit einer Salt-Value, welche dem Passwort zugefügt wird, wenn es verschlüsselt wird. Die Salt-Value ist 21 stellig und wird zufällig generiert. Der große Vorteil von „bcrypt“ ist der variabel konfigurierbare Algorithmus, welcher je nach Setzung der Variablen schneller oder eben rechen aufwändiger wird. Somit wird ein Brute-Force Angriff unmöglich, da das ausprobieren eines Passwortes zu lange dauert, um alle

Möglichkeiten zu erschöpfen. Dennoch ist der Algorithmus nicht zu langsam, weshalb er den Login-Vorgang nicht unnötig verzögert.

Dieser Verschlüsselungsaufwand ist jedoch nur für Administratoren nötig, bei Usern ist der Zugang nicht zu sensibel, weshalb dort ein einfacherer Algorithmus völlig ausreicht.

Im Folgenden werden die Informationen nochmals gesammelt:

- Administrator Passwort-Handling
  - Passwort-Länge: mindestens 8 Zeichen
  - Passwort Constraints: mindestens 1 Großbuchstabe und 1 Kleinbuchstabe, mindestens ein Sonderzeichen oder eine Zahl
  - Hashing: Bcrypt-Algorithmus, Passwort wird vorher durch zufällig generiertem Salt abgeändert.
- User Passwort-Handling
  - Passwort-Länge: mindestens 8 Zeichen
  - Hashing: Sha256-Algorithmus
- Administrator Passwort zurücksetzen

Der Administrator gibt seinen vollen Namen an. Anhand dessen wird eine E-Mail an die hinterlegte Adresse geschickt, welche einen Link beinhaltet, der den Nutzer anmeldet, womit er sein Passwort ändern kann.

- User Passwort zurücksetzen

Der User navigiert zu dem Unterpunkt Passwort zurücksetzen, wo er durch bestätigen des aktuellen Passworts ein neues hinterlegen kann. Dieses ersetzt das alte Passwort in der Datenbank<sup>12</sup>.

---

<sup>12</sup> Die Passwort Sicherung wird wegen des hohen Zeitaufwands nicht implementiert. Stattdessen werden Yii Funktionalitäten genutzt

## Aufbau der Datenbank

Im folgenden Kapitel werden die Tabellenstrukturen der Datenbank des kollaborativen Spiels dargestellt und erläutert.

### ***Administrator-Tabelle***

Die Administrator Tabelle sammelt alle Nutzer, welche administrative Rechte an der Website besitzen. Dies schließt zum Beispiel die Entwickler und die Betreiber der Website ein. Administratoren können Lehrer verwalten.

#### **Struktur der Tabelle:**

```
CREATE TABLE `admin` (
    `id` int(7) NOT NULL AUTO_INCREMENT,
    `full_name` varchar(40) DEFAULT NULL,
    `username` varchar(20) NOT NULL,
    `password` varchar(60) NOT NULL,
    PRIMARY KEY (`id`),
    UNIQUE KEY `username` (`username`)
)
```

#### **Beispielinput:**

<b>id</b>	<b>full_name</b>	<b>username</b>	<b>password</b>
1	Dennis Brysiuk	Dbrysiuk	root

## ***Lehrer-Tabelle***

Die Lehrer Tabelle sammelt alle Sessionleiter. Diese können eine Session verwalten, die angemeldeten User verwalten und deren Statistiken lesen.

### **Struktur der Tabelle:**

```

CREATE TABLE `teacher` (
    `id` int(7) unsigned NOT NULL AUTO_INCREMENT,
    `full_name` varchar(40) COLLATE utf8mb4_bin NOT
    NULL,
    `username` varchar(20) COLLATE utf8mb4_bin NOT
    NULL,
    `email` varchar(254) COLLATE utf8mb4_bin NOT NULL,
    `password` varchar(60) COLLATE utf8mb4_bin NOT
    NULL,
    PRIMARY KEY (`id`),
    UNIQUE KEY `username` (`username`),
    UNIQUE KEY `email` (`email`),
    UNIQUE KEY `password` (`password`)
)

```

### **Beispielinput:**

<b>id</b>	<b>full_name</b>	<b>username</b>	<b>email</b>	<b>password</b>
1	Mr. Teacher	a_teacher	teacher@example.com	teacher

## **User-Tabelle**

Die User Tabelle sammelt alle angemeldeten User, die an einer Session teilnehmen, um sich kollaborativ in der Mathematik zu verbessern. User haben keine administrativen Rechte und können somit nur auf ihr eigenes Frontend zugreifen.

### **Struktur der Tabelle:**

```

CREATE TABLE `student` (
    `id` int(7) unsigned NOT NULL AUTO_INCREMENT,
    `username` varchar(20) NOT NULL,
    `password` varchar(60) NOT NULL,
    `level` varchar(10) NOT NULL,
    `knowledgePoints` int(11) DEFAULT NULL,
    `collaborationPoints` int(11) DEFAULT NULL,
    `equations` int(11) DEFAULT NULL,
    `inequalities` int(11) DEFAULT NULL,
    `functions` int(11) DEFAULT NULL,
    `linearEquations` int(11) DEFAULT NULL,
    `polynomial` int(11) DEFAULT NULL,
    `quadraticEquations` int(11) DEFAULT NULL,
    `session_id` char(255) DEFAULT NULL,
    PRIMARY KEY (`id`),
    UNIQUE KEY `username` (`username`),
    UNIQUE KEY `password` (`password`),
    KEY `session_id` (`session_id`),
    CONSTRAINT `student_ibfk_1` FOREIGN KEY
    (`session_id`) REFERENCES `session` (`token`)
)

```

### **Beispielinput:**

<b>id</b>	<b>user-name</b>	<b>pass-word</b>	<b>level</b>	<b>know-ledge-Points</b>	<b>collabor-ation-Points</b>	<b>equat-ions</b>	<b>in-equal-i-ties</b>	<b>functi-ons</b>	<b>linea-r-Equation-s</b>	<b>poly-nomi-al</b>	<b>quad-ratic-Equa-tions</b>	<b>sess-ion-id</b>
1	user	pswd	ama-teu-r	50	20	12	33	23	8	1	0	12

Beim User sind folgende Werte anzumerken:

- **session\_id:**

Ein User kann nur an einer Session teilnehmen, die ID dieser wird bei ihm in der Tabelle abgespeichert.

- **Level:**

Das Level eines Users kann entweder Amateur, Fortgeschritten oder Profi sein.<sup>13</sup>

- **knowledge/ collaborationPoints:**

Zeigen den Fortschritt eines Users innerhalb seines Levels an. Die passenden Verhältnisse, welche zum Voranschreiten in das nächste Level qualifizieren, sind in den Regelsätzen beschrieben<sup>14</sup>.

- **Equations/ inequalities/ functions/ linearEquations/ quadraticEquations/ polynomial**

Dienen zum Abgleich mit den in der Session-Tabelle hinterlegten Werten, welche benötigt werden, um ins nächste Level zu gelangen. Die Werte bilden die Anzahl der Aufgaben dar, die der User aus der entsprechenden Kategorie gelöst hat.

**Anmerkung:** Der User sollte in einem endgültigen Produkt die Möglichkeit haben, an mehreren Sessions teilzunehmen. Dies wird im Prototyp aufgrund des Aufwands allerdings nicht implementiert. Jeder User kann an genau einer Session teilnehmen, die Werte für die erledigten Aufgaben in dieser Session werden in den passenden Variablen (z.B. equations) gespeichert. Es wird dann abgeglichen, ob die bearbeiteten Aufgaben mit den zu bearbeitenden Aufgaben in der Session übereinstimmt. Der Vergleich ist abhängig vom Level des Users<sup>15</sup>.

13 Siehe Regelsätze

14 Siehe Regelsatz 1

15 Siehe Regelsatz 1

## **Session-Tabelle**

Die Session Tabelle speichert alle Sessions, in die sich User einschreiben können.  
 Außerdem verweist sie auf den Lehrer, der sie administriert.

### **Struktur der Tabelle:**

```

CREATE TABLE `session` (
  `token` char(10) NOT NULL,
  `name` varchar(30) NOT NULL,
  `admin_id` int(10) unsigned DEFAULT NULL,
  `equations` int(11) DEFAULT NULL,
  `inequalities` int(11) DEFAULT NULL,
  `functions` int(11) DEFAULT NULL,
  `linearEquations` int(11) DEFAULT NULL,
  `polynomial` int(11) DEFAULT NULL,
  `quadraticEquations` int(11) DEFAULT NULL,
  PRIMARY KEY (`token`),
  KEY `admin_id` (`admin_id`),
  CONSTRAINT `session_ibfk_1`
    FOREIGN KEY (`admin_id`) REFERENCES `teacher`
      (`id`)
)
  
```

### **Beispielinput:**

Token	Name	admin_id	Equation s	Inequalit ies	Func-tions	LinearE quations	Polynom ial	quadrati cEquatio ns
123456 7890	exampl eSessio n	1	50	50	50	0	0	0

Bei der Session sind folgende Werte anzumerken:

- **Token:**

Das Token wird automatisch generiert und ist sowohl der Primary Key der Tabelle, als auch der Schlüssel, mit dem sich User in einer Session einschreiben können.

- **admin\_id:**

Die admin\_id ist die ID des Lehrers, der die Session administriert. Somit kann eine Session nur von einem Lehrer verwaltet werden.

- **equations/ inequalities/ function/ linearEquations/polynomial/ quadratic Equations:**

Dies sind die Werte, die mit den entsprechenden Werten der User-Tabelle abgeglichen werden. Sie stellen dar, wie viele Aufgaben die User in einem Themenbereich abschließen müssen, um ins nächste Level zu kommen. Ist ein Wert mit Null oder 0 belegt, so wird der Themenbereich in der Session noch nicht genutzt.

## **Aufgaben-Tabelle**

Der Aufbau der Aufgaben wird im Kapitel Aufgaben näher betrachtet<sup>16</sup>.

---

<sup>16</sup> Siehe Kapitel „Aufbau und Speicherung der Aufgaben“

## Datenbank Inhalte laden

Im Verlauf der Website-Nutzung soll niemals eine ganze Tabelle geladen werden, außer es wird eine Übersicht über Nutzer gefordert. Deshalb werden nur relevante Daten geladen und das nur in der Menge, wie sie benötigt werden. Die folgenden Kapitel gehen genauer auf das Server Verhalten beim Laden von Inhalten ein.

## **Benutzer Anlegen**

Im Frontend wird dem Benutzer ein Formular angezeigt, in welches er die relevanten Daten einträgt. Beim Versenden der Daten wird das neue Passwort mit einer auf dem Client zufällig generierten Salt-Value gehasht, bei Administrator Registrationen mit dem Bcrypt-Algorithmus, bei Usern mit dem Sha256-Algorithmus ohne die Salt-Value. Die Daten werden darauf an den Server verschickt, wo sie der Server auf Sicherheitsfehler (wie z.B. Injection-Versuche) prüft und darauf abspeichert und dem Client dann die Bestätigung schickt.

Bei den Usern wird die Session-ID vorerst mit null belegt, die bearbeiteten Aufgabenbereiche mit der Default Value 0.

## **Login**

Beim Login wird ein Formular ausgefüllt, in dem die Login-Daten gefordert werden. Das Passwort wird auch hier beim Client gehasht und dann mit den anderen Daten an den Server geschickt. Der Server fragt nun die Daten in der Datenbank ab und gleicht sie dann mit den vom Client empfangenen Aufgaben ab.

## **Anlegen einer Session**

Ist ein Administrator angemeldet, kann er Sessions erstellen. Dabei gibt er einen Namen ein, worauf ihm auf dem Server ein Token aus der aktuellen Zeit generiert wird. Dieses sollte sich der Administrator notieren, um es den Nutzer zeigen zu können, da diese es für die Anmeldung in einer Session brauchen. Er kann es sich aber auch jederzeit in einer Übersicht anschauen. Zusätzlich muss er die Aufgabenbereiche auswählen, welche er in die Session integrieren möchte und dazu festlegen, wie viele Aufgaben von den Usern erledigt werden müssen, um in den nächsten Themenbereich („Level“) übergehen zu dürfen. Die Daten werden darauf per Submit-Button an den Server geschickt, wo sie auf dem Server nochmals geprüft werden (Injection-Versuch.) Die Session wird dann wir folgt angelegt.

- Der Name wird genau so gespeichert, wie ihn der Administrator eingegeben hat.
- Das Token wird generiert und abgespeichert, danach wird es an den Client gesendet.
- Ausgewählte Themenbereiche werden als Integer abgespeichert, der Integer vermerkt die zu bearbeitenden Aufgaben.
- Die nicht ausgewählten Themenbereiche werden mit dem Default Wert 0 gespeichert.
- Die Administrator-ID ist die des Administrators, der die Session anlegt.

Im laufenden Betrieb der Session kann der Administrator jederzeit Themenbereiche zur Session hinzufügen. Die mit den Default Werten belegten Themenbereiche werden bei der Auswahl einfach mit den Werten belegt, die der Administrator eingibt.

## **Anmelden eines Users in einer Session**

Meldet sich ein User an, so kann er ein Token eingeben, um sich in einer Session anzumelden. Ist das Token gültig, so werden ihm die Inhalte aus der Datenbank geladen. Der Server prüft beim Token, ob es überhaupt vorhanden ist und schreibt das Token als Fremdschlüssel in den Datenbankeintrag des Users, falls dies der Fall ist.

## **Übersicht über alle User**

Möchte der Administrator eine Übersicht über alle in die Session eingetragenen User sehen, so werden ihm folgende Daten aus der Datenbank geholt:

```
Select username, {equations,inequalities,functions,  
linearEquations,polynomial,quadraticEquations}  
//Nur Themenbereiche, die bearbeitet werden müssen  
From user  
Where session_id = [Token der Session]
```

- Nutzername
- Alle Themenbereich, die in der Session bearbeitet werden sollen

Zusätzlich wird zu jedem Nutzer ein Button angezeigt, welcher das Passwort automatisch auf das Default-Passwort setzt.

## **Bearbeiten von Aufgaben**

Ist ein Nutzer in eine Session eingetragen, so werden ihm alle Themenbereiche angezeigt, die er momentan bearbeiten darf. Der Server filtert die Themenbereiche der Session anhand der Fortschritte, die es aus den Nutzerdaten ausliest. Wählt ein

Nutzer einen Themenbereich aus und bearbeitet eine Übung oder einen Test, so werden ihm nicht bearbeitete Aufgaben geladen, immer 10 am Stück.

Die aufgaben werden alle in einer Tabelle gespeichert, unabhängig vom Themenbereich. Die Aufgaben werden nach den Themenbereichen sortiert, jeder Themenbereich hat 500 Aufgaben. Somit ist die ID der ersten Aufgabe eines Themenbereiches fest definiert. Der Nutzer bekommt nun immer 10 Aufgaben, beginnend mit der ersten ID des Themenbereiches plus die Anzahl der schon bearbeiteten Aufgaben. So muss ein Nutzer nie die gleiche Aufgabe zwei mal bearbeiten.

Aufgaben = **erste ID** in Themenbereich in Aufgaben + **bearbeitete** Aufgaben aus Themenbereich

## Aufbau und Speicherung der Aufgaben

Im folgenden Kapitel werden die Aufgabentypen aufgelistet, die Art der Fragestellung erläutert und das Vorgehen beim Erzeugen und Speichern der Aufgaben dargestellt.

### Aufgabentypen

Im Prototyp werden sechs Aufgabentypen gestellt werden. Das Wort Typ beschreibt in diesem Zusammenhang den mathematischen Hintergrund der Aufgabe, also den Bereich der Mathematik, aus dem die Aufgabe stammt.

Im folgenden werden die sechs Aufgabentypen der Schwierigkeitsstufe aufsteigend aufgelistet:

1. Gleichungen
2. Ungleichungen
3. Lineare Gleichungen
4. Funktionen
5. Polynome
6. Quadratische Terme

Jede der Aufgabentypen wird in verschiedenen Variationen Abgefragt, sodass der Anwender nicht immer die gleiche Aufgabe mit verschiedenen Werten lösen muss.

### Aufgabenarten

Im Prototypen des kollaborativen Mathe-Spiel werden 2 Arten von Aufgaben gestellt. Die erste wird eine einfache Frage-Antwort Aufgabe sein, die zweite Art eine Multiple-Choice Aufgabe mit nur einer richtigen Antwort.

Die Frage-Antwort Aufgabe wird einen Text haben, der als Aufgabenstellung dient, und ein Textfeld, welches eine Eingabe vom Anwender erwartet. Dieses Textfeld soll die Antwort des Anwenders entgegennehmen.

Die Multiple-Choice Aufgaben werden ähnlich aufgebaut. Auch sie werden einen Text haben, der als Aufgabenstellung dient. Zusätzlich werden sie aber kein Textfeld, sondern eine Liste aus vier Radio-Buttons haben, welche die möglichen Antworten enthalten. Multiple-Choice Aufgaben bestehen, wie oben erwähnt, nur aus einer richtigen Antwort, insgesamt werden dem Anwender aber vier mögliche Antworten geliefert.

## Generieren der Aufgaben

Bei der Generierung der Aufgaben wird nicht zwischen den Aufgabenarten unterschieden. Es liegt für jeden Aufgabentyp ein Aufgaben-Generator auf dem Server, welcher beim Start die gewünschte Anzahl an Aufgaben generiert und automatisch auf der Datenbank in der Tabelle „exercises“ abspeichert.

Die Generatoren brauchen zum Start zwei Parameter, die maximale Größe der Variablen in den Aufgaben und die Anzahl der zu generierenden Aufgaben. Als Ausgabe generieren sie einen String, der den Aufgabentyp beschreibt, einen Aufgabentext, die richtige Antwort und drei garantiert falsche Antworten. Diese werden dann in der Datenbank in den entsprechenden Spalten abgespeichert.

Anzumerken ist, dass die Generatoren die Aufgaben zufällig erstellen, die Variablen werden immer mit Zufallszahlen gefüllt. Es wird nicht überprüft, ob zwei oder mehrere Aufgaben identisch ist. Dies würde den Aufwand für einen Prototypen übersteigen. Auch wenn die Generierung identischer Aufgaben beim erhöhen der maximalen Zahlenwerten sehr unwahrscheinlich ist, sollte das Eliminieren redundanter Aufgaben bei einem fertigen Produkt noch vorgenommen werden.

## Speicherung der Aufgaben in der Datenbank

Wie oben beschrieben besitzt die Datenbank des Spieles eine Tabelle namens „exercises“, welche nur die Einträge für Aufgaben enthält. Wie oben beschrieben benötigt die Datenbank nur eine Tabelle für alle Aufgabentypen und auch die Aufgabenarten. Die Tabelle ist wie folgt aufgebaut:

```
exercises: {  
    id int(6) unsigned auto_increment primary key,  
    type varchar(20) not null,  
    question text not null,  
    answer varchar(20) not null,  
    wrong1 varchar(20) not null,  
    wrong2 varchar(20) not null,  
    wrong3 varchar(20) not null  
}
```

## Abrufen der Aufgaben aus der Datenbank

Beim Abrufen der Aufgaben aus der Datenbank muss man zwei Dinge beachten:

- **Die Art der Aufgabe muss beachtet werden**

Bei einer Frage-Antwort Aufgabe reicht es, wenn man die Werte „question“ und „answer“ abgefragt werden. Eine passende Query wäre dann:

```
select question, answer  
from exercises  
where type = “[gewünschter Typ]”
```

Bei einer Multiple-Choice Aufgabe müssen die Werte „question“, „answer“ und die drei „wrongX“-Spalten abgefragt werden. Eine passende Query wäre somit:

```

select question, answer, wrong1, wrong2, wrong3
from exercises
where type = “[gewünschter Typ]”

```

- **Der Typ der Aufgabe muss explizit mit abgefragt werden**

Möchte man nur einen Aufgabentyp, so muss der Typ der Aufgabe immer mit abgefragt werden. Es muss also zusätzlich die Abfrage **where type = “[gewünschter Typ]”** eingefügt werden.

Die Aufgabentypen können folgende Werte annehmen:

Aufgabentyp	Wert in Datenbank
Gleichungen	equations
Ungleichungen	inequalities
Lineare Gleichungen	linearEquations
Funktionen	functions
Polynome	polynomial
Quadratische Terme	quadraticFunctions

## Nachbearbeitung der Aufgaben

Im Folgenden wird erläutert, wie die vom Generator erzeugten Mathematik Aufgaben auf Korrektheit getestet werden und wie diese gewährleistet werden soll.

### **Wer führt die Überprüfung durch?**

Zunächst werden die Aufgaben vom Tester des Programmierteams ausgiebig getestet und abgeglichen. Des Weiteren können die Aufgaben jederzeit durch den Administrator im Admin-Frontend abgeändert werden. Dennoch soll sich für den späteren Betrieb ein Programm um diese Aufgabe kümmern.

### **Wie geht der Tester vor?**

Der Tester sollte vor Inbetriebnahme, aber auch währenddessen auf Fehler achten. Vorher, indem er darauf achtet, dass der Generator allgemein nur richtige Ausgaben ausspuckt und das Programm aufgrund eines Fehlers nicht abstürzt. Während des Spielbetriebs sollte er auf die Log-Datei achten, um mögliche Programmierfehler zu beheben.

### **Wie geht der Lehrer vor?**

Der Lehrer hat stets Zugriff auf die Aufgaben und kann somit sofort eingreifen, sobald ein Fehler auffällt bzw. auftritt. Dies geschieht alles im Admin-Frontend, in welchem er sowohl fehlerhafte Aufgaben abändern oder gegebenenfalls löschen kann. Dass neben den anderen Nachbearbeitungsroutinen zusätzlich der Lehrer jederzeit auf die Aufgaben zugreifen und diese bearbeiten kann, gewährleistet eine noch schnellere und sicherere Fehlerbehebung. Außerdem kann der Lehrer gegebenenfalls auch Aufgaben abändern, die seiner Meinung nach nicht den Ansprüchen seiner Schüler entsprechen.

### **Wie geht das Programm vor?**

Das Programm überprüft, ob der richtige Zahlenrahmen eingehalten wurde und ob die Syntax stimmt. Ebenfalls wird die richtige Lösung generiert und darauf überprüft, ob diese auch vom Programm als richtig akzeptiert wird. Dasselbe wird für nicht richtige oder halb richtige Lösungen durchgeführt. Falls Fehler aufgefunden wurden, löscht das Programm diese Aufgabe, schreibt eine Fehlermeldung ins Log und generiert eine neue.

## ***Warum ist die Überprüfung nötig?***

Dieses Vorgehen ist notwendig, um falsche Ausgaben zu vermeiden, welche den Schüler in seinem Lernerfolg einschränken würden. Da er sonst nicht das richtige erlernt, sondern sich falsche Lösungen einprägt. Es passieren leicht Fehler, allein schon, wenn das Programm während des Generierens der Aufgaben abstürzt ohne dass der Programmierer einen Fehler gemacht hat. Deshalb ist eine ausgiebige Überprüfung unerlässlich und dringend nötig.

## **Datenschutz**

Nach den im Mai neu in Kraft getretenen Datenschutzregeln<sup>17</sup> ist es einer der höchsten Prioritäten des Entwicklerteams, die persönlichen Daten der Nutzer zu schützen. Welche Daten gesondert behandelt werden und auf welche Nutzer das zutrifft, wird im Folgenden erläutert.

## **Datenschutzregelungen**

Im folgenden werden die Regeln beschrieben, die der Prototyp bei der Behandlung von sensiblen Nutzerdaten einhalten muss.

## ***Einwilligungsgrenze***

Um die Datenschutzerklärung einzuvolligen und sich zu dem Spiel anmelden zu können, muss die Person mindestens 16 Jahre alt sein. Fällt der Nutzer unter dieses

---

<sup>17</sup> Verordnung (EU) 2016/679 des Europäischen Parlaments und des Rates vom 27. April 2016 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten, zum freien Datenverkehr und zur Aufhebung der Richtlinie 95/46/EG (Datenschutz-Grundverordnung)

Alter, kann die Datenschutzerklärung nur von den Erziehungsberechtigten eingewilligt werden, ansonsten ist die Einwilligung ungültig.

Die Einhaltung dieser Regelung wird über eine CheckBox und über die Eingabe des Geburtsdatums geprüft.

### **Datenschutz**

Die eingegebenen Daten der Nutzer werden intern behandelt und somit nicht an Dritte weitergeleitet. Ebenfalls werden auch nur die Daten gespeichert, welche in der Datenschutzerklärung aufgezählt wurden.

### **Transparenz**

Um dem Nutzer eine höchstmögliche Transparenz zu bieten, wird dieser stets darüber informiert, was mit seinen Daten passiert und welche Daten von ihnen gespeichert werden.

Die Einhaltung dieser Regelung wird über ständige vertrauliche E-Mails gewährleistet, welche den Nutzer direkt informieren.

### **Verfahrensverzeichnis**

In einem sogenannten Verfahrensverzeichnis wird stets der Umgang mit personenbezogenen Daten dokumentiert. Hier wird sowohl aufgeschrieben, welche Daten gespeichert werden, zu welchem Zweck sie genutzt werden und wer alles Einsicht auf diese hat. Ohne dieses Verzeichnis gilt die gesamte Datenverarbeitung als rechtswidrig und illegal, da kein Verlauf oder Beweis vorgelegt werden kann, was nun wirklich mit den Daten geschehen ist.

## ***Einwilligung***

Die Einwilligung der Datenschutzerklärung ist freiwillig, jedoch zur Nutzung des Lernplattform essentiell. Macht man also von seinem Widerrufsrecht Gebrauch, welches der Nutzer immer ausüben kann, ist der Account ab diesem Zeitpunkt nicht mehr gültig. Außerdem ist die Erklärung nicht unnötig kompliziert geschrieben, sondern so, dass jeder den Inhalt versteht. Dies ist besonders wichtig, da die Mehrheit der Nutzer eher der jüngeren Generation angehören.

## ***Nachfrage***

Der Nutzer hat jederzeit das Recht auf Nachfrage, was genau mit seinen Daten passiert, falls ihm dies nicht in der Datenschutzerklärung und in den informierenden E-Mails ersichtlich erscheint. Dies gewährleistet wiederum Transparenz der Datenverarbeitung und das Vertrauen der Kunden.

Die Einhaltung dieser Regelung wird über die Möglichkeit, eine E-Mail an die im Impressum angegebene E-Mail-Adresse zu senden, gewährleistet.

## ***Datenvernichtung***

Der Nutzer hat zu jedem Zeitpunkt die Möglichkeit, seinen Account und somit seine ganzen gespeicherten Daten komplett löschen zu lassen. Dies bedeutet, dass all seine Daten nicht mehr zugreifbar sind und es auch keine Kopie von diesen gibt.

### **Anmerkung:**

Alle Regelungen sind gemäß der DSGVO (ab Mai 2018) und gemäß dem BDSG.

## **User**

Bezüglich des Users besteht keine Notwendigkeit einer Datenschutzregelung, da er keine persönlichen Daten preisgeben müssen.

Die Registrierung eines Users läuft ohne jegliche Datenerhebung ab. Weder für den Nutzernamen bzw. für das Passwort, noch für die Nutzung des Spiels werden Informationen über die eigene Person benötigt, da dies vollständig anonym abläuft. Selbst die beim Admin angezeigten Fortschritte der User sind inkognito.

## **Administrator**

Beim Administrator hingegen ist eine Datenschutzregelung unumgänglich, da dieser persönliche Daten über sich freigeben muss.

Es wird sein vollständiger Name und seine Email-Adresse zur Registrierung benötigt. Um die Verarbeitung seiner Daten somit rechtlich zu erlauben, muss dieser bei der Registrierung einmalig eine Datenschutzerklärung bestätigen. Ohne die Einwilligung der Datenschutzerklärung kann sich der Administrator nicht registrieren, weil seine Daten zwingend notwendig sind.

## **Datenschutzerklärung**

### ***Allgemeiner Hinweis und Pflichtinformationen***

Verantwortlich für die Datenverarbeitung ist:

Tina Amann, Dzianis Brysiuk, Andreas Glaser,  
Andreas Khouri, Noah Lehmann, Christa Strauß

Alfons-Goppel-Platz 1  
95028 Hof

Die verantwortliche Stelle entscheidet allein oder gemeinsam mit anderen über die Zwecke und Mittel der Verarbeitung von personenbezogenen Daten (z.B. Namen, Kontaktdaten o. Ä.).

## ***Mindestalter***

Um dieser Datenschutzerklärung einzuwilligen, müssen Sie bereits das 16. Lebensjahr erreicht haben. Ansonsten kann diese nur von Erziehungsberechtigen legal bestätigt werden.

## ***Widerruf Ihrer Einwilligung zur Datenverarbeitung***

Nur mit Ihrer ausdrücklichen Einwilligung sind einige Vorgänge der Datenverarbeitung möglich. Ein Widerruf Ihrer bereits erteilten Einwilligung ist jederzeit möglich. Für den Widerruf genügt eine formlose Mitteilung per E-Mail. Die Rechtmäßigkeit der bis zum Widerruf erfolgten Datenverarbeitung bleibt vom Widerruf unberührt.

## ***Recht auf Beschwerde bei der zuständigen Aufsichtsbehörde***

Als Betroffener steht Ihnen im Falle eines datenschutzrechtlichen Verstoßes ein Beschwerderecht bei der zuständigen Aufsichtsbehörde zu. Zuständige Aufsichtsbehörde bezüglich datenschutzrechtlicher Fragen ist der Landesdatenschutzbeauftragte des Bundeslandes, in dem sich der Sitz unseres Unternehmens befindet.

## ***Recht auf Datenübertragbarkeit***

Ihnen steht das Recht zu, Daten, die wir auf Grundlage Ihrer Einwilligung oder in Erfüllung eines Vertrags automatisiert verarbeiten, an sich oder an Dritte aushändigen zu lassen. Die Bereitstellung erfolgt in einem maschinenlesbaren Format.

## ***Recht auf Auskunft, Berichtigung, Sperrung, Löschung***

Sie haben jederzeit im Rahmen der geltenden gesetzlichen Bestimmungen das Recht auf unentgeltliche Auskunft über Ihre gespeicherten personenbezogenen Daten, Herkunft der Daten, deren Empfänger und den Zweck der Datenverarbeitung und ggf. ein Recht auf Berichtigung, Sperrung oder Löschung dieser Daten. Diesbezüglich und auch zu weiteren Fragen zum Thema personenbezogene Daten können Sie sich jederzeit über die im Impressum aufgeführten Kontaktmöglichkeiten an uns wenden.

## ***SSL- bzw. TLS-Verschlüsselung***

Aus Sicherheitsgründen und zum Schutz der Übertragung vertraulicher Inhalte, die Sie an uns als Seitenbetreiber senden, nutzt unsere Website eine SSL-bzw. TLS-Verschlüsselung. Damit sind Daten, die Sie über diese Website übermitteln, für Dritte nicht mitlesbar. Sie erkennen eine verschlüsselte Verbindung an der „https://“ Adresszeile Ihres Browsers und am Schloss-Symbol in der Browserzeile.

### **Anmerkung:**

Die Datenschutzerklärung wurde mithilfe des Datenschutz-Konfigurator von mein-datenschutzbeauftragter.de erstellt.

## Impressum

### Betreiber / Kontakt

<b>Betreiber</b>	Tina Amann, Dzianis Brysiuk, Andreas Glaser, Andreas Khouri, Noah Lehmann, Christa Strauß
<b>Adresse</b>	Alfons-Goppel-Platz 1, 95028 Hof
<b>Telefon</b>	(falls später mal eine Telefonnummer besteht, muss diese hier eingefügt werden.)
<b>Email</b>	(falls später mal eine Email-Adresse besteht, muss diese hier eingefügt werden.)
<b>Internet</b>	(falls später mal eine Webseite besteht, muss diese hier eingefügt werden.)

## 6 Backend Acceptance Tests

Mithilfe der sogenannten „Acceptance-Tests“ ist es möglich, die Software auf ihre Funktionen aus Sicht des Nutzers zu testen, welches durch normale Funktionstest nicht vollständig möglich ist. Acceptance-Tests sind essentiell, da sich durch diese dem Nutzer eine sehr hohe Fehlerfreiheit der Nutzeroberfläche der Software gewährleisten lässt.

Im Folgenden wird nun der Ablauf verschiedener Akzeptanztests für das Backend erläutert.

**Achtung:** Es ist zu beachten, dass mithilfe eines php-Browser getestet wird und deshalb nicht alle Szenarien vollständig mit Acceptance-Tests abgedeckt werden können. Dies betrifft jedoch nur die jeweiligen Aktionen, welche einen JavaScript Confirm-Dialog beinhalten, da der php-Browser diese nicht behandeln kann.

### „Login“

Im Folgenden wird die Login-Seite des Administrators getestet ([Login Admin](#))<sup>18</sup>.

#### 1.1. „noUsernameTest“

Es wird getestet, ob sich ein Administrator einloggen kann, ohne einen Username einzugeben.

Dies soll nicht möglich sein und die Fehlermeldung „Username cannot be blank“ soll ausgegeben werden.

#### 1.2. „invalidUsernameTest“

Es wird getestet, ob sich ein Administrator einloggen kann, ohne einen gültigen bzw. bereits registrierten Username einzugeben. Dabei wird beim Test ein Username genutzt, der sich nicht in der Datenbank befindet.

Dies soll nicht möglich sein und die Fehlermeldung „Incorrect username or password“ soll ausgegeben werden.

---

<sup>18</sup> <http://webdev-2-team-3.netlab.hof-university.de/advanced/backend/web/site/login>

### **1.3. „noPasswordTest“**

Es wird getestet, ob sich ein Administrator einloggen kann, ohne ein Passwort einzugeben.

Dies soll nicht möglich sein und die Fehlermeldung „Password cannot be blank“ soll ausgegeben werden..

### **1.4. „invalidPasswordTest“**

Es wird getestet, ob sich ein Administrator einloggen kann, ohne ein gültiges Passwort zu einem registrierten Username einzugeben. Dabei wird beim Test zu einem Username, der sich in der Datenbank befindet, ein frei gewähltes, ungültiges Passwort eingegeben.

Dies soll nicht möglich sein und die Fehlermeldung „Incorrect username or password“ soll ausgegeben werden.

### **1.5. „noInputTest“**

Es wird getestet, ob sich ein Administrator ohne jegliche Eingaben einloggen kann.

Dies soll nicht möglich sein und die Fehlermeldung „Username cannot be blank“ und „Password cannot be blank“ soll ausgegeben werden.

### **1.6. „validInputTest“**

Es wird getestet, ob ein Administrator bei gültiger und vollständiger Eingabe seiner Login-Daten erfolgreich eingeloggt und auf die richtige Seite weitergeleitet wird.

## **„Language Change“**

Im Folgenden wird das Dropdown Menü zur Sprachauswahl in Kombination mit dem Bestätigungs-Button getestet, welcher sich im unteren Bereich der Webseite befindet.

Dies wird auf der „Login Seite“ des Administrators getestet ([Login Admin](#))<sup>19</sup>. Da die Funktion auf jeder Seite zur Verfügung steht, könnte dieser Test jedoch auch auf allen anderen durchgeführt werden.

Die Ausgangssprache der Tests ist Englisch, da dies die Mindestanforderungssprache der Webseite ist.

## **2.1. „englishToGermanTest“**

Es wird getestet, ob sich die Sprache der Webseite auf Deutsch ändert, sobald man im Sprachauswahl-Menü „Deutsch“ ausgewählt und dies bestätigt hat.

## **2.2. „englishToEnglishTest“**

Es wird getestet, ob die Sprache der Webseite unverändert bleibt, wann man im Sprachauswahl-Menü die bereits angewandte Sprache auswählt. In diesem Fall die Sprache Englisch.

## **2.3. „englishToRussianTest“**

Es wird getestet, ob sich die Sprache der Webseite auf Russisch ändert, sobald man im Sprachauswahl-Menü „Русский“ ausgewählt und dies bestätigt hat.

# **„Teacher Site“**

Im Folgenden wird die Teacher-Verwaltung-Seite des Administrators getestet ([Teacher-Seite](#))<sup>20</sup>.

## **3.1. „createTeacherButtonTest“**

Es wird getestet, ob der „Create Teacher“-Button auf die Create-Teacher-Seite weiterleitet.

---

19 <http://webdev-2-team-3.netlab.hof-university.de/advanced/backend/web/site/login>

20 <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher>

### **3.2. „searchTest“**

Es wird die Such-Funktion der Tabelle getestet.

Dazu wurden zwei Test-Teacher in der Datenbank angelegt. Bei der Durchführung des Tests wird nach einem Kennzeichen (einer Zahl) in dem Namen von einem der beiden Teacher gesucht. Nun sollte der andere Teacher, welche das Kennzeichen nicht in seinem Namen trägt, verschwinden.

Damit wäre der Test erfolgreich und das Such-Feature somit funktionstüchtig.

### **3.3. „viewIconTest“**

Es wird das View-Icon in der Teacher-Tabelle getestet. Dieses soll, nachdem es angeklickt wurde, auf eine Übersichtsseite über den Lehrer weiterleiten, bei welchem das Icon in der Tabelle betätigt wurde.

### **3.4. „updateIconTest“**

Es wird das Update-Icon in der Teacher-Tabelle getestet. Dieses soll, nachdem es angeklickt wurde, auf die Lehrer-Bearbeitungsseite des Lehrers weiterleiten, bei welchem das Icon in der Tabelle betätigt wurde.

### **3.5 „deleteIconTest“**

Es soll das Delete-Icon in der Teacher-Tabelle getestet werden. Sobald dieses betätigt wurde, soll ein Popup-Fenster erscheinen. Dieses Fenster überprüft, ob man den Lehrer, bei welchem das Icon in der Tabelle betätigt wurde, tatsächlich löschen möchte. Bestätigt man nun dieses Fenster, soll der Lehrer gelöscht werden.

**Achtung:** Dieser Test kann wegen des oben bereits erwähnten Grunds (php-Browser) nicht durchgeführt werden. Dies wird deshalb manuell durch einen Tester geprüft und das Ergebnis durch ein Testprotokoll festgehalten wird.

Test:

Funktionalität des Delete-Icons in der Lehrer-Tabelle.

Voraussetzung:

Es muss ein Teacher in der Tabelle angelegt sein.

Vorgehen:

Es wird das Delete-Icon in der Reihe des Lehrers ausgewählt, welcher gelöscht werden soll. Nun wird das daraufhin erschienene Popup-Fenster mit einem Klick auf „ok“ bestätigt.

Erwartetes Ergebnis:

Der Lehrer soll nun gelöscht und somit nicht mehr in der Datenbank vorhanden sein.

Testergebnis:

Erfolgreich.

## „Header Links“

Im Folgenden wird die Link-Leiste, welche ganz oben auf der Seite, also im Header, zu sehen ist, getestet. Die Ausgangsseite dafür ist die Teacher-Seite ([Teacher-Seite](#))<sup>21</sup>.

Diese Tests sind auf jede Seite übertragbar, da überall die gleiche Funktionalität implementiert ist, und müssen deshalb nur auf einer Seite durchgeführt werden.

### 4.1. „mathRoyaleTest“

Es wird getestet, ob der „Math Royale (Administrator)“-Link an der linken Seite der Link-Leiste funktioniert. Dieser soll sobald er betätigt wurde, den Administrator auf die Home-Seite weiterleiten.

### 4.2. „teachersTest“

Es wird getestet, ob der „Teachers“-Link an der rechten Seite der Link-Leiste funktioniert. Dieser soll sobald er betätigt wurde, den Administrator auf die Teacher-Seite weiterleiten.

---

<sup>21</sup> <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher>

#### 4.3. „homeTest“

Es wird getestet, ob der „Home“-Link an der rechten Seite der Link-Leiste funktioniert. Dieser soll sobald er betätigt wurde, den Administrator auf die Home-Seite weiterleiten.

### „View Teacher“

Im Folgenden wird die View-Teacher-Seite des Admins getestet ([View-Teacher-Seite](#))<sup>22</sup>.

#### 5.1. „updateButtonTest“

Es wird der Update-Button getestet. Dieser soll den Administrator nach Betätigung, auf eine Bearbeitungsseite für den vorher ausgewählten Lehrer führen.

#### 5.2. „deleteButtonTest“

Es soll der Delete-Button getestet werden. Sobald dieser betätigt wurde, soll ein Popup-Fenster erscheinen. Dieses Fenster überprüft, ob man den Lehrer, welcher gerade auf der View-Seite angezeigt wird, tatsächlich löschen möchte. Bestätigt man nun dieses Fenster, soll der Lehrer gelöscht werden.

**Achtung:** Dieser Test kann wegen des oben bereits erwähnten Grunds (php-Browser) nicht durchgeführt werden. Dies wird deshalb manuell durch einen Tester überprüft und das Ergebnis durch ein Testprotokoll festgehalten wird.

#### Test:

Funktionalität des Delete-Buttons auf der Teacher-View-Seite.

#### Voraussetzung:

Man befindet sich auf der Teacher-View-Seite.

#### Vorgehen:

Es wird der Delete-Button betätigt. Nun wird das daraufhin erschienene Popup-Fenster mit einem Klick auf „ok“ bestätigt.

<sup>22</sup> <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher/view?id=1>

### Erwartetes Ergebnis:

Man soll auf die Teacher-Seite weitergeleitet worden sein und der Lehrer soll nun gelöscht und somit nicht mehr in der Datenbank vorhanden sein.

### Testergebnis:

Erfolgreich.

## **„Update Teacher“**

Im Folgenden wird die Teacher-Update-Seite getestet ([Update-Teacher-Seite](#))<sup>23</sup>.

Außerdem werden auch noch die Links auf der hellblauen Navigationsleiste getestet. Da die Links auf jeder Seite zur Verfügung stehen, ist dieser Test auf alle anderen Seiten übertragbar. Lediglich der Link der ID-Nummer des Teachers ist zusätzlich auf der Teacher-Update und auf der Teacher-View-Seite vorhanden, weshalb der Test auf einer dieser Seiten durchgeführt wird.

### **6.1. „noFullNameTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn kein vollständiger Name eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Full Name cannot be blank“ angezeigt werden.

### **6.2. „noPasswordTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn kein Passwort eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Password cannot be blank“ angezeigt werden.

---

<sup>23</sup> <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher/update?id=1>

### **6.3. „noUsernameTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn kein Username eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Username cannot be blank“ angezeigt werden.

### **6.4. „noEmailTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn keine E-mail-Adresse eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Email cannot be blank“ angezeigt werden.

### **6.5. „existingUsernameTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn ein bereits vergebener Username eingegeben wurde. Dafür wird ein Username eintragen, welcher bereits in der Datenbank gespeichert ist.

Dies soll nicht möglich sein und die Fehlermeldung „Username <username> has already been taken“ erscheinen“.

### **6.6. „existingEmailTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn eine bereits verwendete E-Mail-Adresse eingegeben wurde. Dafür wird eine E-Mail eintragen, welche bereits in der Datenbank gespeichert ist.

Dies soll nicht möglich sein und die Fehlermeldung „<Email-Adresse> has already been taken“ erscheinen“.

### **6.7. „noInputTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn keins der Felder ausgefüllt wurde.

Dies soll nicht möglich sein und die Fehlermeldungen „Full Name cannot be blank“, „Username cannot be blank“, „Email cannot be blank“ und „Password cannot be blank“ erscheinen.

### **6.8. „correctChangesTest“**

Es wird getestet, ob ein Lehrer erfolgreich bearbeitet wurde, wenn alle Felder vollständig und valide ausgefüllt wurden.

Es soll auf die Überblick-Seite des gerade bearbeiteten Lehrers verwiesen worden sein.

### **6.9. „noChangesTest“**

Es wird getestet, ob man auf die Überblick-Seite des gerade bearbeiteten Lehrers weitergeleitet wird, wenn keine Änderungen vorgenommen wurden und dennoch der „Save“-Button betätigt wurde.

### **6.10. „homeLinkTest“**

Es wird der Home-Link getestet, hierbei handelt es sich um den Link auf der hellblauen Navigationsleiste. Dieser soll, nachdem er angeklickt wurde, auf die Home-Seite verweisen.

### **6.11. „teacherLinkTest“**

Es wird der Teachers-Link getestet, hierbei handelt es sich um den Link auf der hellblauen Navigationsleiste. Dieser soll sobald er betätigt wurde, den Administrator auf die Teacher-Seite weiterleiten.

## 6.12. „idLinkTest“

Es wird der ID-Link getestet. Dieser soll sobald er betätigt wurde, den Administrator auf die Überblick-Seite des gerade bearbeiteten Lehrers weiterleiten.

## „Create Teacher“

Im Folgenden wird die Create-Teacher-Seite getestet ([Create-Teacher-Seite](#))<sup>24</sup>.

### 7.1. „noFullNameTest“

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn kein vollständiger Name eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Full Name cannot be blank“ angezeigt werden.

### 7.2. „noPasswordTest“

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn kein Passwort eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Password cannot be blank“ angezeigt werden.

### 7.3. „noUsernameTest“

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn kein Username eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Username cannot be blank“ angezeigt werden.

---

<sup>24</sup> <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher/create>

#### **7.4. „noEmailTest“**

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn keine E-Mail-Adresse eingegeben wurde.

Dies soll nicht möglich sein und die Fehlermeldung: „Email cannot be blank“ angezeigt werden.

#### **7.5. „existingUsernameTest“**

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn ein bereits vergebener Username eingegeben wurde. Hierfür verwendet man einen Lehrer, welcher bereits in der Datenbank steht.

Dies soll nicht möglich sein und die Fehlermeldung „Username <username> has already been taken“ erscheinen“

#### **7.6. „existingEmailTest“**

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn eine bereits verwendete E-Mail-Adresse eingegeben wurde. Hierfür verwendet man einen Lehrer, welcher bereits in der Datenbank steht.

Dies soll nicht möglich sein und die Fehlermeldung „Email <Email-Adresse> has already been taken“ erscheinen“.

#### **7.7. „noInputTest“**

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn keins der Felder ausgefüllt wurde.

Dies soll nicht möglich sein und die Fehlermeldungen „Full Name cannot be blank“, „Username cannot be blank“ , „Email cannot be blank“ und „Password cannot be blank“ erscheinen.

## 7.8. „correctInputTest“

Es wird getestet, ob ein Lehrer erfolgreich erstellt wurde, wenn alle Felder vollständig und valide ausgefüllt wurden.

Es soll auf die Teacher-Seite verwiesen worden sein.

## „Logout“

Im Folgenden wird der Logout-Link in der Link-Leiste getestet. Dies wird ausgehend von der Teacher-Seite getestet ([Teacher-Seite](#))<sup>25</sup>.

Dieser soll nachdem er angeklickt wurde, den Lehrer aus geloggt und auf die Login-Seite verwiesen haben.

---

<sup>25</sup> <http://admin.webdev-2-team-3.netlab.hof-university.de/teacher/index>

## 7 Verweise und Quellen

### Abbildungsverzeichnis

Abbildung 1: Markdown (Quelle fehlt).....	15
Abbildung 2: Markdown (Quelle fehlt).....	15
Abbildung 3: Bsp. halb richtige Antwort.....	21
Abbildung 4: Bsp. leichte Aufgabe.....	22
Abbildung 5: Bsp. schwere Aufgabe.....	22
Abbildung 6: Bsp. Fragen-Antwort-Aufgabe mit Rechenweg.....	23
Abbildung 7: Smileys zur Bewertung der Hilfe.....	37
Abbildung 8: Ablauf der Fairness-Regelung.....	39
Abbildung 9: Wireframes (Aufgaben).....	40
Abbildung 10: Wireframes (Info).....	41
Abbildung 11: Wireframes (Hilfe Button).....	41
Abbildung 12: Wireframes (Fortschritt).....	41
Abbildung 13: Wireframes (Frage).....	42
Abbildung 14: Wireframes (Hilfe benötigt).....	42
Abbildung 15: Wireframes (Antwort).....	42
Abbildung 16: Wireframes (Frage beantwortet).....	43
Abbildung 17: Wireframes (Benachrichtigung).....	43
Abbildung 18: Erstellen eines Backups (Screenshot, Andreas Khouri).....	45
Abbildung 19: Wiederherstellung eines Backups (Screenshot, Andreas Khouri).....	46

### Tabellenverzeichnis

Rollenverteilung.....	7
Backup-Strategie.....	13
Markdown-Syntax.....	15
Entscheidungsmatrix.....	18
Entscheidungsmatrix(Bewertungen).....	19
Meetings/Teamtreffen.....	42
Aufgabenverteilung(Sprint1).....	45
Unterschriftensammlung(Selbstarbeitserklärung).....	48

**Abbildungsquellen:**

Abb.1:	selbst ausgedachtes Beispiel
Abb.2:	Gerhard Steidle (1975), Einführung in die Algebra, Bayerischer Schulbuch-Verlag, ISBN 3-7627-3136-5, Seite 33. (Zahlen geändert)
Abb.3, 4:	Gerhard Steidle (1975), Einführung in die Algebra, Bayerischer Schulbuch-Verlag, ISBN 3-7627-3136-5, Seite 46. (Zahlen geändert)
Regelsatz 3	Bild von Dzianis Brysiuk (Affinity Designer)
Abb. 7	<a href="https://t4.ftcdn.net/jpg/01/81/80/29/240_F_181802976_C5WkxXnVJuc2HurWuTdvFTAVuhoJWfJZ.jpg">https://t4.ftcdn.net/jpg/01/81/80/29/240_F_181802976_C5WkxXnVJuc2HurWuTdvFTAVuhoJWfJZ.jpg</a>

**Quellenverzeichnis:**

Filter:	<a href="https://docs.angularjs.org/api/ng/filter">https://docs.angularjs.org/api/ng/filter</a>	24.04.2018
Services:	<a href="https://docs.angularjs.org/api/ng/service">https://docs.angularjs.org/api/ng/service</a>	24.04.2018

## 8 Anhang

Im Anhang sind für den Prototypen des kollaborativen Spieles irrelevante Dokumente, die aber sehr wohl für den Ablauf des Projektes wichtig sind, aufgeführt. Zu diesen gehören im allgemeinen Projektmanagement-Dokumente, wie die Sprintplanungen, das Project-Backlog, eine Risikoanalyse, ein Backup-/ und Restoreplan und die Dokumentationen der für den Prototypen notwendigen Installationen auf dem Server.

## Sprintplanungen

Hier wird darauf eingegangen, wie die einzelnen Sprints geplant und durchgeführt wurden. Es werden die Aufgaben aufgelistet, den Teammitgliedern zugeordnet, zeitlich abgeschätzt und anschließend priorisiert. Abschließend wird jeder Sprint nochmals in der Retrospektive bewertet.

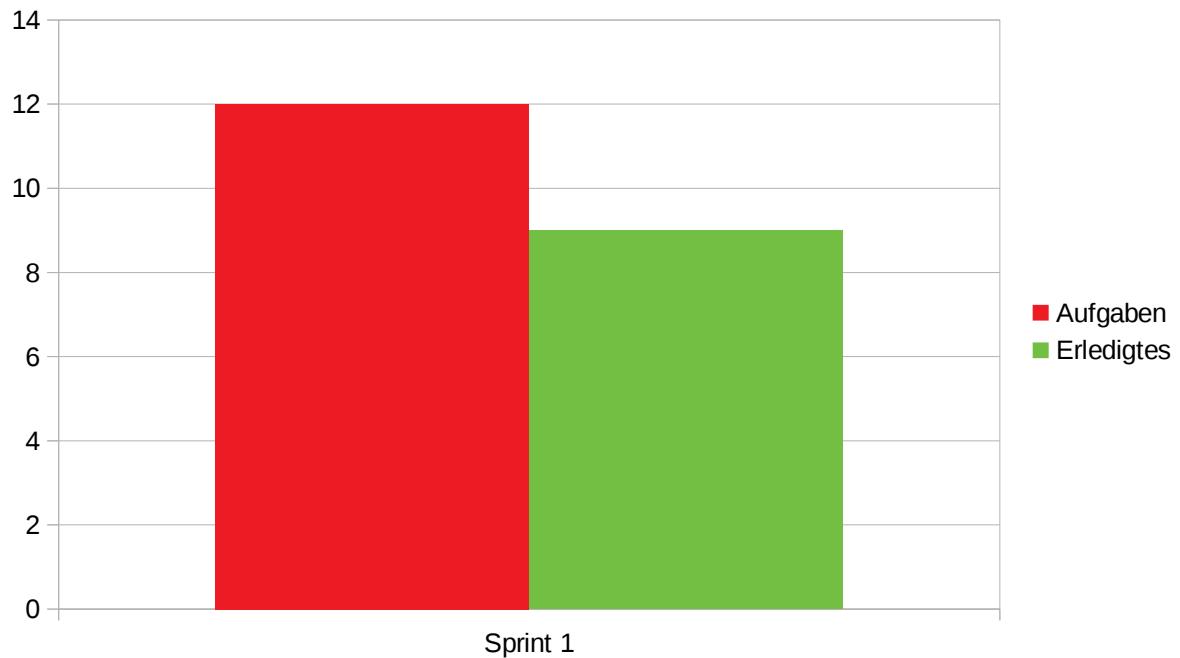
### Erster Sprint

Dieser initiale Sprint ist für Grundlegende Aufgabenverteilung und Sammlung von Ideen bezüglich des kollaborativen Spieles zuständig. In ihm werden die Zuständigkeiten der Teammitglieder für elementare Teile des Projektes festgelegt.

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Rollenverteilung	Team	26.03.'18	1h	1h	4
2	GitHub Repository erstellen, Einladungen versenden	Andreas G.	29.03.'18	1h	0,5	5
3	Trello einrichten, Einladungen versenden	Andreas G.	29.03.'18	1h	1h	5
4	Testen Zugriff auf Systeme	Dzianis, Noah	29.03.'18	0,5h	0,5h	5
5	Backup und Restore Plan erstellen	Andreas K.	02.04.'18	1,5h	1,5h	3
6	Installation von HumHub	Andreas K., Dzianis, Noah	05.04.'18	1h	1h	5
7	Def. Regelsatz 1	Christa, Tina	10.04.'18	2*2,5h	2*2,5h	5
8	Def. Regelsatz 2	Andreas G. Andreas K.	10.04.'18	2*2,5h	2*3h	5
9	Def Regelsatz 3	Dzianis, Noah	10.04.'18	2*2,5h	2*3h	5

10	Einarbeiten in Projektframeworks (Microsoft Project)	Dzianis, Noah	16.04.'18	3h	offen	
11	Risikoplan fertigstellen	Noah	19.04.'18	3h	offen	
12	Changelog fertigstellen	Noah	19.04.'18	2h	offen	



## Rückblick

In diesem Sprint wurden einige wichtige Aufgaben erledigt, das Team hat untereinander die Rollenverteilung ausgemacht und es wurden die ersten Dokumente zum festhalten des Projektverlaufes angefertigt. Rückblickend ist die strukturierte Planung des Verlaufes allerdings noch besser zu bewältigen. Einige Kennzahlen sollten gesammelt werden, nach denen Übersichten erstellt werden können, um beim Abschluss eines Sprints mit Prof. Dr. Heym die Fortschritte besser evaluieren zu können. Positiv zu bewerten ist die Arbeitseinstellung der einzelnen Teammitglieder, die ihre Aufgaben bis jetzt ausnahmslos ohne Verzug erledigt haben. Demnach ist der Projektverlauf zu

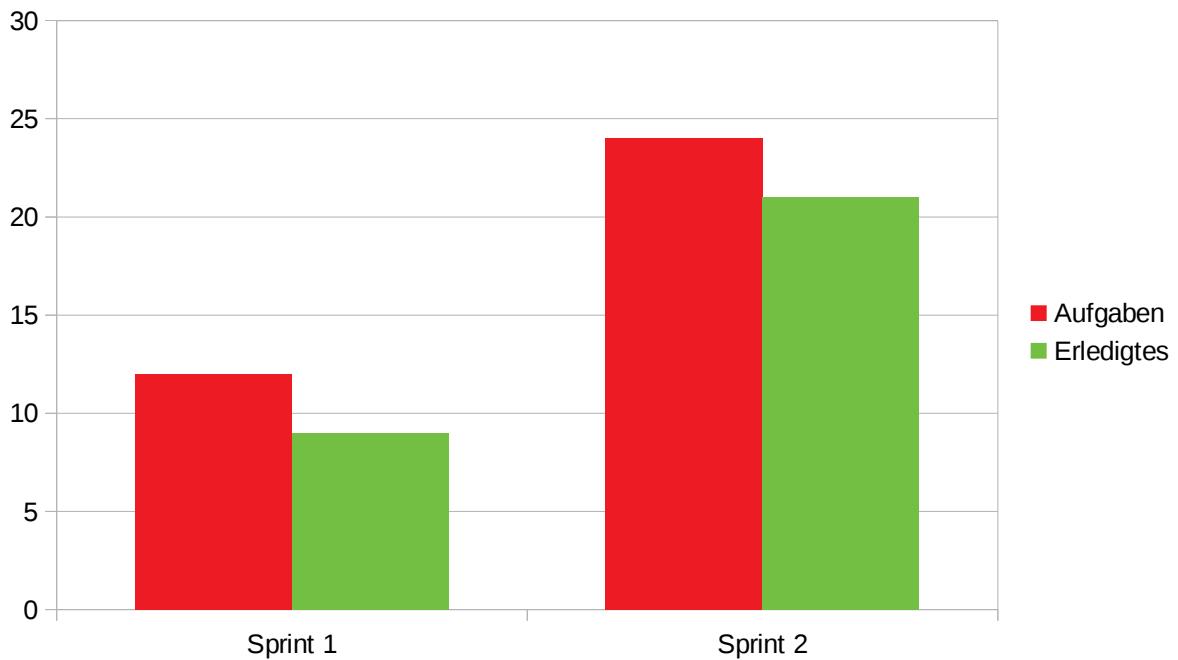
diesem Zeitpunkt absolut im Soll, der Abschluss der Projektes ist momentan nicht gefährdet.

## Zweiter Sprint

Im zweiten Sprint liegt der Fokus nicht mehr auf den Regeln der Kollaboration, sondern auf den prototyphaften Aufbau des Endproduktes an sich. Es sollen Ideen gesammelt werden, die das Design, die Usability und die Funktionen der Oberfläche des Users beschreiben

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Regeln für das Nutzen von GitHub	Andreas G.	14.05.'18	2h		3
2	Analyse von AngularJS	Tina, Christa	07.05.'18	3h		3
3	Beispielhafte Aufgaben in HumHub	Noah	08.05.'18	2h	1,5h	4
4	Präsentation Sprint 1	Noah	08.05.'18	3h	3,5h	5
5	Erste Prototypen in Angular/Yii/etc.	Andreas G., Andreas K.	07.05.'18	5h		4
6	Installation von YiiBasic	Noah	03.05.'18	2h	3h	5
7	Installation von YiiAdvanced auf Server	Andreas K., Andreas G., Dzianis	14.05.'18	2h		5
8	Risikoanalyse	Noah	19.03.'18	2h	3h	2
9	Wireframes	Dzianis	03.05.'18	4h		4
10	Beschreibung der Wireframes	Noah	16.05.'18	3h	3h	3
11	Installation von HumHub Modulen	Andreas K., Noah	03.05.'18	1h	1h	1
12	User Stories	Tina, Christa	16.05.'18	5h		3



## Rückblick

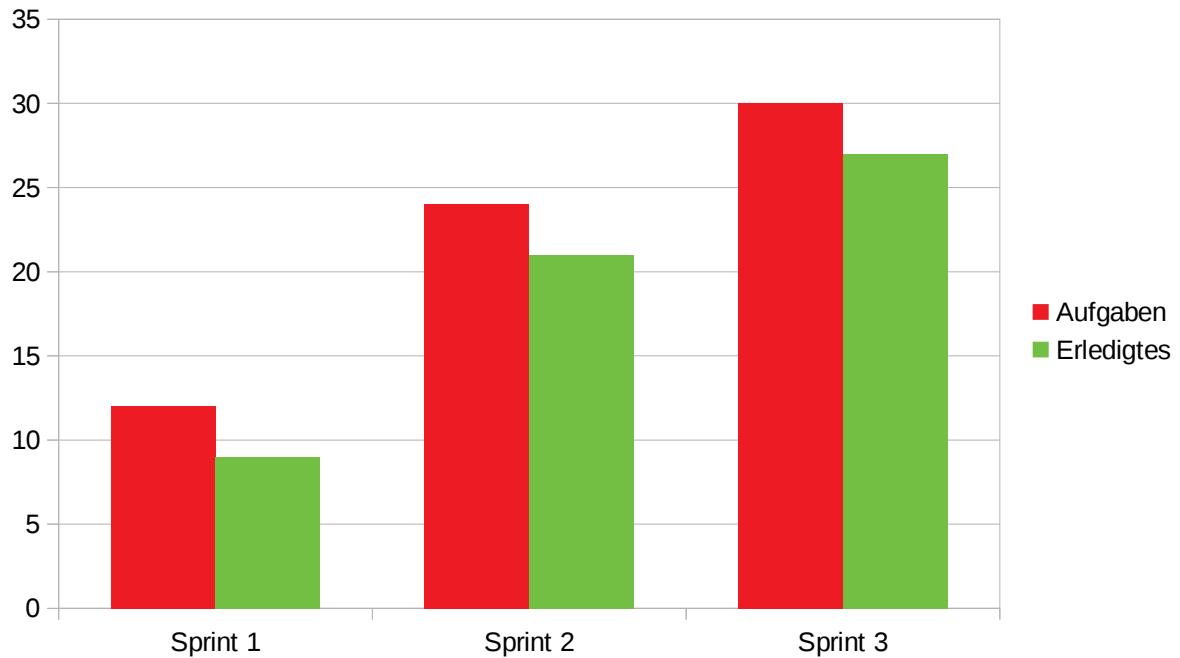
Der zweite Sprint legte weit mehr Wert auf die Aufgabe der Entwicklung einer Website, als es der erste tat. Das Team hat sich dementsprechend einige Arbeit gemacht, Ideen zu sammeln, Oberflächen prototyphaft zu designen und die Ergebnisse alle festzuhalten. Auf die User Experience wurde großer Wert gelegt. Optisch wurden die Ergebnisse aus Sprint 1 in Form von Wireframes festgehalten. Diese enthielten aber auch die Ergebnisse neuer Überlegungen. Neben der Ideensammlung wurde auch weiterhin die Arbeit auf dem Server gepflegt. Mit der Einhaltung des vorgegebenen Zeitplans hat das Team zu diesem Zeitpunkt keinerlei Probleme, was sich voraussichtlich auch nicht ändern wird.

## Dritter Sprint

Der dritte Sprint sammelt jetzt nicht mehr nur Ideen, sonder beschäftigt sich ausgiebig mit dem Sammeln und Beschreiben der Funktionen des Prototypens. Es werden alle Funktionen in die Dokumentation übernommen, um die Programmierung am Ende des Projektes nur noch nach der vorgegebenen Anleitung durchführen zu müssen.

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Trennung des Admin Front-/Backends und des User Frontends	Dennis, Noah	06.06.'18	4h	3,5h	4
2	DSGVO Regeln und Impressum	Christa	06.06.'18	3h	5h	2
3	Ideen zur Veränderung des Layouts durch Lehrer im Admin Frontend	Andreas K.	06.06.'18	1h	1h	1
4	Aufgaben und Datenstrukturen der Aufgaben	Andreas G.	06.06.'18	2h	2h	4
5	Aufgabengeneratoren	Andreas K., Noah	06.06.'18	6h	6h	5
6	Wie können Aufgaben vom Lehrer nach bearbeitet werden	Christa, Tina	06.06.'18	2,5h	3h	2



## **Rückblick**

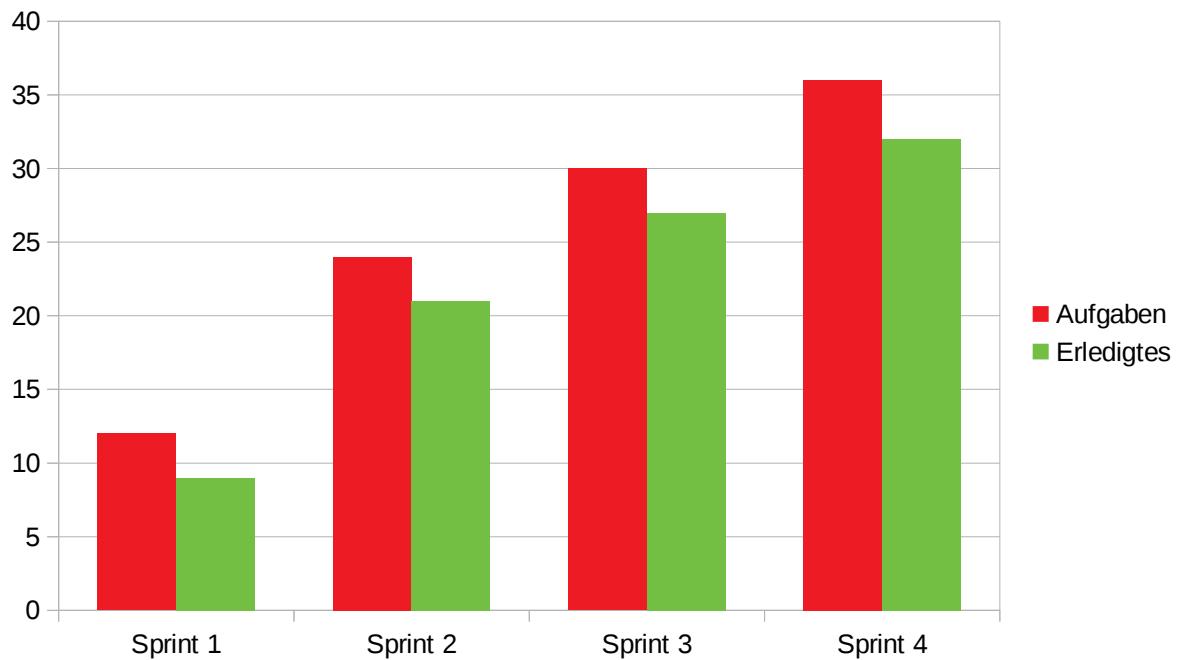
Es wurden alle Aufgaben aus dem Sprint vollständig bearbeitet und im Team gemeinsam evaluiert. Die grobe Struktur der Website steht, das Team hat sich auf die Funktionen des Prototypen und die Art der Umsetzung dieser Funktionen geeinigt. Das Team liegt in der Bearbeitung der Aufgaben immer noch im von Prof. Dr. Heym vorgegebenen Zeitraum.

## Vierter Sprint

Der vierte Sprint ist ein sehr kurzer Sprint. Er ist lediglich 8 Tage lang, vom 07.06.-15.06.'18. Er wurde vom Auftraggeber Prof. Dr. Heym angesetzt, um die Abschließenden Vorbereitungen zu treffen, damit in den späteren Sprints nur noch Programmiert werden muss. Die primäre Aufgabe ist es also, die Server Konfiguration so abzuschließen, dass danach nichts mehr daran geändert werden muss.

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Installation von Gii in Yii2 Advanced	Andreas K.	15.06.'18	1h	1h	5
2	Aktivierung von PrettyURL in Yii2 Advanced	Andreas G.	15.06.'18	0,5h	0,5h	3
3	Ausarbeitung der Funktionen im Backend	Noah, Dennis, Andreas K., Andreas G.	15.06.'18	2h	2,5h	4
4	DB Migration und Beispieldatensätze	Noah, Dennis	15.06.'18	3h	3,5h	3
5	Deaktivierungsroutine für Beispieldatensätze	Noah, Dennis	15.06.'18	1h	-	1
6	Codeception Akzeptanz Tests	Tina, Christa	15.06.'18	8h	9h	4



## **Rückblick**

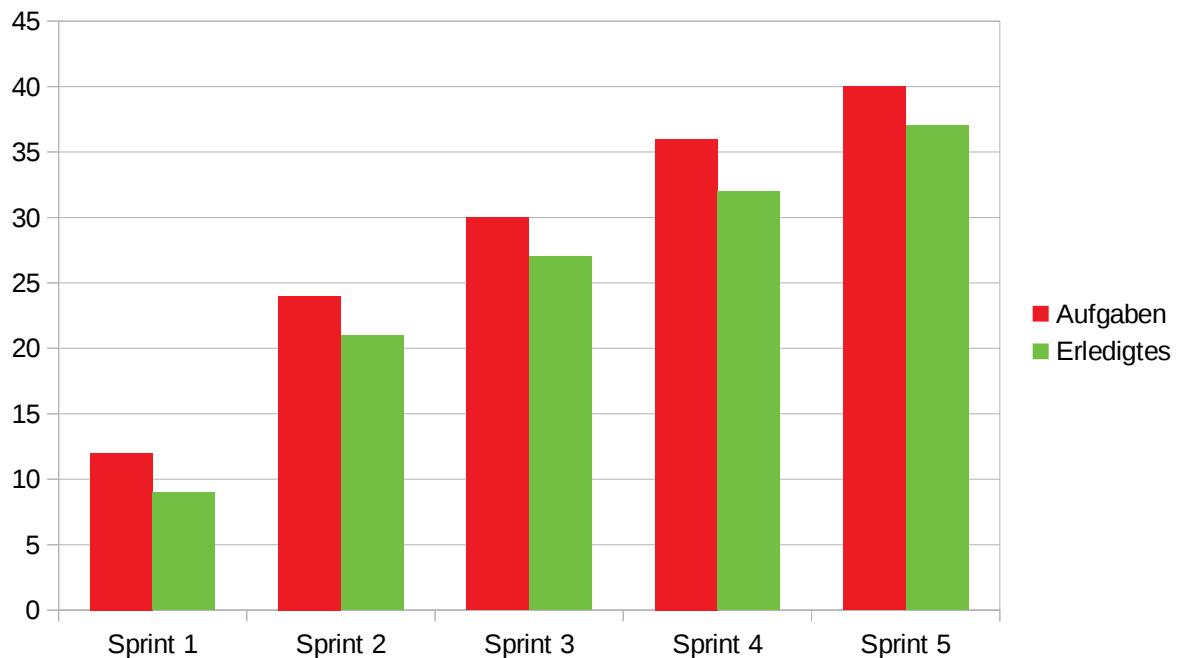
Im vierten Sprint wurden nun die letzten Vorbereitungen für die Programmierung getroffen. Einige Funktionen, die noch unklar waren, wurden nun endgültig definiert. Der Server ist komplett konfiguriert und kann nun genau so genutzt werden, wie er gebraucht wird. Auch die Datenbank ist fertig gestellt und bereits nutzbar. Lediglich in der Implementierung der Tests ist das Team nicht fertig geworden, da diese aufwendiger waren, als geplant.

## Fünfter Sprint

Der fünfte Sprint soll sich nun erstmals mit der Programmierung der Anwendung beschäftigen. Jedoch ist auch der fünfte Sprint sehr kurz. Die kurze Zeit ist deshalb gewählt, weil der fünfte Sprint die einzelnen Teile der Website, z.B. Authentifizierung und Layout, prototyphaft zu implementieren, um sie im letzten Sprint zu verbinden.

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Prototyp App AngularJS Yii2 (Routing/Layout)	Andreas G., Andreas K.	22.06.'18	4h	4h	5
2	Prototyp App AngularJS Yii2 (Authentifizierung)	Dennis	22.06.'18	4h	6h	5
3	Literaturliste	Noah	22.06.'18	2h	2,5h	2
4	Codeception Tests	Christa, Tina	22.06.'18	10h	9,5h	3



## Rückblick

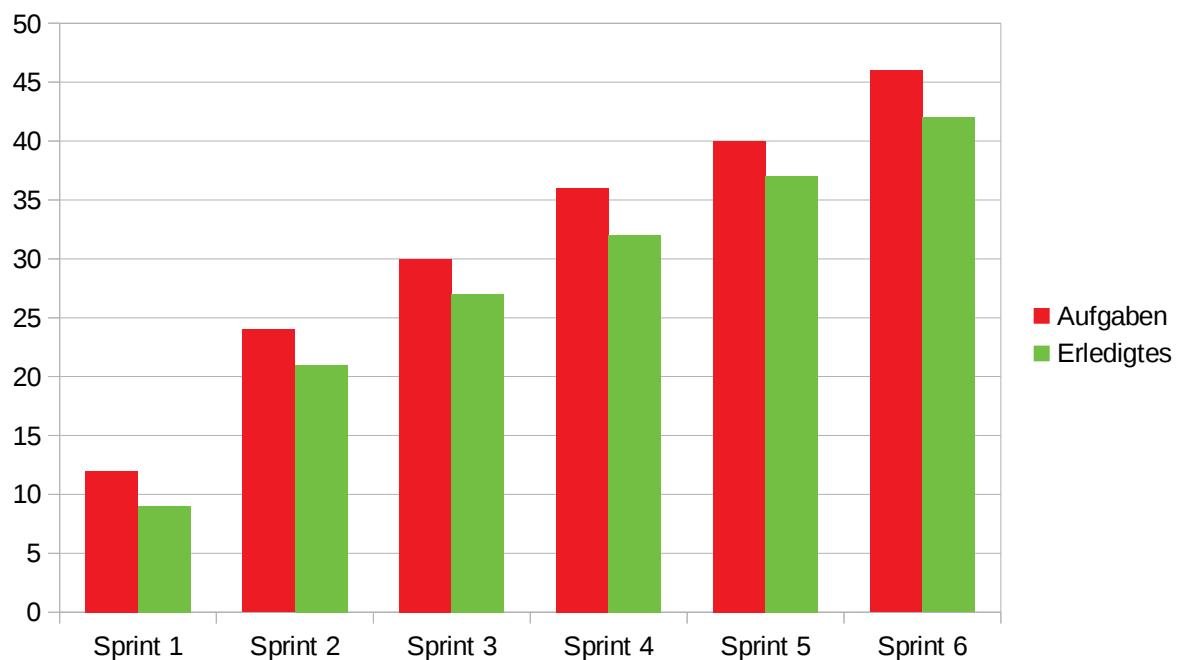
Nachdem der Sprint sehr kurz war, die Liste der Aufgaben allerdings auch, gibt es in diesem Sprint nicht viel zu betrachten. Die Prototypen sind fertig implementiert und müssen nur noch verbunden werden. Auch die Tests aus dem vorherigen Sprint wurden vollständig nachgeholt. Das Team ist somit noch im vorgegebenen Zeitraum und wird das Projekt aller Voraussicht nach auch zufriedenstellend abschließen können.

## Sechster Sprint

Der sechste Sprint ist der letzte Sprint, in dem am fertigen Prototypen gearbeitet wird. Er ist dafür da, die Prototypen aus Sprint 5 zu verbinden und die letzten Funktionen zu implementieren. Dies sollte bis zum 01.07.'18 abgeschlossen sein. Dort wird der Codefreeze angesetzt, um Probleme bei der Präsentation zu vermeiden.

### Aufgaben

Nr	Bezeichnung	Bearbeiter	Fällig bis	Geschätzter Aufwand	Wirklicher Aufwand	Priorität (0<5)
1	Kommunikation mit Server und Client auf RESTful API umstellen	-	-	-	-	0
2	Anzeigen der Statistik einer Session	Tina, Christa	29.06.'18	6h	5h	4
3	Anlegen von Nutzern	Andreas G.	29.06.'18	2h	2h	5
4	Buttons zur Kollaboration	Andreas K.	29.06.'18	4h	3,5h	3
5	Stream für Fragen	Dennis, Andreas G.	29.06.'18	3h	4,5h	3
6	Mehrsprachigkeit	Dennis	29.06.'18	5h	7h	4



## Rückblick

Im sechsten Sprint wurden die letzten Funktionen implementiert, außerdem wurden mehr Funktionen implementiert und zumindest ansatzweise fertiggestellt, als Anfangs gedacht. Zu beachten ist, dass die Dokumentation einige Punkte enthält, die ein fertiges Produkt haben sollte, welche aber aufgrund der kurzen Zeit dieses Projektes nicht wirklich implementiert werden konnten.

## Termine und zeitliche Abläufe

In diesem Kapitel wird kurz dargestellt, wann das Team sich für Meetings getroffen hat und was der thematische Inhalt dieser Meetings war.

### Meetings und Team-Treffen

Bezeichnung	Datum	Inhalt
Meeting 1	22.03.'18	In diesem Meeting wurden die Teams gebildet, es wurde ein allgemeiner Ausblick auf die Aufgaben gegeben und die ersten Vorbereitungen wurden getroffen.
Meeting 2	23.03.'18	Hier wurden den Teams im Allgemeinen das agile Vorgehen in Projekten vermittelt. Gezeigt wurden hier die nötigen Planungen und die Dokumente, die angefertigt werden könnten, um den Überblick über die Planung zu behalten.
Meeting 3	26.03.'18	Das erste Meeting für Team 3 in voller Besetzung. Es wurden Aufgabenbereiche und Zuständigkeiten geklärt. Diese werden in Kapitel 3 genauer beschrieben. Außerdem wurde eine Entscheidungsmatrix erstellt, die die Benutzung der Module in HumHub als Thema hatte.
Meeting 4	05.04.'18	Es wurde die allgemeine Intention des Projektes wiederholt, Hierfür hat Prof. Dr. Jürgen Heym für alle Teams die Regelsätze klar definiert, nach denen sich die Teams untereinander Ideen für das Spiel sammeln sollen.
Meeting 5	06.04.'18	Diesem Meeting wurde genauer auf das zu benutzende Framework Bootstrap eingegangen, eine Kurzeinführung wurde den Studenten nahegelegt.
Meeting 6	06.04.'18	Das Team hat die Regelsätze auf 2er Gruppen verteilt. Diese sollen sich Ideen und Maßnahmen für die beschriebenen Probleme überlegen.
Meeting 7	10.04.'18	Das Team hat sich getroffen, um seine Ergebnisse bei der Entwicklung der Regelsätze zu besprechen und gegebenenfalls abzuändern.
Meeting 8	12.04.'18	Es wurden die ersten Züge des Frameworks AngularJS vorgestellt

Meeting 9	13.04.'18	Der letzte Teil von AngularJS wurde vorgestellt, es wurde gezeigt, wie die Angular-Anwendungen in die HTML Seiten eingefügt werden sollen. Meetings für den Sprint Review mit Prof. Dr. Heym wurden festgelegt.
<b>Ende Sprint 1</b>		
Meeting 10	16.04.'18	Absprache und Abstimmung mit Auftraggeber über Dokumentation und Projektverlauf
Meeting 11	17.04.'18	Teamtreffen zu Evaluation der Ergebnisse aus dem Sprintreview mit Prof. Dr. Heym
Meeting 12	19.04.'18	Vorlesung zum Thema AnglarJS
Meeting 13	20.04.'18	Vorlesung zum Thema AnglarJS
Meeting 14	03.05.'18	Vorlesung
Sprint Vortrag	08.05.'18	Mit Prof. Joe Oaks, Penn-State University, wurde ein kurzes Review der Ergebnisse des 1. Sprints durchgeführt.
Meeting 15	10.05.'18	Einführung in Yii2
Meeting 16	11.05.'18	Einführung in Yii2
Meeting 17	16.05.'18	Absprache und Abstimmung mit Auftraggeber, Evaluierung der Dokumentation und Ansprechen von Backup Problemen
<b>Ende Sprint 2</b>		
Meeting 18	17.05.'18	Das Ziel und die Aufgaben für Sprint 3 wurden von Prof. Dr. Heym vorgestellt und weiter erläutert.
Meeting 19	24.05.'18	Vorlesung zum Thema Yii und die Nutzung der Advanced Version des Frameworks
Meeting 20	25.05.'18	Vorlesung und Erinnerung an die Pflichten eines Website Betreibers, Impressum und DSGVO
Meeting 21	25.05.'18	Treffen zur gemeinsamen Bearbeitung verschiedener Aufgaben
Meeting 22	01.06.'18	Webkonferenz
<b>Ende Sprint 3</b>		
Meeting 22	07.06.'18	Aufstellung neuer Aufgaben für Sprint 4
Meeting 23	08.06.'18	Besprechung der Test Kriterien der Website und des Codes, Nutzung von Codeception
Meeting 24	14.06.'18	Einführung in den Code Generator Gii in dem Framework Yii2 und die Nutzung von Gii
<b>Ende Sprint 4</b>		
Meeting 25	15.06.'18	Aufstellung der neuen Aufgaben für Sprint 5
Meeting 26	21.06.'18	Erinnerung an Quellenangaben
<b>Ende Sprint 5</b>		
Meeting 27	22.06.'18	Aufstellung der neuen Aufgaben für Sprint 6

Meeting 28	23.06.'18	Arbeit am Projekt
Meeting 29	28.06.'18	Arbeit am Projekt mit Möglichkeit zu Fragen an Prof. Dr. Jürgen Heym
Meeting 30	29.06.'18	Teamarbeit bis 9:45
Meeting 31	19.06.'18	Arbeit am Projekt mit Möglichkeit zu Fragen an Prof. Dr. Heym, Aufgaben sind auf Trello und wurden unter den Teammitgliedern verteilt. Zusammenarbeit ist deshalb sinnvoll, da man einander helfen kann.
Ende Sprint 6		
<b>Codefreeze am 01.07.2018</b>		

## Zusammenarbeitsregeln

In diesem Teil des Anhangs werden Dokumente festgehalten, die sich auf das Projekt und die Zusammenarbeit im Team beziehen. Im Allgemeinen handelt es sich damit immer um Regeln, die bei der Nutzung von Ressourcen beachtet werden müssen.

## Backup- / Restore-Vorgang

In diesem Dokument wird kurz die Strategie vorgestellt, nachdem das Team Backups des Servers anlegt und verwaltet.

### ***Erstellen eines Backups***

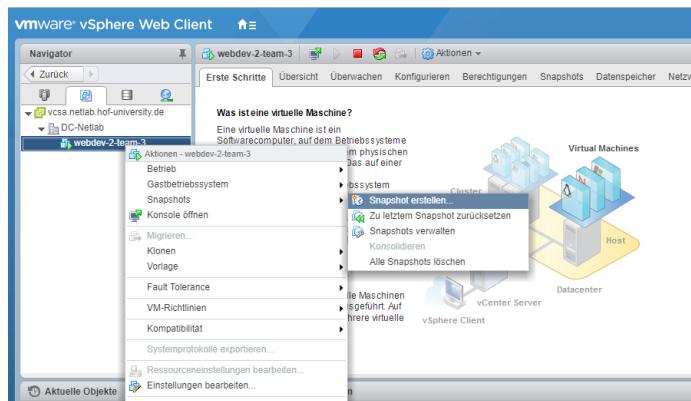


Abbildung 21: Erstellen eines Backups (Screenshot, Andreas Khouri)

- 1) Einloggen in „vSphere Web Client (Flash)“
- 2) Auswahl der virtuellen Maschine im Navigator
- 3) Rechtsklick auf die virtuelle Maschine
- 4) "Snapshot" wählen
- 5) "Snapshot erstellen" wählen
- 6) Namen des Snapshots einstellen
- 7) Beschreibung des Snapshots einstellen

- 8) „Snapshot des Arbeitsspeichers der virtuellen Maschine erstellen“ auswählen
- 9) Eingabe bestätigen

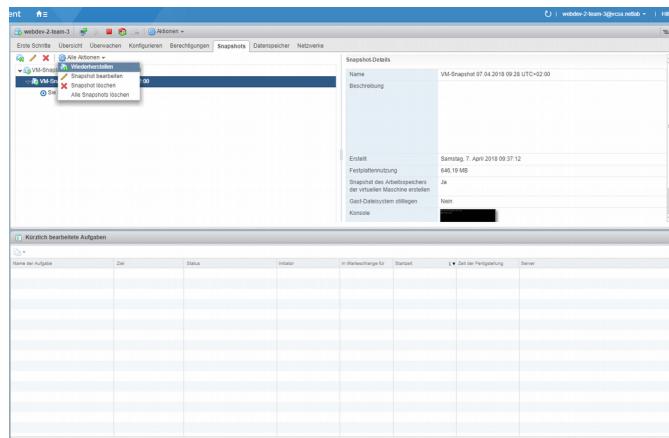
Wird der Snapshot zum ersten Mal erstellt wird ein komplettes Abbild der virtuellen Maschine inklusive Arbeitsspeicher erstellt. Die Speicherung des Arbeitsspeicherinhaltes ist sofern wichtig, dass es die Wahrscheinlichkeit eines fehlerfreien, konsistenten Wiederherstellens erhöht, falls Dateien bei der Erstellung des Snapshots geöffnet und nicht geschlossen sind.

Beim Erzeugen weiterer Snapshots wird nur das Delta zwischen den beiden Snapshots gespeichert, was zu reduzierter Speicherplatzverwendung führt.

Backups sollen entweder zu regelmäßigen Zeitpunkten (z.B. täglich oder an festgelegten Wochentagen) oder vor Einspielen eines Updates erstellt werden.

### ***Wiederherstellen eines Backups***

- 1) Einloggen in „vSphere Web Client (Flash)“
- 2) Auswahl der virtuellen Maschine im Navigator
- 3) Auswahl des Reiters "Snapshots"
- 4) Auswahl des wiederherzustellenden Snapshots
- 5) "Alle Aktionen" auswählen
- 6) "Wiederherstellen" auswählen
- 7) Sicherheitsfrage im folgenden Feld mit "Ja" beantworten



*Abbildung 22: Wiederherstellung eines Backups  
(Screenshot, Andreas Khouri)*

Falls bei der Wiederherstellung „Anhalten der virtuellen Maschine“ ausgewählt wurde, muss diese nach erfolgreichem Wiederherstellen wieder gestartet werden.

Ein Wiederherstellen eines Backups wird dann notwendig, wenn die virtuelle Maschine Inkonsistenzen aufweist oder ein eingespieltes Update rückgängig gemacht werden soll.

Dadurch, dass mehrere Wiederherstellungspunkte vorhanden sind, ist es möglich den Wiederherstellungszeitpunkt auch früher als das letzte Backup auszuwählen.

## **Backup-Strategie**

Wir haben uns dafür entschieden zu Beginn des Projekts und anschließend ab Beginn der Implementierungsphase täglich um 18.00 Uhr Backups zu erstellen, da Backup-Verwalter und Entwickler so ohne Leerlaufzeit unabhängig voneinander agieren können. Außerdem benötigt das Erstellen eines Backups keinen großen Zeitaufwand, da diese lokal auf dem Server der Hochschule Hof gespeichert werden. Zudem führt ein kurzes Backup-Intervall zu möglicher Arbeitersparnis nach einer Wiederherstellung.

Backup-Strategie	
<b>Backup-Intervall:</b>	<i>Täglich</i>
<b>Intervall-Beginn:</b>	<i>Beginn der Implementierungsphase</i>
<b>Intervall-Ende:</b>	<i>Fertigstellung des Projekts</i>
<b>Backup-Zeitpunkt:</b>	<i>18.00 Uhr</i>
<b>Speicherort:</b>	<i>Server der Hochschule Hof</i>
<b>Verantwortlicher:</b>	<i>Andreas Khouri</i>

## GitHub Nutzungsregeln

In diesem Kapitel wird kurz beschrieben, was bei der Nutzung des Versionierungstools GitHub beachtet werden muss, damit es nicht zur Zerstörung oder zum generellen Verlust oder Inkonsistenz von Daten kommt.

### ***Repository***

Das Repository stellt die zu Versionierenden Daten dar, die mithilfe von Git verwaltet und geteilt werden. Es umfasst erstellten Code sowie dazu gehörige Dokumente.

### ***Git-Admin***

Ein Teammitglied dass für die Verwaltung des Git Repositorys zuständig ist. Er/Sie führt merges zwischen Branches aus und löscht nicht mehr benötigte Branches. Für dieses Projekt wurde Andreas Glaser als Git-Admin bestimmt.

### ***Pull***

Pullen ladet die aktuelle Version des Repository vom Server herunter und überprüft hierbei auf Konflikte. Bevor mit dem Arbeiten begonnen wird ist ein Pull stark zu empfehlen, dadurch können viele Konflikte vermieden werden.

### ***Commit***

Erstellte Änderungen werden mit einem Commit zusammengefasst und kurz erläutert. Hierbei ist es wichtig nicht nach Beendigung eines Arbeitstages ein einziges mal zu Commiten, sondern nach Beenden einer Aufgabe oder wenn man sich einer anderen Aufgabe widmen möchte. Dadurch wird die erledigte Arbeit nicht mit anderen Arbeiten zu vermischt und die Änderungen können einzeln rückgängig gemacht

werden. Die Commit-Nachricht soll kurz und prägnant beschreiben welche Änderungen durch diesen Commit stattgefunden haben, dies erleichtert die Fehlersuche enorm und auch die anderen Programmierer können so schnell und ohne großen Aufwand verstehen was dieser Commit geändert hat.

## **Push**

Bei einem Push werden die aktuell durch Commits dokumentierten Änderungen auf den Server hochgeladen. Um zu vermeiden dass Konflikte erst hierbei auftreten sollte immer zuerst gepullt werden bevor Änderungen gepusht werden. Gepusht wird grundsätzlich nur auf nicht Release Branches, Änderungen kommen auf den Release Branch erst durch Tests und die Abnahme des Git-Admins. Hiervon ausgenommen sind nicht Code Dokumente wie Dokumentation, Bilder und Diagramme.

## **Konflikte**

Konflikte können bei verschiedenen Vorgängen auftreten und sind immer von der Person bei der sie Auftreten zu beheben, sollte dies nicht möglich sein, wendet sich diese Person an den Git-Admin. Dieser kann entweder den Konflikt lösen oder findet den Verursacher des Konfliktes und lässt ihn diesen lösen.

## **Branches**

Branches ermöglichen es das Repository in verschiedene Varianten zu unterteilen. Ein Branch für den Release und mehrere für verschiedene Aufgaben. Der Release Branch enthält immer eine funktionierende Version der Software. Änderungen die auf den Release Branch hochgeladen werden sollen erst ausführlich getestet werden um ihre Korrektheit sicherzustellen. Die Branches für die Bearbeitung von Aufgaben können von allen Teammitgliedern bei Bedarf erstellt werden, gelöscht werden sie aber nur vom Git-Admin. Jeder Aufgaben Branch besitzt einen Verantwortlichen dieser muss bei der Erstellung festgelegt werden, er kann bei Konflikten hinzugezogen werden. Ist die

Aufgabe eines Branches erledigt worden, so wird er, nach ausführlichen Tests, mit dem Release Branch gemerged, dies übernimmt der Git-Admin. Er löst auch hierbei auftretende Konflikte und nach erfolgreichen Mergen löscht er den Branch.

### ***Mergen***

Mergen bezeichnet das Zusammenführen von zwei Branches. Hierbei treten häufig viele Konflikte auf deswegen mergen immer mindestens zwei Personen, ist einer der Branches der Release Branch muss eine der Personen der Git-Admin sein.

## 9 Erklärung zur Selbstarbeit

### Erklärung

Ich erkläre hiermit, dass ich meinen Beitrag zur vorliegenden Gruppenarbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; das gleiche gilt für die von den auf dem Titelblatt der Arbeit genannten Autoren gemeinsam verfassten Teile. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde nach meiner besten Kenntnis bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Hof, den 04.07.2018

Andreas Glaser,	<u>A. Glaser</u>
Andreas Khouri,	<u>A. Khouri</u>
Dzianis Brysiuk,	<u>Dzianis Brysiuk</u>
Christa Strauß,	<u>C. Strauß</u>
Noah Lehmann,	<u>Noah Lehmann</u>
Tina Amann,	<u>T. Amann</u>