

Bearbeitungsbeginn: 01.03.2016

Vorgelegt am: 31.08.2016

Thesis

zur Erlangung des Grades

Bachelor of Science

im Studiengang Online Medien

an der Fakultät Digitale Medien

Jennifer Jurreit

Matrikelnummer: 244136

**Konzeption und prototypische Entwicklung einer Web-
Applikation für kollaboratives Schreiben fiktiver Texte im Rol-
lenspiel-Stil**

Erstbetreuer: Prof. Wilhelm Walter

Zweitbetreuer: Prof. Dr. Dirk Eisenbiegler

Abstract

Die vorliegende Bachelor-Arbeit beschreibt die Konzeption und prototypische Entwicklung einer Web-Applikation für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil. Sie klärt dabei die Frage, wie eine Web-Applikation für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil funktions-technisch und software-technisch umgesetzt werden kann. Die Applikation soll sowohl sequentielles als auch reagierendes kollaboratives Schreiben unterstützen.

Der Prototyp dieser Applikation wurde mittels Requirements Engineering entwickelt. Bei der Analyse wurden funktionale und nicht-funktionale Anforderungen über die Definitionen für Rollenspiele (J. Arjoranta, 2011) und kollaboratives Schreiben (Paul Benjamin Lowry, Aaron Curtis und Michelle René Lowry, 2004) aufgestellt. Anschließend wurde ein Funktionsumfang der Grundfunktionen und Erweiterungen erarbeitet. Zudem wurden vorhandene und aktuell verwendete Applikationen für kollaboratives Schreiben und Schreibrollenspiele auf ihren Funktionsumfang analysiert. Durch den Vergleich der Funktionsumfänge wurde so herausgestellt, dass eine Applikation für kollaboratives Schreiben fiktiver Texte benötigt wird.

Im praktischen Teil der Thesis wurden Entwurfsmuster für die Web-Applikation erstellt. Dazu wurden Use-Case-Diagramme und -Beschreibungen angefertigt. Außerdem wurden Sequenzdiagramme der technischen Abläufe der einzelnen Use-Cases beschrieben.

Implementiert wurde die Web-Applikation ausgehend von der Anforderungsanalyse mit einem RESTful Web-Service und Web-Client. Der Web-Service basiert auf Java Spring und einer relationalen Datenbank. Der Web-Client wurde auf Basis von JavaScript, Backbone.js, jQuery, HTML 5 und CSS 3 entwickelt.

Der vorläufige Prototyp wurde auf Basis einer qualitativen und quantitativen Umfrage evaluiert. Auf Grundlage der Evaluationsergebnisse wurde der Prototyp angepasst. Ergebnis der Arbeit ist ein evaluierter und verbesserter Prototyp für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil.

Inhaltsverzeichnis

Abstract	I
Inhaltsverzeichnis	II
Abbildungsverzeichnis	IV
Tabellenverzeichnis	VI
Abkürzungsverzeichnis	VIII
1 Einleitung	1
1.1 Ausgangssituation und Problemstellung	1
1.2 Ziel der Arbeit	1
1.3 Aufbau und Methodik der Arbeit	1
1.3.1 Analyse	1
1.3.2 Entwicklung.....	2
1.3.3 Implementierung	2
1.3.4 Evaluation.....	2
2 Definitionen	2
2.1 Web-Applikationen	2
2.2 Kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil	3
2.2.1 Kollaboratives Schreiben	4
2.2.2 Rollenspiele	4
3 Schreibrollenspiel-Konzept	4
3.1 Schreibrollenspiele	4
3.1.1 Charaktere	5
3.1.2 Spielwelten	5
3.1.3 Geschichten	5
3.2 Vergleich mit interaktiver Fiktion.....	9
3.3 Vergleich mit Stift und Papier Rollenspielen.....	10
4 Analyse	12
4.1 Zielgruppe	12
4.2 Anforderungsanalyse.....	12
4.2.1 Funktionale Anforderungen	14
4.2.2 Nicht-funktionale Anforderungen	19
4.3 Funktionsumfang.....	22
4.4 Analyse vorhandener Web-Applikationen für Schreibrollenspiele	23
4.4.1 Web-Applikationen für kollaboratives Schreiben	24

4.4.2	Für Schreibrollenspiele genutzte Web-Applikationen	28
4.4.3	Für Schreibrollenspiele entwickelte Web-Applikationen	41
5	Entwurf	51
5.1	Anwendungsfälle.....	51
5.2	Sequenzdiagramme	70
5.3	Datenbankstruktur	78
6	Implementierung	84
6.1	Wahl der Programmierumgebung	85
6.1.1	Wahl der Web-Technologien: Frontend.....	85
6.1.2	Wahl der Web-Technologien: Backend	86
6.2	Implementierung	90
6.2.1	Frontend-Entwicklung.....	90
6.2.2	Backend-Entwicklung	93
6.2.3	Server-Einbettung.....	95
6.3	Ergebnis.....	95
7	Evaluation	106
7.1	Ergebnis der Evaluation	106
7.2	Verbesserungen als Resultat der Evaluation	116
8	Fazit	117
8.1	Fazit der Thesis	117
8.2	Weiterentwicklung der Software.....	118
9	Literaturverzeichnis.....	122
10	Eidesstattliche Erklärung.....	127

Abbildungsverzeichnis

Abbildung 1: Ausschnitt einer SL auf WriRo.....	7
Abbildung 2: Ausschnitt eines Random RPs auf Twitter	8
Abbildung 3: . n.d. Anteil der Befragten, die erwarten, dass eine mobile Website innerhalb von drei Sekunden geladen sein sollte (nach Ländern). Statista. Zugriff am 23. August 2016. Verfügbar unter http://de.statista.com/statistik/daten/studie/202654/umfrage/erwartungen-an-ladezeiten-von-webseiten-auf-mobiltelefonen-nach-laendern/	19
Abbildung 4: Schema einer Splitstory (Quelle: http://www.splitstory.com/de/tour)	27
Abbildung 5: Profil eines Rollenspiel-Accounts auf Twitter	29
Abbildung 6: Random RP auf Twitter	30
Abbildung 7: Ausschnitt eines RP-Beitrags auf Twitter.....	31
Abbildung 8: Twitter TL mit Threading	32
Abbildung 9: Mitteilungs-Seite eines RP-Accounts auf Twitter	32
Abbildung 10: Typische Struktur eines Forum-Rollenspiels.....	39
Abbildung 11: Beispiel für ein Forum-Rollenspiel; jeder Thread entspricht einer Geschichte	40
Abbildung 12: Aufbau eines Threads im Foren RP	40
Abbildung 13: Forum-Übersicht auf Roleplay.me	43
Abbildung 14: Charakter-Profil auf Roleplay.me.....	44
Abbildung 15: Ein Blog auf Roleplay.me.....	45
Abbildung 16: Vorstellung von Idee und Charakteren im Forum-Thread	45
Abbildung 17: Rollenspiel im gleichen Thread (siehe Abbildung 16)	46
Abbildung 18: OOC-Forum-Bereich in RolePlayGateway	48
Abbildung 19: Profilseite eines Autors auf RolePlayGateway	49
Abbildung 20: Übersichtsseite einer Geschichte auf RolePlayGateway	50
Abbildung 21: Struktur von Geschichtsbeiträgen auf RolePlayGateway	50
Abbildung 22: Use-Case-Diagramm der Client-Anwendung mit allen Grundfunktionen	52
Abbildung 23: Sequenzdiagramm zum Anlegen eines neuen Accounts	71
Abbildung 24: Sequenzdiagramm zum Login eines Nutzers.....	72
Abbildung 25: Sequenzdiagramm zum Ändern von Autor-Einstellungen	73
Abbildung 26: Sequenzdiagramm zum Erstellen eines neuen Charakters	74

Abbildung 27: Sequenzdiagramm zum Wechsel zwischen Charakteren.....	75
Abbildung 28: Sequenzdiagramm zum Erstellen eines neuen Beitrages.....	76
Abbildung 29: Sequenzdiagramm zum Editieren eines Beitrages.....	77
Abbildung 30: Sequenzdiagramm zum Löschen eines Beitrages.....	77
Abbildung 31: Sequenzdiagramm zum Abonnieren eines anderen Nutzers.....	78
Abbildung 32: Übersicht des vollständigen ER-Modells der Datenbank.....	79
Abbildung 33: ER-Diagramm der User-Tabellen.....	79
Abbildung 34: ER-Diagramm der Event-Tabellen.....	80
Abbildung 35: ER-Diagramm der Enumerations-Tabellen.....	80
Abbildung 36: ER-Diagramm der Autor-Tabellen.....	81
Abbildung 37: ER-Diagramm der Charakter-Tabellen.....	82
Abbildung 38: ER-Diagramm der Geschichts-Tabellen.....	82
Abbildung 39: Ausschnitt der Startseite von WriRo.....	96
Abbildung 40: Hauptseite des Autors auf WriRo.....	97
Abbildung 41: Eigenes Autor-Profil auf WriRo.....	97
Abbildung 42: Hauptseite eines Charakters in WriRo mit Random-Beiträgen.....	98
Abbildung 43: Hauptseite eines Charakters in WriRo mit der Geschichtsübersicht.....	98
Abbildung 44: Eigenes Charakter-Profil.....	99
Abbildung 45: Geschichtsseite einer eigenen Geschichte auf WriRo.....	99
Abbildung 46: Geschichtsseite einer abonnierten Geschichte auf WriRo.....	100
Abbildung 47: Textfeld zum Hinzufügen eines Geschichts-Beitrags auf WriRo....	100
Abbildung 48: Random-Beiträge auf WriRo.....	101
Abbildung 49: Editieren eines Beitrags auf WriRo.....	101
Abbildung 50: Suche nach einem Autor auf WriRo.....	102
Abbildung 51: Lister aller Autoren auf WriRo.....	102
Abbildung 52: Suche eines Charakters auf WriRo.....	102
Abbildung 53: Liste aller Charaktere auf WriRo.....	103
Abbildung 54: Profil eines Autors auf WriRo.....	103
Abbildung 55: Profil eines Charakters auf WriRo.....	104
Abbildung 56: Impressum von WriRo.....	104
Abbildung 57: About-Seite auf WriRo.....	105
Abbildung 58: Datenschutzrichtlinien auf WriRo.....	105
Abbildung 59: First Steps Seite auf WriRo.....	105

Tabellenverzeichnis

Tabelle 1: Funktionale und Nicht-Funktionale Anforderungen.....	14
Tabelle 2: Funktionsumfang	23
Tabelle 3: Funktionsvergleich von Web-Applikationen für kollaboratives Schreiben	24
Tabelle 4: Funktionsvergleich von für Schreibrollenspiele genutzte Web-Applikationen	

29

Tabelle 5: Funktionsvergleich von Web-Applikationen für Schreibrollenspiele	42
Tabelle 6: Use-Case 1: Autor-Account erstellen	53
Tabelle 7: Use-Case 2: Autor-Einstellungen ändern.....	54
Tabelle 8: Use-Case 3: Charakter erstellen.....	54
Tabelle 9: Use-Case 4: Charakter bearbeiten.....	55
Tabelle 10: Use-Case 5: Charakter löschen	56
Tabelle 11: Use-Case 6: Charakter auswählen	56
Tabelle 12: Use-Case 7: Charakter wechseln	56
Tabelle 13: Use-Case 8: Aus Charakter ausloggen.....	57
Tabelle 14: Use-Case 9: Nutzer abonnieren	57
Tabelle 15: Use-Case 10: Nutzer-Abonnement aufheben	58
Tabelle 16: Use-Case 11: SL zur Leseliste hinzufügen	59
Tabelle 17: Use-Case 12: SL aus Leseliste entfernen.....	59
Tabelle 18: Use-Case 13: SL erstellen.....	60
Tabelle 19: Use-Case 14: SL-Einstellungen ändern	60
Tabelle 20: Use-Case 15: SL löschen	61
Tabelle 21: Use-Case 16: Neuer Beitrag zu einer vorhandenen SL	62
Tabelle 22: Use-Case 17: SL archivieren	62
Tabelle 23: Use-Case 18: Beitrag erstellen.....	63
Tabelle 24: Use-Case 19: Beitrag editieren	63
Tabelle 25: Use-Case 20: Beitrag löschen	64
Tabelle 26: Use-Case 21: Medien-Upload.....	64
Tabelle 27: Use-Case 22: Einen anderen Nutzer erwähnen.....	65
Tabelle 28: Use-Case 23: Auf einen Beitrag antworten	66
Tabelle 29: Use-Case 24: Private Nachricht senden.....	66

Tabelle 30: Use-Case 25: Freitextsuche	67
Tabelle 31: Use-Case 26 Event anlegen	68
Tabelle 32: Use-Case 27: Event bearbeiten	68
Tabelle 33: Use-Case 28: Gäste zu einem Event einladen	69
Tabelle 34: Use-Case 29: Gäste aus der Gästeliste entfernen.....	69
Tabelle 35: Use-Case 30: Beitrag als Favorit markieren	70

Abkürzungsverzeichnis

ANSI	American National Standards Institute
API	Application Programming Interface
AU	Alternate Universe
BDSG	Bundesdatenschutzgesetz
CORS	Cross Origin Requests
CRUD	Create Read Update Delete
CSRF	Cross Site Request Forgery
CSS	Cascading Style Sheet
DB	Datenbank
DBMS	Datenbankmanagementsystem
DM	Direct Message
DOM	Document Object Model
DTO	Data Transfer Object
EJB	Enterprise Java Beans
ER-Model	Entity-Relationship-Model
GIF	Graphics Interchange Format
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IC	In Character
IF	Interactive Fiction
JDBC	Java Database Connectivity
JNDI	Java Naming and Directory Interface
JS	JavaScript
JSON	JavaScript Object Notation
JSP	Java Server Pages
JVM	Java Virtual Machine
LDAP	Lightweight Directory Access Protocol
LOC	Lines of Code
MVC	Model View Controller
NCSA	National Center for Supercomputing Applications
noSQL	Not only SQL
OC	Original Character
OOC	Out Of Character

PBKDF2	Password-Based Key Derivation Function 2
PDF	Portable Document Format
PHP	Hypertext Preprocessor
REST	Representational State Transfer
RP	Role Play
SL	Storyline
SOAP	Simple Object Access Protocol
SPA	Single Page Application
SPARC	Standards Planning and Requirements Committee
SQL	Structured Query Language
SSL	Secure Sockets Layer
TL	Timeline
URL	Uniform Resource Locator
WAR	Web Archive
WriRo	Writing Role Play
XML	Extensible Markup Language

1 Einleitung

Die vorliegende Bachelorarbeit wurde im Zuge des Online Medien Studiums durchgeführt und soll die Frage klären, wie eine Web-Applikation für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil funktions-technisch und software-technisch aufgebaut sein muss, um die Bedürfnisse der Schreibrollenspiel-Nutzer zu befriedigen. In diesem Kapitel soll eine Einleitung in das Thema gegeben werden.

1.1 Ausgangssituation und Problemstellung

Schreibrollenspiele werden häufig im Internet – auf Seiten wie Twitter, Facebook und Tumblr – ausgeführt. Anwendungen, wie Twitter, eignen sich nur teilweise für das kollaborative Schreiben fiktiver Texte im Rollenspiel-Stil. Daher lassen sich für alle Applikationen dieser Art Mängel in der Verwendung als Schreibrollenspiel-Anwendung feststellen (siehe Kapitel 4.4).

1.2 Ziel der Arbeit

Das Ziel dieser Arbeit ist die Entwicklung eines Prototyps einer Web-Applikation, die das Teilnehmen an Schreibrollenspielen ermöglicht und vereinfacht, und den Mangel an einer passenden Software (siehe Kapitel 1.1) behebt. Durch die Evaluation der Web-Applikation werden Verbesserungsmöglichkeiten herausgestellt, die später in einer Weiterentwicklung des Prototyps berücksichtigt werden können.

1.3 Aufbau und Methodik der Arbeit

Die Arbeit folgt der Methode des Requirements Engineerings aus dem Bereich der Software-Entwicklung. Diese folgt den Schritten Analyse, Entwicklung, Implementierung.

1.3.1 Analyse

Bei der Analyse wurden funktionale und nicht-funktionale Anforderungen über die Definitionen für Rollenspiele (über J. Arjoranta, 2011 und Duus Henriksen 2011) und kollaboratives Schreiben (über Paul Benjamin Lowry, Aaron Curtis und Michelle René Lowry, 2004) aufgestellt. Anschließend wurde ein Funktionsumfang der Grundfunktionen und Erweiterungen erarbeitet (siehe Kapitel 4.3). Dieser Funktionsumfang wurde dann den Funktionen bereits genutzter Web-Applikationen gegenübergestellt.

1.3.2 Entwicklung

Im praktischen Teil der Thesis, der Entwicklung, wurden Entwurfsmuster für die Web-Applikation erstellt. Dazu wurden zunächst Use-Case-Diagramme und -Beschreibungen angefertigt (siehe Kapitel 5.1). Zusätzlich wurden Sequenzdiagramme der technischen Abläufe der einzelnen Use-Cases beschrieben (siehe Kapitel 5.2).

1.3.3 Implementierung

Die Implementierung beschreibt die Programmierung der Web-Applikation und deren Einbettung in eine Server-Umgebung (siehe Kapitel 6.2). Die Programmierung erfolgte in der Programmierumgebung Eclipse und die Einbettung in eine Server-Umgebung erfolgte mit Hilfe von Apache Tomcat. Zur Entwicklung des Prototyps wurden hauptsächlich das Java Spring Framework, die Backbone.js Bibliothek und eine relationale Datenbank verwendet.

1.3.4 Evaluation

Nachdem der Prototyp fertiggestellt wurde, wurde eine Online-Umfrage zur Evaluation des Prototyps mit Hilfe der Anwendung einfacheumfrage.de durchgeführt. Bei der Evaluation wurden vierzehn Testprobanden ausgewählt, die bereits jahrelange Erfahrung mit Schreibrollenspielen hatten, die Web-Applikation zu testen. Nach Beendigung der Testphase wurden die Testprobanden gebeten einen Fragebogen, der sowohl qualitative als auch quantitative Elemente enthielt, auszufüllen. Auf Grundlage dieser Umfrage und des Testverhaltens der Probanden wurde der Prototyp evaluiert (siehe Kapitel 7).

2 Definitionen

2.1 Web-Applikationen

Mit dem Beginn des World Wide Webs Ende der 1980er Jahre (siehe Kirpal und Vogel 2006: 142) gewannen Web-Seiten zunehmend an Bedeutung. Mit der Entwicklung des NCSA Mosaic 2.0 Web-Browsers (1993) wurden die ersten Webanwendungen entwickelt (vgl. Klute 1994: 150). Heute sind zahlreiche Web-Applikationen im Internet vertreten, die unterschiedliche Technologien verwenden.

Laut Jim Conallen werden Web-Applikationen als ein System bestehend aus einem Webserver, einem Netzwerk, HTTP und Browser definiert (siehe Conallen 1999: 63). Die Geschäftsdaten dieses Systems werden durch Nutzereingaben geändert. Genauer

beschreibt er eine Web-Applikation als Softwaresystem, dessen Frontend über ein Websystem läuft, und das einen Geschäftsstatus hat (siehe Conallen 1999: 64).

Die Architektur einer Web-Applikation beschreibt Conallen als eine Art Client-Server-Struktur. Die serverseitige Komponente würde in einem Netzwerk installiert, während die clientseitige Komponente – anders als bei einem Client-Server-Modell – keine speziellen Programme oder Konfigurationen benötigen würde (siehe Conallen 1999: 64).

Marc J. Hadley von der Firma Sun Microsystems Inc. hingegen definiert eine Web-Applikation als „dynamische HTTP-basierte Applikation, dessen Interaktionen für maschinelle Bearbeitung zugänglich sind. [Anm.: das Zitat wurde aus dem Englischen übersetzt]“ (Hadley 2006: 1).

In dieser Arbeit wird der Begriff Web-Applikation nach beiden Definitionen verstanden. Eine Web-Applikation bzw. Webanwendung ist eine Anwendung, die in einem Web-Browser auf Basis von HTTP läuft, aus einem System aus Webserver, Netzwerk, HTTP, und Browser besteht, und eine zumindest Client-Server-ähnliche Struktur aufweist. Im Gegensatz zu statischen Seiten, sind Web-Applikationen dynamisch und lassen eine weite Nutzerinteraktion zu. Das heißt, über Webanwendungen können Nutzer indirekt Daten in einer Datenbank beeinflussen und somit Einfluss auf die angezeigten Inhalte der Applikation nehmen.

2.2 Kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil

In diesem Kapitel soll eine kurze Definition von kollaborativen Schreiben fiktiver Texte im Rollenspiel-Stil (auch Schreibrollenspiel) gegeben werden. Das Konzept von Schreibrollenspielen wird in Kapitel 3 erläutert.

Kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil beschreibt das gemeinsame Verfassen fiktionaler Texte durch mehrere Autoren. Dabei repräsentiert jeder Autor in seinem Beitrag zum Gesamttext jeweils einen Hauptcharakter. Die Texte werden sowohl sequentiell, d.h. nacheinander, als auch reagierend verfasst.

Das kollaborative Schreiben fiktiver Texte im Rollenspiel-Stil vereint kollaboratives Schreiben und Rollenspiele. Folgend werden diese definiert.

2.2.1 Kollaboratives Schreiben

Nach Paul Benjamin und Michelle René Lowry, und Aaron Curtis sind vier verschiedene Arten des kollaborativen Schreibens definiert; Gruppen Ein-Autor Schreiben, Sequentielles Schreiben, Paralleles Schreiben, und Reagierendes Schreiben (siehe Lowry et al. 2004: 81).

In dieser Arbeit werden ausschließlich die Modelle des Sequentiellen und Reagierenden Schreibens im Zusammenhang mit Schreibrollenspielen betrachtet. Beim sequentiellen Verfassen wird im Vorfeld eine Schreibreihenfolge festgelegt, in der die Autoren schreiben. Dabei entsteht eine lineare Textstruktur. Beim reagierenden Schreiben hat jeder Autor die Möglichkeit auf jeden vorherigen Beitrag zu antworten. Dadurch entsteht eine Baumstruktur mit mehreren Geschichtsverläufen (vgl. Lowry et al. 2004: 81).

2.2.2 Rollenspiele

Der Begriff Rollenspiel wird in vielen Zusammenhängen definiert und angewendet. Unterschieden werden Definitionen im Zusammenhang mit Spielen und Definitionen im Zusammenhang mit Aktivitäten (siehe Arjoranta 2011: 3). Diese Definitionen finden Anwendung in der Psychologie, Soziologie, und Kommunikations- und Computerwissenschaften (vgl. Arjoranta 2011: 3-4). In dieser Arbeit wird der Begriff Rollenspiel aus der Perspektive der Kommunikations- und Computerwissenschaften betrachtet. Er beschreibt grundsätzlich das Repräsentieren und Imitieren einer Rolle in einer fiktionalen Spielwelt (vgl. Duus Henriksen 2011: 176).

3 Schreibrollenspiel-Konzept

Nachdem in Kapitel 2.2 eine kurze Definition von Schreibrollenspielen gegeben wurde, soll in diesem Kapitel das Konzept des Schreibrollenspiels beschrieben werden.

3.1 Schreibrollenspiele

Der Begriff Schreibrollenspiel beschreibt das Verfassen eines kollaborativen, fiktionalen Textes, bei dem jeder Autor eine Rolle bzw. einen Charakter repräsentiert (vgl. Kapitel 2.2).

Schreibrollenspiele sind eine Form des Menschen-zu-Menschen interaktiven Geschichtenerzählens (engl. Interactive Storytelling) (vgl. Berger und Marbach 2008: 330).

Die Hauptelemente des Schreibrollenspiels sind: Charaktere, Spielwelten, und Geschichten.

3.1.1 Charaktere

Charaktere sind die Protagonisten und Antagonisten der zu schreibenden Geschichte, d.h. jeder Charakter ist ein Hauptcharakter der verfassten Geschichte.

Jeder Autor bzw. Nutzer kann beliebig viele Charaktere erstellen, die er in einem Profil beschreibt. In diesem Profil sind die wichtigsten Merkmale dieses Charakters festgehalten. Diese sind: Name, Aussehen, Alter, Rasse, eine kurze Beschreibung der Persönlichkeit des Charakters, und die zugehörige Spielwelt.

Charaktere können vom Autor frei erfunden sein, sogenannte Originale Charaktere (OC; engl. Original Character), oder einem Buch, einer Serie, oder einem Film entstammen.

Alle erstellten Charaktere stehen dem zugehörigen Autor jederzeit zur Verfügung und können in Geschichten verwendet werden. Sie sind der Grundbaustein jeder Geschichte.

3.1.2 Spielwelten

Die Spielwelten beschreiben die Umgebung der Charaktere, d.h. die Welt, in der die Charaktere agieren. Dies können die reale Welt, oder fiktionale Welten sein.

Die Spielwelten werden sowohl jedem Charakter, als auch jeder Geschichte zugeordnet.

Jeder Charakter kann einer oder mehr Spielwelten zugehören, während jede Geschichte nur einer, bestimmten Spielwelt zugehört.

Fiktionale Welten können ebenfalls von den Autoren frei erfunden werden oder einem Buch, einer Serie, oder einem Film entstammen. In jedem Fall werden die Spielwelten innerhalb der Geschichten erläutert.

3.1.3 Geschichten

In Schreibrollenspielen unterscheidet man drei verschiedene Modelle von Geschichten, Handlungsstränge (engl. storylines; SLs), Solos und die Ungeordneten Rollenspiele (engl. random role plays; Random RP).

Als SLs werden die Geschichten bzw. Teilgeschichten bezeichnet, die dem Prinzip des sequentiellen kollaborativen Schreibens folgen (siehe Abbildung 1).

Sie werden im Vorfeld von den mitwirkenden Autoren geplant und anschließend unter einem bestimmten Titel verfasst. Zur Planung gehören stets die Festlegung der mitwirkenden Charaktere, die Bestimmung der Schreibreihenfolge, eine grobe Zusammenfassung der zu schreibenden Geschichte, und der Titel der Geschichte. Desweiterem können klassische Textentwürfe, wie ein Spannungsbogen, eine Zeitachse, und wichtige Schlüsselszenen im Vorfeld ausgearbeitet werden.

Solos sind, wie der Name suggeriert, einzeln verfasste Geschichten. Sie werden ebenfalls unter einem bestimmten Titel verfasst und dienen lediglich der Vorstellung eines Charakters. Diese Geschichten enthalten die Hintergrundgeschichte eines Charakters, oder Monologe, die zur Charakterentwicklung beitragen. Solos bilden eine Basis, bzw. eine Erweiterung, zu SLs und Random RPs.

Random RPs sind ungeordnete Geschichten, die durch die Interaktion von Charakteren entstehen (siehe Abbildung 2). Sie werden reagierend kollaborativ verfasst. Autoren haben die Möglichkeit auch außerhalb einer SL oder eines Solos einen Charakter zu repräsentieren. Dies geschieht durch Kommentare, bzw. kurzen Szenen, im Charakter-Profil. Interaktion entsteht dann, wenn ein anderer Charakter auf diese Szene mit einer eigenen kurzen Szene reagiert. Auf diese Weise entstehen zufällige, nicht geplante Geschichten. Die so entstehenden Kurzgeschichten haben keinen Titel, keine vorgegebene Schreibreihenfolge, keinen festgelegten Inhalt, oder vorher festgelegte Charakter-Beteiligung – d.h. es können beliebig viele Charaktere jederzeit dazu stoßen – und dienen der Unterhaltung.

Andere Länder, andere Leute

Description:

Lea Fuchs, eine Austauschstudentin aus Österreich, trifft an ihrem dritten Tag in Whitmore, Virginia, eine junge Frau in der Bar. Wird sie ihre erste Freundin in den USA? Amy Miles, eine Hexe auf der Suche nach jungen Gleichgesinnten, begibt sich nach einem langem Tag in eine Bar. Dort trifft sie eine junge Frau. Irgendetwas scheint seltsam. Ist sie womöglich eine Hexe?

Characters:

Lea Fuchs (FoxTrickery)

Amy Miles (AmyMiles)

Fandom:
Original
Story

Status:
ONGOING

Age
Restriction:
0

Unsubscribe

Download as PDF

Story

Amy Miles (@AmyMiles)

Ich hatte mich gerade mit so einem dummen Hexer rumschlagen müssen, der es noch nicht einmal schaffte sich nicht selbst zu verletzen. Ich hatte also den Großteil meines Tages damit zugebracht ihn quer durch die Stadt zu jagen, bis dass ich ihn endlich in einem Lagerhaus stellen konnte. Das Problem mit solchen Leuten ist ganz einfach: Sie gehen mir auf die Nerven. Der Gipfel des Tagesgeschehens bestand darin, einem seiner versuchten Zauber auszuweichen. Nachdem er endlich sicher von meinen Kollegen zur Akademie gebracht worden ist, wo er lernen wird wie man zaubert ohne Land und Leute zu verletzen, es sei denn man möchte es, kann ich nun endlich für heute Feierabend machen. Die Bar in der Nähe sieht sehr vielversprechend aus auch wenn ich hier noch nie war, werde ich sie einfach mal testen. "Tequilla gold, bitte," schreie ich schon fast dem Barkeeper über die laute Musik hinweg an. Ich lasse mich in einer der kleinen Buchten mit Bänken nieder und beobachte die Menschen um mich herum. Manchmal wünsche ich mir mein Job würde es nicht erfordern, dass ich immer in Alarmbereitschaft bin. Naja, der Alkohol hilft definiiv dabei das Gefühl der Anspannung etwas zu mindern.

Lea Fuchs (@FoxTrickery)

Vor zwei Tagen bin ich endlich in Amerika gelandet - nach einer zweimonatigen Planungsphase für mein Auslandssemester, Visumsbeantragung, Unibewerbungen und Unterkunftssuche. Virginia, der Bundesstaat im Osten des Kontinents mit der Hauptstadt Richmond. Eigentlich wollte ich ja lieber nach Kalifornien, aber mein Budget lies dies nicht zu. So wurde es schlussendlich nur eine kleine Universität in Whitmore. Bevor das Semester richtig los geht und ich aus den College Gebäuden nicht mehr rauskomme, habe ich geplant mir ein paar Tage Zeit zu nehmen, um mir die Stadt anzusehen. Daher bin ich am Donnerstag gelandet - nach einem zehnstündigen Flug, zwei Wochen vor dem eigentlichen Studienbeginn. Bei der Sicherheitskontrolle hätten sie mich ja fast gar nicht erst in den Flieger gelassen, hatte ich doch glatt vergessen mein Taschenmesser daheim zu lassen. Nach einer längeren, nervenaufreibenden Unterhaltung mit dem doch recht verärgerten Sicherheitspersonal und einer Vereinbarung ihnen das Messer zu überlassen, konnte ich letztlich doch in das Flugzeug einsteigen. Die ersten Tage waren der Horror. Schon im Taxi vom Flughafen zum Hotel - mein Zimmer in der Studentenunterkunft wird ja erst in zwei Wochen frei - bin ich fast eingeschlafen. Endlich in meinem Hotelzimmer, einem kleinen Raum mit winzigem Fenster, einem ebenso winzigen Bad mit Dusche - für eine Wanne war dort kein Platz - einen Schreibtisch aus Buche geschnitzt mit Stuhl aus demselben Holz, und einem Doppelbett, angekommen fiel ich direkt ins Bett. Anderthalb Tage habe ich durchgeschlafen. Jetzt, bin ich endlich bereit mir die Gegend anzusehen. Grinsend betrachte ich mein Spiegelbild in dem ovalen Spiegel an der gefliesten Wand im Badezimmer. In dem dunklen Licht der Glühbirne wirken meine roten Haare fast braun, die smaragd-grüne Bluse beinahe oliv-grün und die weiße Jeans schimmert mir gelblich entgegen. Ein letztes Mal streiche ich mit der Hand durchs Haar, um die wilden Locken vielleicht doch noch zu bändigen, bevor ich aus dem Bad stolpere, die schwarze Lederjacke von ihrem Haken reiße und mir über die Schulter hänge, und aus dem Zimmer gehe. Auf dem Weg durch das Treppenhaus in den ersten Stock, stopfe ich die Schlüsselkarte in die hintere Tasche meiner Hose und checke mein Handy. Keine neuen Nachrichten. Kein Wunder, meine Freunde in Österreich schlafen alle bereits um diese Zeit. Immerhin ist es dort jetzt drei Uhr morgens. Hier in Whitmore ist die Nacht noch jung und irgendwie kühl. Sobald ich aus dem Hotel komme, weht mir ein kühler Wind entgegen, der mir eine Gänsehaut auf die Arme treibt. Fröstelnd greife ich sofort nach meiner Jacke und schlüpfe hinein, während ich mit der freien Hand versuche die nächste Bar zu googlen. Wo kann man besser Studenten kennen lernen als in einer Bar? So mal ja Semesterferien sind. Fünfhundert Meter, direkt um die nächste Ecke, liefert das Handy schließlich zurück. "Wirklich nicht sehr weit," schmunzle ich, locker los spazierend. Beinahe laufe ich an der Bar vorbei, obwohl sie nun wirklich nicht unscheinbar ist. Ein großes blinkendes Schild mit dem Titel "Hell's Kitchen" weist das Gebäude aus. Froh, nun endlich aus der Kälte zu kommen, schlüpfe ich durch die Tür. Sofort hallt mir laute Rock-Musik entgegen. Rock der Achtziger. Nicht schlecht. Die Bar ist nicht so voll wie ich erhofft hatte, aber ein paar junge Leute sind schon da. "Vielleicht werden es später mehr. Es ist ja noch früh am Abend," murmele ich vor mich hin, als ich mir einen Weg zur Bar bahne. "One shot of tequila, please," begrüße ich den Barkeeper mit einem freundlichen Lächeln. "I need to see your ID," kommt es von dem Mann Ende dreißig, mit kurzen, schwarzen Haaren und Ziegenbart, zurück. Dass ich nicht von hier bin, hat er wohl an meinem Akzent bemerkt. Mühselig fummle ich meinen Ausweis aus der Tasche und zeige ihm den, als Beweis, dass ich bereits einundzwanzig bin. Dann bekomme ich auch endlich meinen Tequila. Den ersten Shot trinke ich direkt. Mit dem zweiten in der Hand, lasse ich meinen Blick durch die Menge schweifen. Da vorne ist noch ein Tisch frei. Bisher sitzt nur eine junge Frau - könnte in meinem Alter sein - dort. Ein paar Minuten später, stehe ich vor ihrem Tisch in der kleinen Bucht. "Do you mind..." frage ich, auf den Platz neben ihr deutend.

Amy Miles (@AmyMiles)

"No, please take a seat," antworte ich der jungen Frau, die durchaus in meinem Alter sein könnte. Ich betrachte sie erst einmal etwas skeptisch, aber, nachdem ich bei ihr keine magischen Fähigkeiten spüren kann, ringe ich mich zu einem Lächeln durch. Das letzte, was ich heute Abend gebrauchen kann, ist ein Tracker von der schwarzen Seite. Der Hexer heute war vielleicht leicht zu bekommen, aber nach meiner Verletzung vor einem halben Jahr hatte er es doch geschafft meine magischen Ressourcen bis aufs Äußerste zu strapazieren. Alles was ich heute Abend will ist schlafen, essen, duschen und trinken. Naja, vielleicht nicht unbedingt in dieser Reihenfolge. Ich habe mir diese Bar ausgesucht, da sie vielversprechend wirkt was Privatsphäre angeht. Mit meiner blauen used-look

Abbildung 1: Ausschnitt einer SL auf WriRo

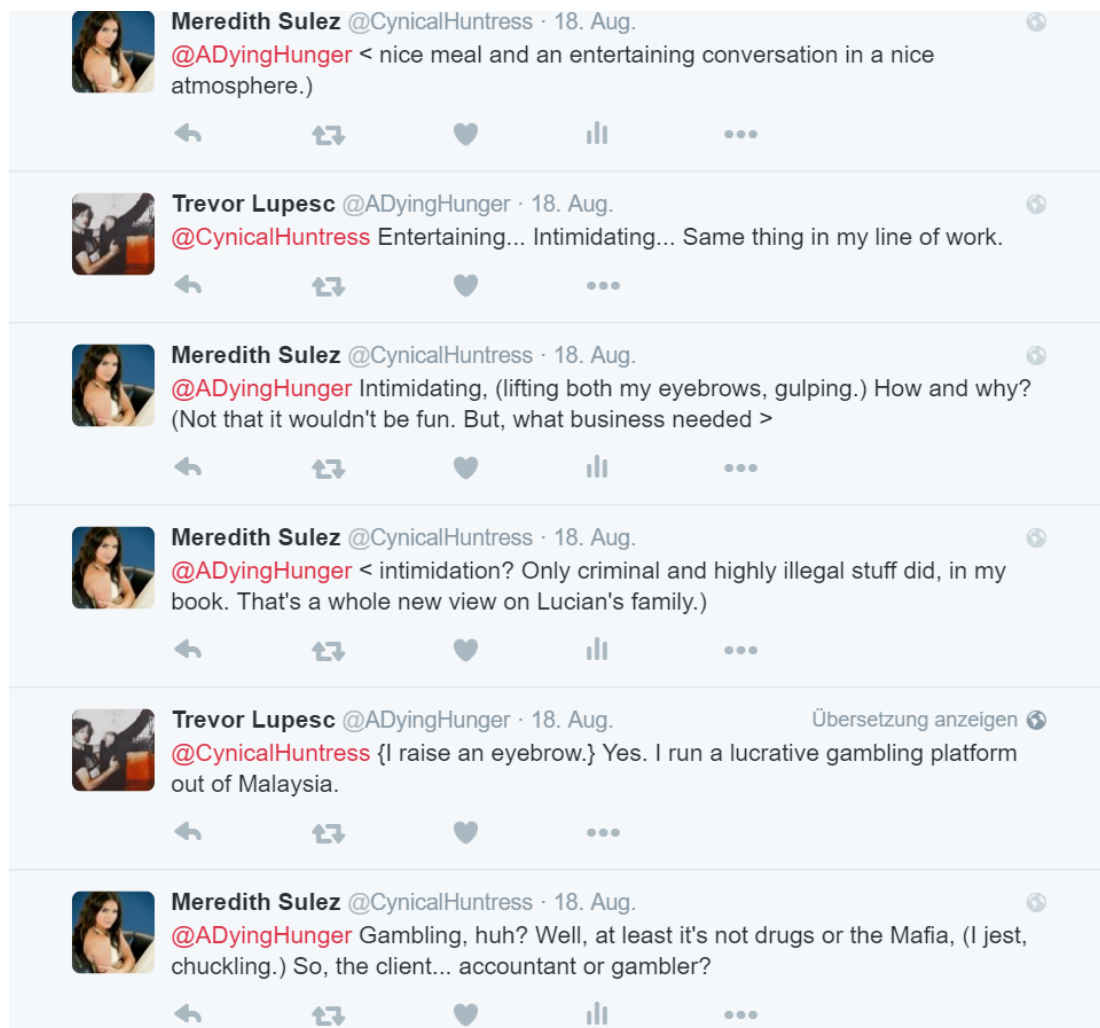


Abbildung 2: Ausschnitt eines Random RPs auf Twitter

Zusätzlich unterscheidet man zwei verschiedene Geschichts-Arten, Fanfiktion und Originale Geschichten (engl. Original Stories).

Original Stories, sind Geschichten, die von den mitwirkenden Autoren eigenständig entwickelt wurden. Eine Geschichte zählt nur dann zu den Original Stories, wenn alle drei Elemente – Charaktere, Spielwelt, und Geschichte – original sind, d.h. von den mitwirkenden Autoren selbstständig entwickelt wurden.

Fanfiktion sind Geschichten, die auf bekannte Geschichten aus Büchern, Serien, oder Filmen basieren, oder Charaktere aus diesen enthalten.

Unabhängig der Art und des Modells der Geschichte, wird diese stets aus den Perspektiven der Hauptcharaktere geschrieben. Das heißt, jeder mitwirkende Autor erzählt die Geschichte aus der Perspektive seines Charakters. Die Geschichte wird in der Ich-Perspektive, oder von einem personalen Erzähler erzählt.¹ Folglich wechselt der Erzähler nach jedem Beitrag ab. Beiträge in SLs sind in der Regel 100 bis 1500 Wörter lang, während Beiträge in Solos 400 bis 3000 Wörter umfassen, und Beiträge in Random RPs meist zwischen 10 und 200 Wörter lang sind.

3.2 Vergleich mit interaktiver Fiktion

Interaktive Fiktion (engl. Interactive Fiction; abgekürzt IF), auch Textadventure genannt, beschreibt Computerspiele, bei denen die Interaktion zwischen Spiel und Spieler ausschließlich oder hauptsächlich über Text stattfindet (vgl. Gordon und Swanson 2008: 32, vgl. auch Montfort 2009: 55, vgl. auch Slator et al. 2007: 121).

In diesen Spielen simuliert der Computer eine fiktionale Welt über textuelle Beschreibungen oder kleinen Grafiken. Diese Welten beschreiben meisten einen fiktionalen Ort, wie z.B. einen Kerker, einen verwunschenen Wald, oder einen realen Ort, wie z.B. New York City (vgl. Montfort 2009: 55, vgl. auch Slator et al. 2007: 121). Über Textbefehle kann der Spieler die Spielfigur durch diese Welt steuern. Diese Textbefehle sind kurze narrative Texte, wie z.B. „geh weiter“, „Richtung Nordwest“, „hebe die Schaufel auf“, „töte Drachen mit Schwert“ (vgl. Montfort 2009: 55, vgl. auch Slator et al. 2007: 121).

Der Computer nimmt diese Befehle entgegen, wertet sie aus, und reagiert auf die Benutzereingaben indem er mit einem neuen Teil der Geschichte antwortet.

Bekannte Spiele aus diesem Genre sind Adventure (1976) und Zork (1977) (vgl. Montfort 2009: 55).

Sowohl interaktive Fiktion als auch Schreibrollenspiele bestehen aus den Elementen Charaktere, Spielwelt, und Geschichte. Außerdem wird bei beiden durch Interaktion der Nutzer in Form von Text eine Geschichte erzeugt. Dennoch lassen sich Schreibrollenspiele deutlich von interaktiver Fiktion abgrenzen.

Interaktive Fiktion, oder Textadventures, sind Computerspiele. Ziel des Spiels ist das Lösen einer bestimmten Aufgabe, nicht aber die Erschaffung eines narrativen Textes.

¹ Der personale Erzähler erzählt eine Geschichte aus der dritten Person („er“, „sie“, „es“), allerdings nur aus dessen Perspektive; er ist nicht allwissend.

Daher sind die verwendeten Textstücke sehr kurz und beinhalten keine stilistischen Mittel. Zudem findet die Interaktion ausschließlich zwischen dem Nutzer und dem Computersystem statt, nicht aber zwischen mehreren Nutzern. Außerdem sind alle Charaktere der Geschichte vom Entwickler des Spiels festgelegt worden. Der Nutzer selber kann keine Charaktere hinzufügen oder zwischen mehreren Charakteren wählen.

Insgesamt lässt sich feststellen, dass Schreibrollenspiele zwar Ähnlichkeiten mit interaktiver Fiktion aufweisen, aber keineswegs als interaktive Fiktion bezeichnet werden können.

3.3 Vergleich mit Stift und Papier Rollenspielen

Stift und Papier Rollenspiele (engl. Pen and Paper Role Playing Games) sind eine Form des ältesten Rollenspiels. Die ersten bekannten Rollenspiele wurden 1971 (Blackmoor von Dave Arneson) und 1974 (Dungeons & Dragons (D&D) von Gary Gygax) entwickelt (vgl. Jenderek: 316).

Zum Spielen wurde ausschließlich Stift und Papier benötigt, weshalb diese Spiele unter dem Namen „Stift und Papier Rollenspiele“ bekannt wurden (siehe Jenderek: 316).

Laut Bastian Jenderek besteht das Spiel immer aus den folgenden Elementen (vgl. Jenderek: 316-317):

1. Spielleiter:

Der Spielleiter übernimmt im Stift und Papier Rollenspiel die Rolle des Computers in Interaktiven Fiktionen. Er beschreibt die Spielwelt und die Ausgangssituation der Charaktere, wertet die Spieler-Eingaben aus, und reagiert auf diese mit einer entsprechenden Fortsetzung der Geschichte bzw. einer Änderung der Spielwelt.

2. Spieler:

Die Spieler interagieren über fiktionale Charaktere mit der Spielwelt. Dazu müssen sie zunächst einen Charakter erschaffen. Als erstes entscheidet sich der Spieler für eine der Rollen, die im Regelwerk des Spiels festgelegt wurden, z.B. Krieger, Magier, Schurke, Mönch. Dann kann der Spieler seinen Charakter durch die Zuweisung von bestimmten Eigenschaften individualisieren. Diese Eigenschaften beinhalten die Rasse des Charakters, sowie körperliche und geistige Fähigkeiten.

Um mit der Spielwelt zu interagieren formulieren die Spieler Handlungen und Dialoge, die sie auf Papier festhalten und den Mitspielern vortragen. Über diese Handlungen und Dialoge beeinflussen die Spieler die Spiel-Welt. Wichtige Handlungen, wie das Besiegen eines Gegners, werden stets belohnt. Als Belohnung erhalten die Spieler, z.B. nützliche Gegenstände, die im weiteren Verlauf der Geschichte benötigt werden.

3. Zufallselement:

Das Zufallselement ist ein wichtiges Element des Spiels, welches willkürliches Verhalten der Spieler und willkürliche Entscheidungen des Spielleiters verhindert. Meistens wird dieses Zufallselement in Form von Würfeln eingebaut. Die Würfel entscheiden dann über den Erfolg oder Misserfolg einer Handlung, bzw. über den weiteren Verlauf der Geschichte.

4. Charakterentwicklung:

Im Laufe des Spiels sammelt der Charakter Erfahrungspunkte, z.B. durch das Lösen einer Aufgabe. Diese dienen der Verbesserung des Charakters, er wird stärker, reicher, oder erhält bessere Ausrüstungen.

5. Kommunikation:

Während des Spiels herrscht eine ständige Kommunikation zwischen den Spielern. Diese muss stattfinden, um bestimmte, schwierige Aufgaben in Kooperation mit einander zu lösen, oder die Handlungen eines Charakters nachzuverfolgen.

Im Gegensatz zu weitverbreiteten Theorien im Internet (z.B. auf Wikipedia), handelt es sich bei Schreibrollenspielen nicht um eine digitale Erweiterung von Stift und Papier Rollenspielen, bei denen lediglich auf das Zufallselement und Erfahrungspunkte verzichtet wird. Es ist festzustellen, dass Schreibrollenspiele zwar Elemente des Stift und Papier Rollenspiels enthalten, aber keineswegs als eine Art des digitalen Stift und Papier Rollenspiels betrachtet werden können. Zwar wird bei beiden eine fiktionale Welt durch die „Spieler“ über fiktionale Charaktere durchlaufen und verändert, so handelt es sich aber bei dem Stift und Papier Rollenspiel um ein Spiel mit Regeln, einem Spielleiter, Punkten, und dem Ziel eine bestimmte Aufgabe zu lösen und das Spiel zu beenden, während es sich bei Schreibrollenspielen um das kollaborative Schreiben einer fiktionalen Geschichte handelt.

4 Analyse

4.1 Zielgruppe

Die Web-Applikation WriRo (Writing Role Play) richtet sich an junge Menschen, zwischen 14 und 40 Jahren, die Interesse am Schreiben und Lesen haben. Jeder, der eine Geschichte liest oder in Film und Fernsehen sieht und gerne in die Rolle eines Charakters dieser Geschichte schlüpfen möchte, findet in WriRo eine Möglichkeit dieses Interesse auszuleben. Schreibrollenspiele bieten eine Alternative zu weitverbreiteten Fanfiktionen, den Fortsetzungen einer bekannten Geschichte durch ihre Fans.

Die Applikation richtet sich auch an Nutzer, die Freude am Schreiben eigener fiktionaler Geschichten haben und diese Freude gerne mit anderen Nutzern teilen.

Vor allem Personen, die bereits an Schreibrollenspielen mit Hilfe einer der im Kapitel 4.4 erläuterten Anwendungen teilnehmen, gehören zur Zielgruppe von WriRo.

4.2 Anforderungsanalyse

Die Anforderungsanalyse erfolgt auf Basis des Konzepts von Schreibrollenspielen, das in Kapitel 3 ausführlich diskutiert wurde, und dient als Grundlage für den Funktionsumfang.

Die folgende Tabelle liefert einen Überblick über die funktionalen und nicht-funktionalen Anforderungen, die an die Web-Anwendung gestellt werden.

Funktionale Anforderungen	Nicht-funktionale Anforderungen
Autor-Account erstellen	Schnelligkeit: Das Laden von Daten aus der DB darf nicht länger als 3 sek. dauern
Autor-Account verwalten	Benutzerfreundlich (leicht zu bedienen): Die Benutzeroberfläche muss übersichtlich gestaltet sein. Die Applikation muss unter Verwendung des responsiven Webdesigns entwickelt werden. Die Applikation muss sowohl in deutscher als auch englischer Sprache vorliegen. Bei Fehlern muss das System eindeutige und verständliche Fehlermeldungen geben.
Charakter erstellen	Zuverlässigkeit: Das System muss zu 80% ausfallsicher sein. Es dürfen keine Daten verloren gehen.
Charakter bearbeiten	Sicherheit: Heutige Sicherheit-Standards müssen erfüllt werden
Charakter löschen	Evaluation
Charakter auswählen / wechseln	Datenschutz: Die Applikation muss dem deutschen Datenschutz genügen.
Nutzer abonnieren / nicht abonnieren	Korrektheit: Maximal 10 Fehler pro 1000 LOC (Prototyp) Maximal 4 Fehler pro 1000 LOC (fertige Applikation)
Nutzer blockieren	Kompatibilität: Mit allen modernen Browsern kompatibel.
Geschichte erstellen	
Geschichten-Profil bearbeiten	
Geschichte bearbeiten	
Geschichte löschen	
Geschichte archivieren	
Beitrag erstellen	

Beitrag bearbeiten	
Beitrag löschen	
Geschichte abonnieren / nicht abonnieren	
Beitrag favorisieren / nicht favorisieren	
Medien-Upload	
Nutzer erwähnen	
Private Nachrichten senden / empfangen	
Auf Beiträge antworten	
Freitextsuche / Suche	
Events anlegen	
Events verwalten	
Notifikationen	
Mobilfähig	
Alle Daten müssen in der DB gespeichert werden	
Das System muss zwischen Charakter- und Autoreneingaben unterscheiden	
Das System muss Rechte verschiedener Charaktere / Autoren verwalten	

Tabelle 1: Funktionale und Nicht-Funktionale Anforderungen

4.2.1 Funktionale Anforderungen

In diesem Kapitel werden die funktionalen Anforderungen an die Web-Applikation erläutert.

4.2.1.1 Autor-Account

Zur Verwaltung von Charakteren und Nutzer-Informationen und -Interaktionen wird ein Autor-Account benötigt. Der Autor-Account ist der Account mit dem sich die Nutzer in die Anwendung einloggen. Von dort aus können die Nutzereinstellungen verwaltet werden, Charaktere erstellt und verwaltet werden, und mit anderen Nutzern interagiert

werden. Auf der Hauptseite des Autor-Accounts können Statusmeldungen hinterlegt werden.

4.2.1.2 Charakter-Account

Wie bereits in Kapitel 3.1.1 erläutert sind Charaktere ein wichtiger Bestandteil von Schreibrollenspielen. Charakter-Accounts gehören daher zu den essentiellen funktionalen Anforderungen der Web-Applikation. In Charakter-Accounts werden Charaktere verwaltet. Über den Charakter-Account findet die Interaktion mit anderen Charakteren statt und es können Geschichten erstellt werden. Außerdem können Metadaten des Charakters im Charakter-Profil beschrieben werden. Zum Beispiel können Charaktere miteinander verknüpft werden, die miteinander in Beziehung stehen (z.B. Ehepaare, Partner, Geschwister, etc.).

4.2.1.3 Geschichten

Das wichtigste Element von Schreibrollenspielen sind die Geschichten (vgl. Kapitel 3.1.3).

Funktionale Anforderungen im Bereich der Geschichten sind die Erstellung von Geschichts-Profilseiten, die Verwaltung des Geschichts-Profiles – bearbeiten und löschen – und die Möglichkeit die Geschichte zu abonnieren oder in Form eines PDF-Dokuments herunterzuladen.

Die Geschichts-Profilseite dient der Übersicht über die Geschichte, dem Schreiben der Geschichte, und dem Lesen der Geschichte, sowie dem Vorstellen einer Geschichts-idee. Die Profilseite muss folgende Elemente beinhalten: Titel, Kurzbeschreibung, Charakterliste, Genre, Altersbeschränkung, und Metadaten (Zeitstrahl, Spannungsbogen, Schlüsselszenen, etc.).

Zudem können die mitwirkenden Charaktere auf der Geschichtsseite an der Geschichte teilnehmen und Beiträge für diese verfassen. Nicht-mitwirkende Charaktere können die Geschichte auf dieser Seite abonnieren. Dadurch erscheint diese in der Leseliste des jeweiligen Charakters. Durch den Status **Searching** kann der Verfasser der Geschichte angeben, dass er weitere Charaktere für diese Geschichte sucht.

Außerdem kann die Geschichte über die Profilseite von allen Nutzern archiviert, d.h. als PDF-Dokument heruntergeladen werden.

Der Download bietet die Möglichkeit zur Offline-Speicherung und zum Versenden der Geschichte. So können die Geschichten auch mit Personen, die sich nicht in der Anwendung registriert haben, geteilt werden.

4.2.1.4 Gruppen-Seiten

Gruppen-Seiten dienen der Zuteilung von Charakteren in Gruppen, z.B. Familien. Auf den Gruppen-Seiten kann ein Profil angelegt werden, in dem die Gruppe – ähnlich wie einzelne Charaktere – beschrieben werden kann. Hier wird eine Übersicht über die Mitglieder der Gruppe, sowie eine kurze Beschreibung der Gruppe hinterlegt. Mögliche Metadaten, wie zum Beispiel ein Stammbaum der Familie, können ebenfalls im Profil der Gruppe angelegt werden.

Diese Gruppen-Seiten dienen der näheren Übersicht über Charakter-Zugehörigkeiten und -Verknüpfungen. Sie dienen der ausführlichen Information von Charakter-Geflechten zur Ergänzung zu den in Geschichten gegebenen Informationen.

4.2.1.5 Beiträge

Das Verfassen, Löschen, Bearbeiten, und Beantworten von Beiträgen gehört ebenfalls zu den funktionalen Anforderungen.

Beiträge dienen der Kommunikation zwischen Nutzern und sind für die Organisation von Geschichten notwendig. Über öffentliche Beiträge, die für alle registrierten Nutzer sichtbar sind, erfolgen Statusmeldungen, wie z.B. „Ich bin für 3 Wochen im Urlaub. Alle Geschichten müssen solange pausieren.“, oder Random RP (siehe Kapitel 3.1.3).

4.2.1.6 Nutzer-Interaktion

Die Interaktion zwischen Nutzern ist ebenfalls eine funktionale Anforderung an die Web-Applikation. Sie beschreibt sowohl das Blockieren eines Nutzers, als auch das Versenden von privaten Nachrichten.

Die Möglichkeit des Blockierens eines Nutzers trägt zur Selbstverwaltung des Systems bei. Konflikte zwischen Nutzern können so von den Nutzern selber gelöst werden. Geblockte Nutzer können keine Beiträge, Charaktere oder Geschichten des blockierenden Nutzers einsehen. Auf diese Weise können Nutzer unerwünschte Nutzer von ihren Seiten aussperren.

Das Versenden privater Nachrichten ist ebenfalls eine wichtige funktionale Anforderung. Durch private Nachrichten können Nutzer in Kontakt treten, um Geschichten oder

Charaktere zu planen, ohne ihre Planungen dabei für alle Nutzer offen zu legen. Auf diese Weise können Planungen vorgenommen werden, ohne den Lesern den Inhalt der Geschichte vorzeitig zu verraten.

Außerdem können private Nachrichten zur Konfliktlösung zwischen Nutzern beitragen. Streitigkeiten können in privater Atmosphäre ausgetragen und beseitigt werden, ohne dass sich die Beteiligten in der Öffentlichkeit zur Schau stellen.

4.2.1.7 Berechtigungen

Die Nutzer sollen die Möglichkeit haben individuelle Berechtigungen für Accounts, Geschichten, und Beiträge zu vergeben. Durch die Vergabe dieser Berechtigungen sollen die Nutzer die Möglichkeit haben zu bestimmen wer (einzelne Personen oder Benutzergruppen) entsprechende Inhalte lesen, bearbeiten, speichern und löschen darf.

Dies dient ebenfalls der Selbstverwaltung des Systems, sowie der Selbstbestimmung über die Inhalte der Web-Applikation durch den Nutzer.

4.2.1.8 Altersbeschränkung

Eine besondere Art der Berechtigungen bildet die Altersbeschränkung eines Accounts, oder einer Geschichte. Der Nutzer soll die Möglichkeit haben eine Altersbeschränkung für seine Profile (Autor, Charaktere, Geschichten) anzugeben. Das System soll dann prüfen ob der Nutzer, der auf diese Seiten zugreifen möchte, dieser Altersbeschränkung entspricht und zu junge Nutzer blockieren.

Auf diese Weise kann der Jugendschutz auf WriRo garantiert werden.

4.2.1.9 Suche

Die Suche beinhaltet sowohl eine Freitextsuche, als auch filterbasierte Suchen, und die Auflistung aller Nutzer. Die Suche ist essentiell um Nutzer, und insbesondere Charaktere und Geschichten in der Anwendung zu finden.

4.2.1.10 Events

Events stellen größere SLs oder Random RPs dar, an denen in der Regel mindestens zehn Charaktere teilnehmen. Typischerweise folgen diese Events einem festlichen Thema, z.B. einer Hochzeit, einer Geburtstagsfeier, einer Silvester-Feier, oder einem Weihnachtsfest.

Zur Planung und Ausführung von Events innerhalb der Web-Anwendung soll eine Events-Kategorie geschaffen werden. In dieser kann ein Event-Profil, und eine Gästelis-

te erstellt werden. Das Event-Profil dient der Übersicht über das Event mit allgemeinen Informationen, wie „Worum geht es bei diesem Event“, einem Terminkalender und Angaben über den Ort des Events und eventuellen Dresscodes und Besonderheiten.

Zudem können über das Event-Profil Einladungen zum Event versendet werden.

Die Gästeliste dient der Übersicht und Verwaltung der Teilnehmer an diesem Event.

4.2.1.11 Medien-Upload

Der Medien-Upload soll dem Hinzufügen von Grafiken in Profilen und Beiträgen dienen. Visuelle Grafiken, z.B. Bilder von Charakteren oder Karten der virtuellen Welt, dienen der Ergänzung von Beschreibungen und Geschichten.

4.2.1.12 Notifikationen

Notifikationen in Form von Email- und Browser-Notifikationen dienen der Übersicht und Information des Nutzers. Durch Notifikationen wird er über Änderungen, z.B. neuen Beiträgen in SLs, privaten Nachrichten, oder neuen Benachrichtigungen informiert.

4.2.1.13 Mobilität

Die Web-Applikation soll mobil-fähig sein, damit die Nutzer sie auch Unterwegs über Smartphone oder Tablet nutzen können. Dazu soll es mindestens eine responsive Version der Web-Anwendung geben.

4.2.1.14 Datenverwaltung

Alle Daten der Web-Applikation – Nutzerdaten, Beiträge, interne Daten – sollen in entsprechenden Tabellen einer relationalen Datenbank gespeichert werden.

4.2.1.15 Systemeigenschaften

Das System muss zwischen einem Autor- und einem Charakter-Bereich unterscheiden können, damit der Nutzer die Möglichkeit hat sowohl als Autor, als auch als Charakter mit der Anwendung zu interagieren.

Außerdem muss das System Rechte des Nutzers verwalten. Bestimmte Funktionen und Bereiche der Anwendung können nur von bestimmten Nutzern verwendet werden. So können z.B. nur mitwirkende Charaktere Beiträge zu einer Geschichte formulieren oder das Geschichtsprofil ändern.

4.2.2 Nicht-funktionale Anforderungen

Neben den funktionalen Anforderungen, gibt es auch einige nicht-funktionale Anforderungen, die an die Web-Applikation gestellt werden. Diese werden in diesem Kapitel erläutert.

4.2.2.1 Performance

Laut einer Studie des Equation Research von 2016 geben 68% der deutschen, 73% der chinesischen, 58% der indischen und amerikanischen, 57% der französischen, 50% der australischen, und 48% der englischen Befragten an, dass sie eine Ladezeit von drei Sekunden oder weniger von mobilen Webseiten erwarten (siehe Abbildung 3).

Daher darf die zu entwickelnde Web-Anwendung eine Ladezeit von drei Sekunden nicht überschreiten.

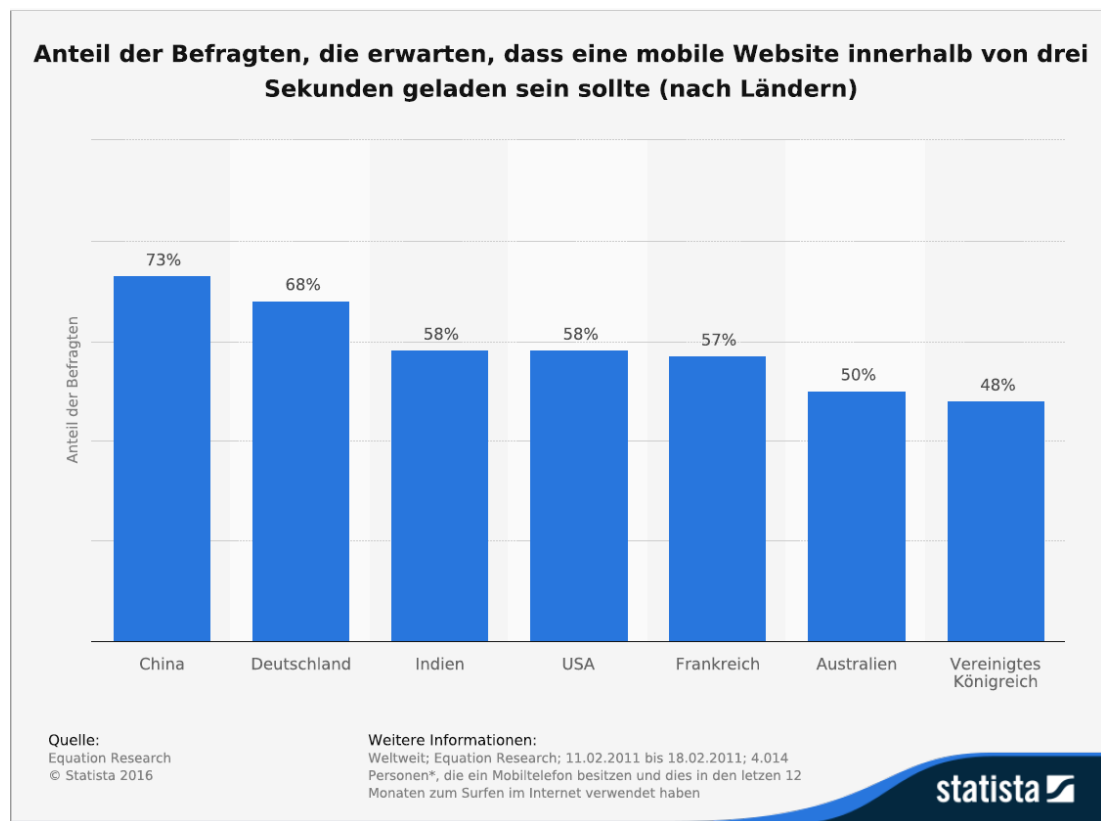


Abbildung 3: . n.d. Anteil der Befragten, die erwarten, dass eine mobile Website innerhalb von drei Sekunden geladen sein sollte (nach Ländern). Statista. Zugriff am 23. August 2016. Verfügbar unter <http://de.statista.com/statistik/daten/studie/202654/umfrage/erwartungen-an-ladezeiten-von-webseiten-auf-mobiltelefonen-nach-laendern/>.

4.2.2.2 Zuverlässigkeit

Die Web-Anwendung muss zu 80% ausfallsicher sein (eine Ausfallsicherheit von 100% kann aus rein technischen Gründen nicht erreicht werden). Das heißt, es müssen Sicherheitsmechanismen geschaffen werden, die einen Server- oder Datenbankausfall kompensieren. Diese sind z.B. Backup-Server, Backup-Datenbanken, etc.

Bei einem Ausfall dürfen keine Daten verloren gehen. Das heißt, die Datenbank muss durch zusätzliche Maßnahmen geschützt werden. Diese Maßnahmen können zum Beispiel Datenbank-Backups auf unterschiedlichen Servern sein.

4.2.2.3 Sicherheit

Die Web-Anwendung muss den heutigen Sicherheit-Standards für Web-Anwendungen genügen. Das heißt, alle Daten müssen verschlüsselt (SSL) über den Browser übermittelt werden, nicht autorisierte Nutzer müssen geblockt werden, und Angriffe von außen – wie SQL-Injection und CORS – müssen blockiert werden.

4.2.2.4 Korrektheit

Die Fehlerdichte ist eine Größe zur Bestimmung der Qualität einer Software. Sie beschreibt die Anzahl von Fehlern pro Systemgröße. Meistens werden hierbei Lines of Code (LOC) als Systemgröße verwendet, also die Länge eines Codes in Zeilen (vgl. Witte 2016: 103).

Allgemein gilt eine Fehlerdichte von < 2 pro 1000 Zeilen Code als erstrebenswert (vgl. Witte 2016: 103). Eine Fehlerdichte von 1,0 oder weniger Fehler pro 1000 LOC gilt in der Industrie als hervorragende Qualität (vgl. Neumann 24-Apr-14). Fehlerdichten von 2-6 Fehler pro 1000 LOC sind normal und Fehlerdichten von 6-10 Fehler pro 1000 LOC zählen in Ausnahmefällen, z.B. bei einer einfachen Informations-Webseite oder einer kleinen App, als gerade noch akzeptabel (vgl. Witte 2016: 103).

In Anbetracht dieser Zahlen sollte der Prototyp der zu entwickelnden Web-Applikation eine Fehlerdichte von 10 Fehlern pro 1000 LOC nicht übersteigen.

Die weiterentwickelte, fertige Anwendung sollte eine Fehlerdichte von 4 Fehlern pro 1000 LOC – Mittel zwischen 2 und 6 – nicht übersteigen.

4.2.2.5 Kompatibilität

Die Web-Anwendung soll von allen Browsern unterstützt werden, die JavaScript und HTML 5 unterstützen. Das trifft auf alle gängigen Browser – Google Chrome, Mozilla Firefox, Safari, Edge, und Opera – zu.

4.2.2.6 Bedienbarkeit

Die Benutzeroberfläche muss übersichtlich gestaltet sein.

Zudem soll der Prototyp zunächst in englischer Sprache, und die fertige Applikation sowohl in deutscher als auch englischer Sprache vorliegen.

Bei Fehlern muss das System eindeutige und verständliche Fehlermeldungen ausgeben. Verständliche Fehlermeldungen geben stets einen Grund für den Fehler an, sofern die Software einen Grund erkennen kann.

4.2.2.7 Datenschutz

Die Web-Anwendung muss dem deutschen Datenschutzgesetz genügen. Dieses legt unter anderem fest:

„Nach dem Bundesdatenschutzgesetz (BDSG) sind die Erhebung, Verarbeitung und Nutzung personenbezogener Daten verboten, es sei denn,

- sie sind durch das BDSG oder eine andere Rechtsvorschrift ausdrücklich erlaubt oder angeordnet oder*
- die / der Betroffene hat dazu seine Einwilligung erklärt.*

Soll eine Einwilligung Grundlage für eine Erhebung, Verarbeitung oder Nutzung sein, sind die Voraussetzungen des § 4a BDSG zu beachten:

- 1. Die Einwilligung bedarf der Schriftform, soweit nicht wegen besonderer Umstände eine andere Form angemessen ist. Die Einwilligung in die Verarbeitung oder Nutzung personenbezogener Daten für Zwecke des Adresshandels oder der Werbung kann auch elektronisch erteilt werden.*
- 2. Die oder der Betroffene ist vorher über die Tragweite seiner Einwilligung aufzuklären (z.B. über den Zweck der Erhebung, Verarbeitung oder Nutzung). Soweit nach den Umständen des Einzelfalles erforderlich oder auf Verlangen ist die oder der Betroffene auch darüber zu informieren, was geschieht, wenn sie bzw. er nicht einwilligt (z.B. dass Ansprüche verloren gehen können).*

3. *Die Einwilligung muss auf der freien Entscheidung der / des Betroffenen beruhen; d.h. sie muss frei von Zwang sein. In diesem Zusammenhang ist auch zu berücksichtigen, ob sich die / der Betroffene in einer besonderen Situation (z.B. Arbeitsverhältnis) befindet, oder ob aufgrund einer faktischen Situation (z.B. Monopolstellung desjenigen, der die Einwilligung einholen will) ein Zwang für die / den Betroffenen besteht.*

Bei der Verarbeitung besonderer Arten personenbezogener Daten gem. § 3 Abs. 9 BDSG (Angaben über die rassische und ethnische Herkunft, politische Meinungen, religiöse oder philosophische Überzeugungen, Gewerkschaftszugehörigkeit, Gesundheit oder Sexualleben) muss sich die Einwilligung ausdrücklich auf diese Daten beziehen.“
(Internetauftritt der Bundesbeauftragten für den Datenschutz und die Informationsfreiheit - Datenschutz - Einwilligung)

Das heißt, jeder Nutzer muss über die Erhebung seiner Daten – Name, Alter, Email-Adresse, Wohnort – informiert werden und diese akzeptieren. Desweiterem werden diese Daten lediglich zur internen Verarbeitung in der Web-Anwendung verwendet.

Die Datenschutzrichtlinien der Seite müssen jederzeit, insbesondere vor und während der Anmeldung, für den Nutzer einsehbar sein.

4.2.2.8 Evaluation

Nach der Entwicklung des Prototyps soll die Web-Anwendung evaluiert werden. Das heißt, ausgewählte Testprobanden sollen die Anwendung testen und bewerten. Das Ergebnis dieser Evaluation soll der Weiterentwicklung des Prototyps dienen.

4.3 Funktionsumfang

Der Funktionsumfang basiert auf den im Vorfeld erarbeiteten funktionalen Anforderungen (siehe Kapitel 4.2.1). Die folgende Tabelle liefert einen Überblick über den Funktionsumfang.

Grundfunktionen	Zusatzfunktionen
Autor-Account erstellen	Nutzer blocken
Autor-Account verwalten	Geschichte archivieren (in Form von PDF)
Charakter erstellen	Geschichte abonnieren / nicht abonnieren
Charakter bearbeiten	Beitrag favorisieren / nicht favorisieren
Charakter löschen	Medien-Upload
Charakter auswählen / wechseln	Öffentliche Nachrichten / Auf Beiträge antworten
Nutzer abonnieren / nicht abonnieren	Private Nachrichten
Geschichte erstellen	Verbesserte Suche (Filter, etc.)
Geschichts-Profil bearbeiten	Events anlegen
Geschichte bearbeiten	Events verwalten
Geschichte löschen	Notifikationen
Beitrag erstellen	Auflistung aller Nutzer
Beitrag bearbeiten	Geschichts-Metadaten
Beitrag löschen	Gruppen-Seiten
Suchfunktion (Freitextsuche)	Rollenspiel-Anfragen
Verschiedene Boards zur Übersicht (Main-Board, Random-Board, Story-Board, Reading List, Mentions-Board)	Berechtigungen für unterschiedliche Bereiche der Applikation (Accounts, Beiträge, Geschichten) anlegen
	Charakter-Metadaten

Tabelle 2: Funktionsumfang

4.4 Analyse vorhandener Web-Applikationen für Schreibrollenspiele

Im Folgendem werden die wichtigsten Web-Applikation, die für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil genutzt werden bzw. genutzt werden könnten, analysiert und dem erarbeiteten Funktionsumfang gegenübergestellt. Im Anschluss werden diese Anwendungen bezüglich ihrer Tauglichkeit als Schreibrollenspiele evaluiert.

4.4.1 Web-Applikationen für kollaboratives Schreiben

Applikationen für kollaboratives Schreiben sind im Web zahlreich vertreten. Bekannte Anwendungen bzw. Anwendungsgruppen werden folgend auf ihre Tauglichkeit für Schreibrollenspiele evaluiert.

Die folgende Tabelle liefert einen Überblick über den Funktionsvergleich der aufgeführten Applikationen mit den geplanten Funktionen von WriRo.

WriRo	Textverarbeitungsprogramme	Splitstory
Autor-Account	JA	JA
Charakter-Profil	NEIN	NEIN
Nutzer abonnieren	NEIN	NEIN
Geschichts-Profil	NEIN	JA
Geschichte herunterladen (PDF)	JA	JA
Geschichte abonnieren	NEIN	NEIN
Random-Beiträge erstellen	NEIN	NEIN
Suchfunktion	NEIN	JA
Verschiedene Übersichtsseiten	NEIN	JA
Charakter-Metadaten	NEIN	NEIN
Berechtigungen	JA	NEIN
Altersbeschränkung	NEIN	NEIN
Nutzer blocken	NEIN	NEIN
Gruppen-Seiten	NEIN	JA
Beiträge favorisieren	NEIN	NEIN
Medien-Upload	JA	JA
Auf Beiträge antworten	NEIN	NEIN
Private Nachrichten	JA	JA
Events	NEIN	NEIN
Notifikationen	NEIN	NEIN

Tabelle 3: Funktionsvergleich von Web-Applikationen für kollaboratives Schreiben

Alle Funktionen über dem Strich (dickere Linie) wurden im Prototyp von WriRo bereits implementiert. Die anderen Funktionen sind bisher lediglich geplant worden. In diesem Zusammenhang bedeutet **Ja**, dass die Funktion von der jeweiligen Applikation unterstützt wird, und **Nein**, dass sie nicht oder nur teilweise unterstützt wird.

4.4.1.1 Textverarbeitungsprogramme

Um die Möglichkeiten von Textverarbeitungsprogrammen als Anwendungen für Schreibrollenspiele zu untersuchen, wurde zunächst analysiert welche Möglichkeiten diese Anwendungen zur Kollaboration bieten. Anschließend wurden diese Möglichkeiten auf ihre Tauglichkeit für Schreibrollenspiele hin untersucht. Repräsentativ für typische kollaborative Textverarbeitungsprogramme wurden Google Docs und Microsoft Word untersucht.

Microsoft Word ist ein bekanntes Textverarbeitungsprogramm der Firma Microsoft, und wird in jeder Version der Office Pakete ausgeliefert. Mit Hilfe des Programms können Text-Dokumente verfasst, gelesen und bearbeitet werden. Über einen sogenannten SharePoint Server oder über den Cloud-Service OneDrive können die Benutzer ihre Dokumente für andere Benutzer mit verschiedenen Berechtigungen freigeben. Alle freigegebenen Dokumente können anschließend im SharePoint betrachtet und bearbeitet werden (vgl. Freigeben eines Dokuments in SharePoint oder auf OneDrive - Word) .

Änderungen im Dokument werden chronologisch gespeichert und nach Benutzer sortiert. Zudem hat jeder Benutzer die Möglichkeit Kommentare zum Dokument hinzuzufügen, die ebenfalls mit allen anderen Benutzern geteilt werden.

Auf diese Weise können sequentiell kollaborative fiktionale und nicht-fiktionale Texte verfasst werden.

Google Docs, das von der Firma Google GmbH bereitgestellt wird, liefert ebenfalls ein gesamtes Office Paket, welches kollaborativ verwendet werden kann.

Im Gegensatz zu Microsofts Office Paket, handelt es sich bei Google Docs um eine reine Web-Anwendung, die nicht offline verwendet werden kann.

Dokumente, die über Google Docs erstellt wurden, können unter der Verwendung einer Email-Adresse für beliebig viele Benutzer freigegeben werden. Dabei können auch hier für jeden Benutzer unterschiedliche Berechtigungen – lesen, schreiben, bearbeiten – zugeteilt werden. Alle vorgenommenen Änderungen werden chronologisch aufgelistet und dem entsprechenden Benutzer zugeordnet. Da die Dokumente stets online bearbeitet werden, können alle Benutzer gleichzeitig an einem Dokument arbeiten.

Im Gegensatz zu Microsoft Office, bietet Google Docs auch die Möglichkeit Dokumente in der Weise zu veröffentlichen, dass alle Benutzer, die einen Google-Account

besitzen das Dokument lesen oder bearbeiten können, unabhängig davon ob das Dokument für sie freigegeben wurde oder nicht.

Auf diese Weise kann auch Google Docs zur kollaborativen Erstellung von fiktiven Texten genutzt werden. Trotz der Möglichkeit des kollaborativen Schreibens eignen sich diese Anwendungen allerdings nicht für Schreibrollenspiele, da sie keine Rollenspiel-Struktur unterstützen.

Es können keine Charakter-Profil-Seiten angelegt werden, da es nur Autor-Accounts gibt. Da es keine Charakter-Profile gibt, können diese auch nicht abonniert werden. Außerdem ist es nicht möglich bestimmte Geschichten zu abonnieren oder zu favorisieren. Desweiterem kann auch kein Geschichts-Profil angelegt werden, welches Metadaten zur Geschichte speichert und einen Überblick über diese liefert. Die Geschichten werden auch nicht automatisch nach Charakter-Beiträgen unterteilt. Eine solche Unterteilung im Text kann lediglich manuell von jedem Benutzer vorgenommen werden.

Das Anlegen von Gruppen- oder Familienseiten zur Vorstellung und internen Verwaltung von Gruppen ist ebenfalls nicht möglich. Es können auch keine Event-Seiten, die zur Verwaltung der Teilnehmer und Termine dienen, angelegt werden.

Außerdem können keine RP-Anfragen erstellt und verwaltet werden, mit Hilfe dessen Benutzer ihre Ideen vorstellen und nach Mit-Autoren suchen können.

Es besteht auch keine Möglichkeit einer gefilterten Suche nach Geschichten, Charakteren, oder Autoren.

Insgesamt lässt sich feststellen, dass Textverarbeitungsprogramme mit kollaborativen Funktionen zwar für kollaboratives Schreiben, aber nicht für Schreibrollenspiele geeignet sind.

4.4.1.2 Splitstory

Mit Hilfe der Web-Anwendung Splitstory lassen sich kollaborativ fiktionale Geschichten, sogenannte **Splitstories**, entwerfen.

„Eine **Splitstory** ist ein abgeschlossener Handlungsstrang aus verknüpften Texten von einem oder mehreren Autoren.“ (Tour: Überblick - Splitstory)

Nach der Anmeldung kann jeder Nutzer sogenannte **Ideen** erstellen und veröffentlichen. Die **Idee** beschreibt eine kurze Zusammenfassung der Idee für den zu schreibenden Text und gibt diesem einen Titel. Außerdem können der Idee Beschreibungen der

Handlungsorte und der Figuren der Geschichte hinzugefügt werden. Diese Beschreibungen dienen zusammen mit der **Idee** als Grundlage für die **Splitstory** (vgl. Tour: Ideen - Splitstory).

Jeder Beschreibung kann auch ergänzend ein Bild oder eine Grafik hinzugefügt werden.

Nach dem veröffentlichen der **Idee** können Beiträge zu dieser **Idee** verfasst werden. So wird die eigentliche Geschichte geschrieben. Jeder Beitrag kann von einem anderen Autor mit einem anderen Beitrag beantwortet werden (vgl. Tour: Texte und Bilder - Splitstory).

Dies kann sequentiell oder reagierend ausgeführt werden (vgl. Kapitel 2.2.1). Zur Übersicht ist die Baumstruktur der **Splitstory** in sogenannten Ebenen angegeben (siehe Abbildung 4).

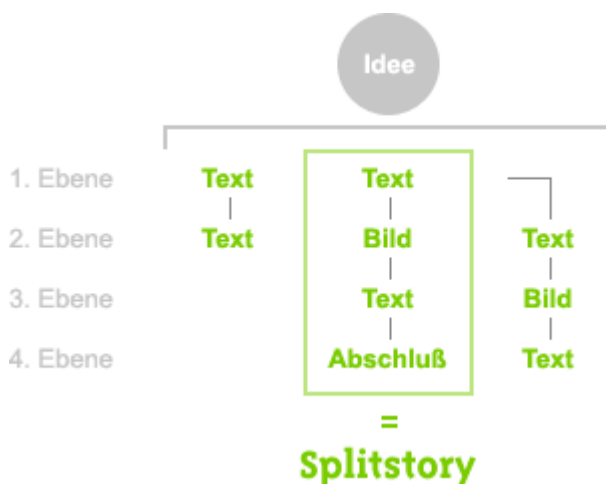


Abbildung 4: Schema einer Splitstory (Quelle: <http://www.splitstory.com/de/tour>)

Sobald ein Autor der Ansicht ist, dass er die Geschichte mit seinem Beitrag abgeschlossen hat, kann er diese beenden. Allerdings bleibt es den anderen Autoren frei die Geschichte mit alternativen Beiträgen weiterzuführen (vgl. Tour: Texte und Bilder - Splitstory).

Die Anwendung liefert auch die Möglichkeit Gruppen zu bilden. So können sich Autoren zu Gruppen zusammenschließen und Gruppen-Ideen veröffentlichen. Dies fördert die Kollaboration (vgl. Tour: Gruppen - Splitstory).

Die Seite bietet die Möglichkeit zum direkten Download der beendeten Geschichte als PDF. Außerdem kann jede Geschichte in einem sogenannten **Lesemodus** im Browser

geöffnet werden. Im **Lesemodus** wird die Geschichte als ein zusammengesetzter Text präsentiert, der sowohl den Titel der Geschichte enthält als auch die mitwirkenden Autoren aufführt.

Die Analyse der Anwendung ergibt, dass sie sich sehr gut als kollaboratives Werkzeug zur Erstellung von fiktionalen und nicht fiktionalen Texten eignet. Allerdings bietet diese Anwendung keine Funktionen für das Schreiben im Rollenspiel-Stil (vgl. Kapitel 3.1.3). Es ist also keine geeignete Anwendung für Schreibrollenspiele.

4.4.2 Für Schreibrollenspiele genutzte Web-Applikationen

Folgende Anwendungen werden häufig von Schreibrollenspielern verwendet, obwohl diese nicht für Schreibrollenspiele entwickelt wurden. Im Folgenden werden diese Anwendungen auf ihre Tauglichkeit als Schreibrollenspiel-Anwendung untersucht.

Die folgende Tabelle liefert einen Überblick über den Funktionsvergleich zwischen den Anwendungen und WriRo.

WriRo	Twitter	Tumblr	Facebook V1	Facebook V2	Facebook V3	Foren
Autor-Account	NEIN	JA	JA	NEIN	JA	NEIN
Charakter-Profil	JA	JA	NEIN	JA	JA	JA
Nutzer abonnieren	JA	JA	NEIN	JA	JA	NEIN
Geschichts-Profil	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN
Geschichte herunterladen (PDF)	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN
Geschichte abonnieren	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN
Random-Beiträge erstellen	JA	JA	JA	JA	JA	NEIN
Suchfunktion	NEIN	NEIN	NEIN	NEIN	NEIN	JA
Verschiedene Übersichtsseiten	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN
Charakter-Metadaten	NEIN	NEIN	NEIN	JA	NEIN	NEIN
Berechtigungen	NEIN	NEIN	NEIN	NEIN	NEIN	NEIN
Altersbeschränkung	NEIN	JA	NEIN	NEIN	NEIN	NEIN

Nutzer blocken	JA	JA	NEIN	NEIN	NEIN	JA
Gruppen-Seiten	NEIN	NEIN	NEIN	JA	NEIN	NEIN
Beiträge favorisieren	JA	JA	JA	JA	JA	NEIN
Medien-Upload	JA	JA	JA	JA	JA	JA
Auf Beiträge antworten	JA	JA	JA	JA	JA	NEIN
Private Nachrichten	JA	JA	JA	JA	JA	JA
Events	NEIN	NEIN	NEIN	JA	NEIN	NEIN
Notifikationen	JA	JA	JA	JA	JA	JA

Tabelle 4: Funktionsvergleich von für Schreibrollenspiele genutzte Web-Applikationen

Alle Funktionen über dem Strich (dickere Linie) wurden im Prototyp von WriRo bereits implementiert. Die anderen Funktionen sind bisher lediglich geplant worden. In diesem Zusammenhang bedeutet **Ja**, dass die Funktion von der jeweiligen Applikation unterstützt wird, und **Nein**, dass sie nicht oder nur teilweise unterstützt wird.

4.4.2.1 Twitter

Die Micro-Blogging-Anwendung Twitter wird von einigen Schreibrollenspielern als Schreibrollenspiel Plattform genutzt (vgl. Bruckman 2013: 200). Diese Nutzer legen Accounts an, die einen fiktionalen Charakter repräsentieren und benutzen Tweets und Mitteilungen zum Schreiben ihrer Geschichten. Das Profil dieses Charakters besteht aus einer kurzen Biografie, in der kurze Angaben zu dem Charakter gemacht werden. Zudem hat jedes Profil einen Usernamen in der Form „@Username“, einen vollen Namen, einen Avatar (Bild des Charakters) und einen Header (siehe Abbildung 5).

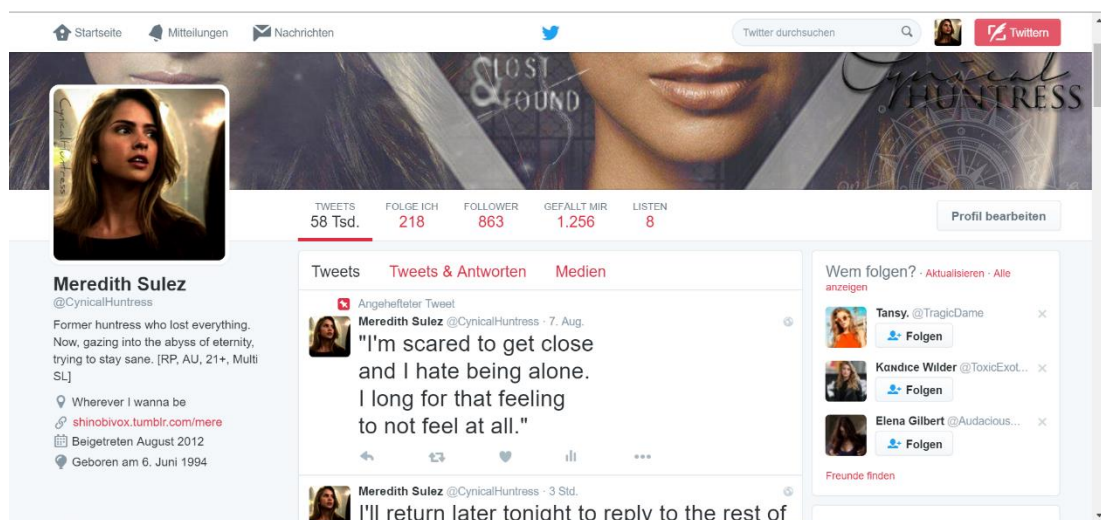


Abbildung 5: Profil eines Rollenspiel-Accounts auf Twitter

Random RP findet auf der sogenannten Timeline (TL), einer chronologischen Auflistung der Tweets, statt (siehe Abbildung 6).

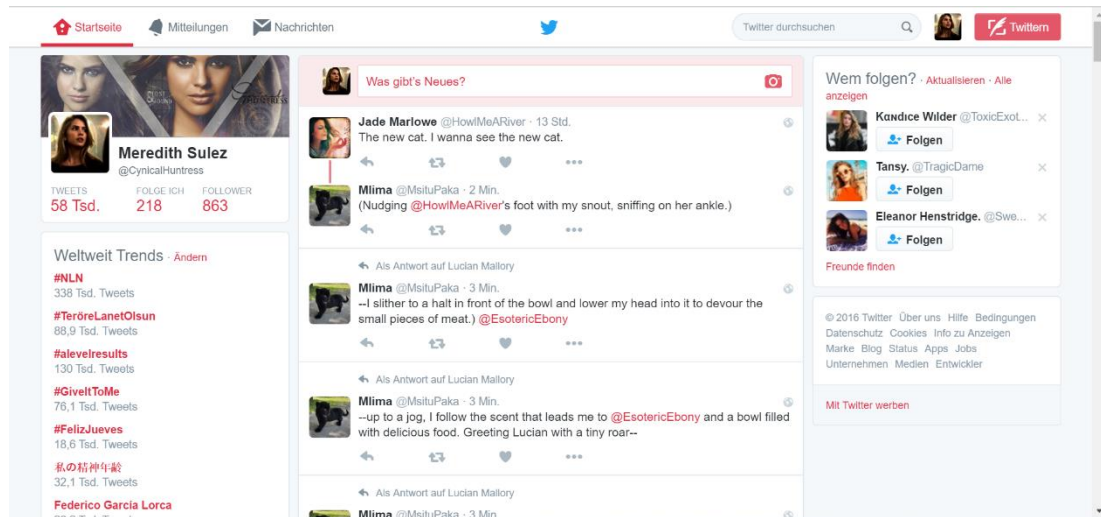


Abbildung 6: Random RP auf Twitter

SLs werden im Vorfeld über Private Nachrichten (DM) von den Mitwirkenden geplant und anschließend ebenfalls auf der TL gepostet. Im Gegensatz zu Random RPs beinhalten SL Tweets zu Beginn stets einen Hashtag in der Form „#Titel“, der den Titel der Geschichte enthält. Da alle Tweets ein Zeichenlimit von 140 Zeichen haben, ein typischer RP-Beitrag aber bis zu 1500 Wörter lang ist, werden die zusammenhängenden Tweets mit entsprechenden Zeichen markiert. Diese Zeichen sind zum Beispiel Pfeile, Klammern, oder Striche, die symbolisieren, dass es vor und nach dem Tweet einen Tweet gab, der mit diesem verbunden ist (siehe Abbildung 7).

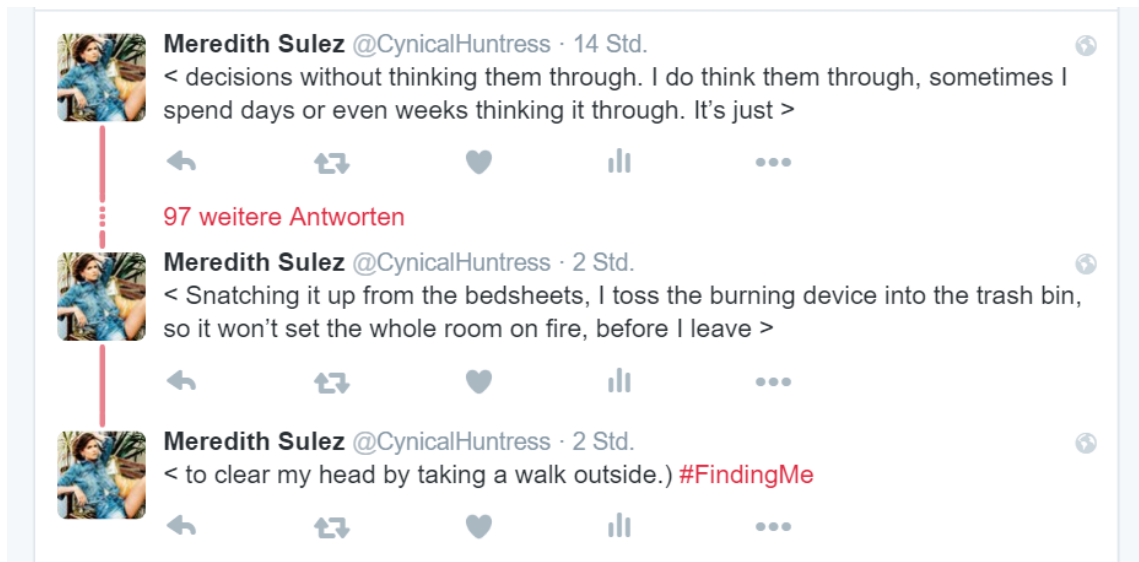


Abbildung 7: Ausschnitt eines RP-Beitrags auf Twitter

Wie beschrieben kann Twitter als Anwendung für Schreibrollenspiele verwendet werden. Es ist allerdings nicht ausreichend für Schreibrollenspiele geeignet.

Es gibt keinen übergeordneten Account, mit welchem die Charaktere verwaltet werden könnten. Wenn ein Nutzer mehrere Charaktere repräsentieren möchte, muss er mehrere separate Accounts anlegen.

SLs und Random RPs vermischen sich alle auf der Hauptseite und in den Mitteilungen der Accounts (siehe Abbildung 9). Es gibt keine Möglichkeit separate Geschichtsseiten zu erstellen. Daher ist es auch nicht möglich Geschichten zu abonnieren oder herunterzuladen. Es kann höchstens ein Archiv des gesamten Profils angefordert werden. Beiträge (Tweets) auf Twitter können nicht bearbeitet werden. Sie können nach dem Erstellen lediglich komplett gelöscht werden.

Es gibt zwar eine Übersichtsseite, die alle Tweets der gefolgt Accounts zeigt, diese lässt sich aber nicht sortieren oder ordnen. Die Chronik der TL wird nur unterbrochen, wenn Tweets miteinander verkettet werden. Dies wird **Threading** genannt. Tweets, die aufeinander antworten werden verkettet und untereinander aufgeführt. Werden mehr als drei Tweets miteinander verkettet, wird der erste und die beiden letzten Beiträge in der TL angezeigt, und die Nummer der dazwischenliegenden Tweets in roter Schrift angezeigt (siehe Abbildung 8). Erst bei Klick auf diese Schrift, werden die restlichen Tweets angezeigt. Auf diese Art wird eine Semi-Ordnung erzeugt.

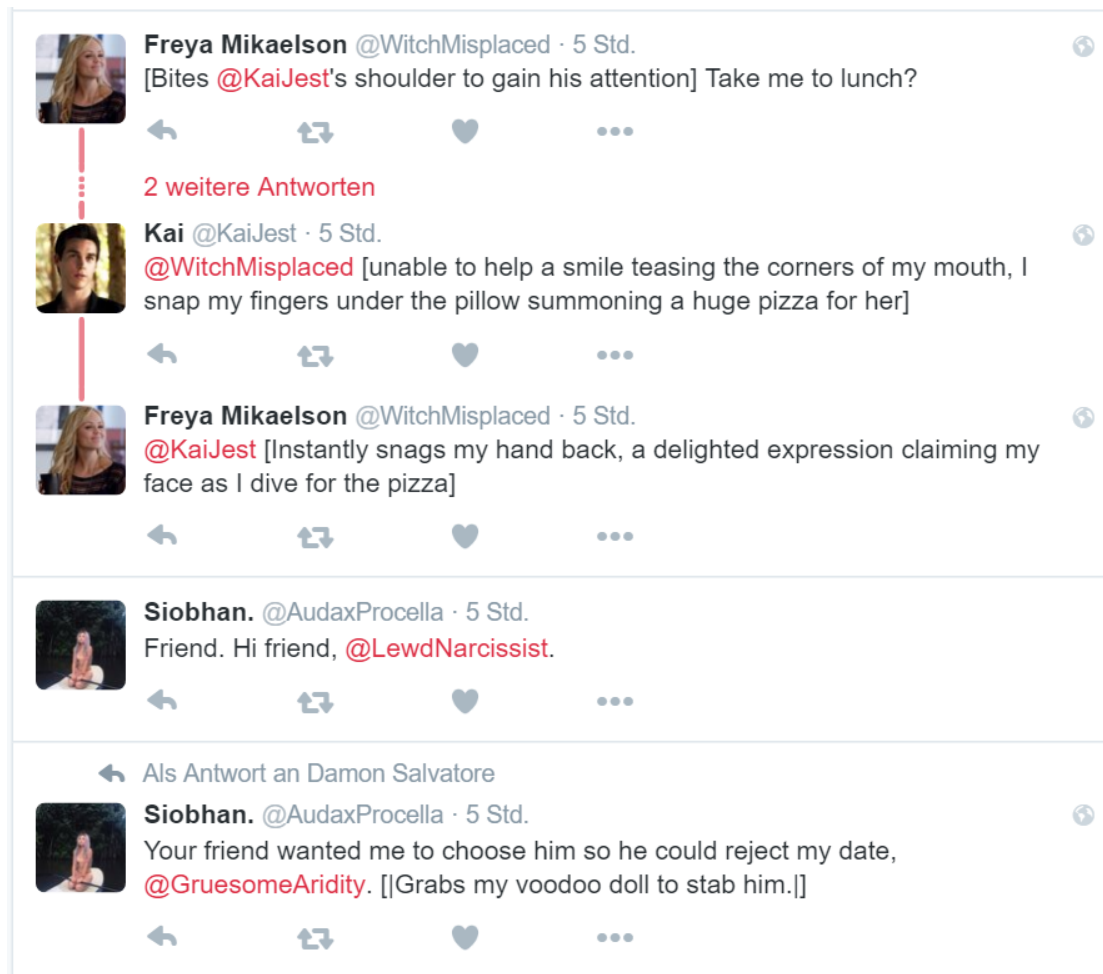


Abbildung 8: Twitter TL mit Threading

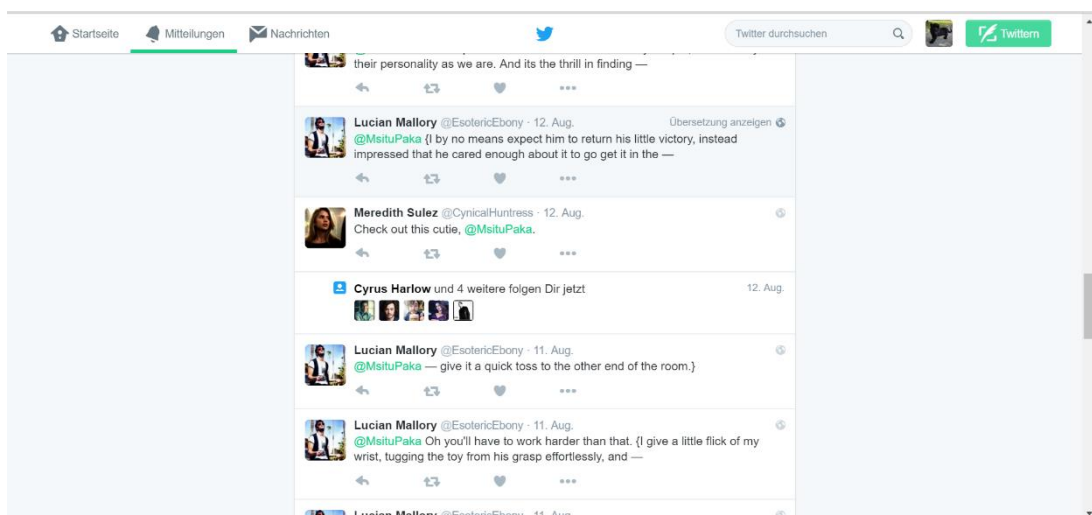


Abbildung 9: Mitteilungs-Seite eines RP-Accounts auf Twitter

Im Profil des Charakters gibt es keine Kategorie für die Metadaten des Charakters. Diese müssen manuell in die Biografie eingetragen werden. Außerdem gibt es keine Möglichkeit Accounts miteinander zu verbinden. Möchte der Nutzer den Lebensgefährten oder Familienmitglieder seines Charakters angeben, muss er diese in der Biografie aufnehmen. Die Biografie des Charakters ist auf 160 Zeichen reduziert (vgl. Das Twitter Glossar). Mit allen zusätzlichen Angaben – z.B. der Hinweis zum Fake-Account (vgl. Richtlinien für Parodie-, Kommentar- oder Fan-Accounts) und der Charakter-Metadaten – bleibt nicht viel Platz für die eigentliche Charakterbeschreibung.

Die Möglichkeit den Account für bestimmte Altersgruppen zu blockieren gibt es nicht. Der Account kann zwar generell für alle Nutzer gesperrt werden (privater Account), d.h. alle Beiträge können nur von Abonnenten gelesen werden, oder für einzelne Nutzer (blockieren), aber das Blockieren von bestimmten Nutzergruppen ist nicht möglich. Außerdem kann in den Twitter Einstellungen ein entsprechender Warnhinweis auf hochgeladenem Bildmaterial eingestellt werden. Reine Text-Tweets oder Profile erhalten diesen Warnhinweis allerdings nicht. Desweiterem besteht auch die Möglichkeit zur Vergabe von Berechtigungen (lesen, schreiben, löschen) für einzelne Tweets, zum Beispiel einer bestimmten Geschichte, nicht.

Die Suche in Twitter liefert die Möglichkeit nach Usernamen, Namen und Tweet-Inhalten zu suchen. Diese Suchfunktionen reichen allerdings für Schreibrollenspiele nicht aus, da nicht nach Geschichten, Genre, Charakteren, Altersfreigaben, oder ähnlichem gesucht werden kann.

Insgesamt lässt sich feststellen, dass Twitter nicht als Anwendung für Schreibrollenspiele geeignet ist, obwohl es durchaus als solche genutzt wird.

4.4.2.2 Tumblr

Tumblr ist eine Blogging-Plattform, die im Jahr 2007 von David Karp gegründet wurde (siehe Über | Tumblr). Mit Hilfe von Tumblr können Nutzer Texte, Bilder, Videos, und Audio-Material veröffentlichen. Die Templates der Blogs können beliebig angepasst werden. Auch HTML-Anpassungen können vorgenommen werden. Von einigen Nutzern wird Tumblr als Schreibrollenspiel-Plattform genutzt.

Für jeden Charakter wird ein Blog angelegt, der mit Hilfe von Themen individualisiert werden kann. Jeder Blog hat typischerweise eine About-Seite, die den Charakter beschreibt (vgl. Tutorial: Create a Roleplaying Blog on Tumblr | Personal Reflections on

WordPress.com). Sie entspricht dem Charakterprofil mit allen nötigen Informationen über den Charakter.

Random RP und SLs werden auf Tumblr mit Hilfe der Beiträge verfasst. Jeder Beitrag zählt entweder als Random RP Beitrag oder als SL Beitrag, auf den andere Charaktere antworten können. Dazu wird der Beitrag zitiert und kommentiert. Eine gewisse Ordnung entsteht durch die Verwendung von sogenannten Tags (vgl. [How to Roleplay on Tumblr](#), vgl. auch [Tutorial: Create a Roleplaying Blog on Tumblr | Personal Reflections on WordPress.com](#)). Tags sind Hashtags, die jedem Beitrag hinzugefügt werden können. Typische Tags sind der Titel der Geschichte, und #rp oder #roleplay.

Es ist ebenfalls möglich andere Nutzer zu abonnieren und Berechtigungen für Blogs und Beiträge zu vergeben. Diese Berechtigungen beschränken sich auf „öffentlich“ und „privat“, d.h. ein Beitrag oder Blog kann entweder von jedem gelesen werden, oder nur von Abonnenten.

Ähnlich wie bei Twitter erscheinen alle eigenen Beiträge und Beiträge von abonnierten Nutzern auf der Startseite eines Accounts, dem sogenannten Dashboard.

Ein Tumblr Account bildet, anders als der Twitter Account, einen übergeordneten Account, von dem aus die unterschiedlichen Blogs verwaltet werden können. Daher können ohne Probleme mehrere Charaktere vom selben Account verwaltet werden.

Die Übersichtsseite, hier das Dashboard, bietet lediglich eine Übersicht der Beiträge aller abonnierten Blogs bzw. Charaktere. Eine Übersicht von Geschichten gibt es nur in Form einer Auflistung aller Geschichten eines Autors auf einer Seite (vgl. [Tutorial: Create a Roleplaying Blog on Tumblr | Personal Reflections on WordPress.com](#)). Die Geschichten können zwar über Hashtags gefunden werden, eine eigene Seite haben einzelne Geschichten aber in der Regel nicht. Daher gibt es auch kein Geschichtsprofil oder die Möglichkeit eine Geschichte zu abonnieren oder herunterzuladen.

Metadaten des Charakters können ebenfalls nur manuell, auf der About-Seite, festgehalten werden. Es gibt kein einheitliches Formular für das Charakterprofil. Daher gibt es auch nicht die Möglichkeit nach bestimmten Charaktermerkmalen zu suchen. Auch können Charaktere nicht miteinander verlinkt werden, sofern diese Funktion nicht vom Nutzer selbst programmiert wird.

Bei Tumblr gibt es die Möglichkeit bestimmte Blogs, d.h. Charaktere, mit einer Altersbeschränkung zu versehen. Dadurch werden alle Beiträge des Blogs als nicht jugendfrei

markiert. Jedem Nutzer steht es frei diese Inhalte zu betrachten oder nicht. Es gibt keine zuverlässige Kontrolle, ob der Nutzer tatsächlich nicht minderjährig ist.

Private Nachrichten zwischen den Nutzern können nur über den Haupt-Account der Nutzer versendet werden. D.h. es gibt keine Möglichkeit private Nachrichten zwischen den Charakteren zu versenden.

Gruppenseiten können in Form von Blogs angelegt werden. Allerdings können die Mitglieder dieser Gruppen nur manuell, d.h. wenn der Nutzer diese Funktion selbst programmiert, mit der Seite verlinkt werden.

Die Möglichkeit eine Event-Seite mit Terminkalender anzulegen gibt es nicht. Zwar kann eine Event-Seite als Blog angelegt werden, es gibt aber keine Terminkalenderfunktion. Die Gästeliste kann ebenfalls nur manuell über eine Seite des Blogs angelegt werden. Die Möglichkeit Charaktere zu einem Event einzuladen, und automatisch zur Gästeliste hinzuzufügen, gibt es nicht.

Im Allgemeinen lässt sich feststellen, dass Tumblr aufgrund der Individualisierungsmöglichkeiten der einzelnen Blogs als Schreibrollenspiel-Anwendung verwendet werden kann, aber nicht ideal für diese Anwendungsart geeignet ist.

4.4.2.3 Facebook

Facebook ist ein bekanntes soziales Netzwerk, welches 2004 von Mark Zuckerberg und Mit-Gründer Dustin Moskovitz, Chris Hughes, und Eduardo Saverin gegründet wurde (siehe FOCUS 2012).

Facebook bietet prinzipiell drei Varianten der Nutzung als Schreibrollenspiel-Anwendung. In der ersten Variante dient der Facebook-Account des Nutzers als Rollenspiel-Plattform, auf der sich Rollenspiel und Realität vermischen, in der zweiten Variante repräsentiert jeder Account einen Charakter, und in der dritten Variante dient der Account als Autor-Account und Charaktere werden als Seiten präsentiert.

Die erste Variante ist die einfachste Variante, die zugleich die ungeeignetste Form für Schreibrollenspiele ist. In dieser Variante benutzen die Schreibrollenspiel-Nutzer ihre persönlichen Facebook-Accounts, auf denen sie auch über reale Ereignisse ihrer persönlichen Person berichten, um mit anderen Facebook-Nutzern zu schreiben.

Dafür wird die sogenannte Pinnwand, einer chronologischen Auflistung von Beiträgen im Profil des Benutzers, verwendet. Durch die verwendete, narrative Sprache des

Beitrages wird zwischen RP-Beiträgen und normalen Beiträgen unterschieden. Ein RP-Beitrag könnte folgendermaßen aussehen:

„Hey, ich bin Sarah,“ begrüße ich das Mädchen, das lässig gegen die Wand lehnt und ebenfalls auf den Beginn der ersten Stunde wartet. Es ist mein erster Tag in dieser Schule und ich würde mich gerne dem ein oder anderem vorstellen, bevor ich von den Lehrern vor die Klasse gestellt und präsentiert werde wie ein neues Exponat für den Unterricht.

Die zweite Variante ist zwar geeigneter für Schreibrollenspiele, verstößt aber gegen die Nutzungsregeln von Facebook.

In dieser Variante legen die Schreibrollenspiel-Nutzer für jeden Charakter einen eigenen Facebook-Account an, der die fiktionale Figur repräsentiert. Im Sinne der Facebook-Richtlinien sind dies Fake-Accounts, da sie keine reale Person darstellen (vgl. Facebook hates roleplayers | OngoingWorlds roleplay blog). Da diese Art von Accounts gegen die Facebook-Richtlinie verstoßen, werden sie regelmäßig von Facebook gelöscht oder gesperrt (vgl. Facebook hates roleplayers | OngoingWorlds roleplay blog). Trotz der Anfrage mehrere Nutzer und einer regen Beteiligung an entsprechenden Petitionen zur Aufnahme von Rollenspiel-Accounts auf Facebook, wurden diese Richtlinien bis heute nicht angepasst. Laut Facebook wurde die Plattform nicht für Rollenspiele oder dem Schreiben fiktionaler Geschichten entwickelt und soll auch nicht als solche verwendet werden (vgl. The Strange World of Internet Role-Play Has Gone Mainstream, vgl. auch Facebook hates roleplayers | OngoingWorlds roleplay blog, vgl. auch Allgemeine Geschäftsbedingungen).

Die dritte Art des Rollenspiels auf Facebook ist die geeignetste Form des Schreibrollenspiels, die Facebook liefert.

In dieser Variante legen die Nutzer zunächst einen normalen Facebook-Account, der als Autor-Account dient, an. Für jeden Charakter werden anschließend Seiten angelegt. Auf dieser Seite kann, ähnlich wie im Account, ein Profil angelegt werden, in dem die wichtigsten Informationen des Charakters festgehalten werden können. Seiten können von anderen Nutzern abonniert werden, wodurch die Beiträge dieser Seiten in der Chronik, der Hauptseite des Facebook-Accounts, des Nutzers angezeigt werden. Allerdings können Seiten keine Seiten oder Accounts abonnieren.

Genauso wie bei den anderen beiden Varianten wird für das Schreiben der Geschichten ausschließlich die Pinnwand einer Seite verwendet.

Unabhängig von der oben beschriebenen Variante liefert Facebook generell die Möglichkeit Beiträge zu favorisieren (liken), Medien hochzuladen, und Notifikationen einzustellen. Außerdem können sowohl Random RP, als auch Solos, und SLs geschrieben werden. Auch eine Suchfunktion ist auf Facebook gegeben. Diese eignet sich allerdings nur bedingt um Schreibrollenspiel-Beiträge (Charaktere, Geschichten, etc.) zu finden, da sie für das Suchen von realen Personen und Seiten ausgelegt wurde.

Facebook liefert keine Möglichkeit Nutzer zu blockieren (es können lediglich Freundschaftsanfragen ignoriert werden), eine Altersbeschränkung oder sonstige Berechtigungen zu vergeben (lediglich unterschiedliche Lesereichweiten – privat, alle Freunde, öffentlich – können vergeben werden), sowie Geschichtsseiten anzulegen. Daher können die Geschichten auch nicht abonniert oder heruntergeladen werden. Ähnlich wie bei Twitter werden die Beiträge lediglich chronologisch sortiert.

In der ersten Variante gibt es nur einen Autor-Account, während es in der zweiten Variante nur einen Charakter-Account gibt, und in der dritten Variante sowohl einen Autor- als auch einen Charakter-Account.

In der ersten und dritten Variante gibt es keine Möglichkeit Metadaten des Charakters oder der Geschichten festzuhalten. Die zweite Variante liefert die Möglichkeit Metadaten der Charaktere (Familienmitglieder können zum Beispiel verlinkt werden) festzuhalten.

Die erste Variante bietet keine Möglichkeit andere Nutzer zu abonnieren, da die private Facebook-Seite nicht von den RP-Seiten getrennt wird. Die zweite Variante liefert die Möglichkeit andere Nutzer zu abonnieren bzw. sich mit ihnen zu befreunden (aus Sicht des Charakters, da es keinen Autor-Account gibt). Die dritte Variante liefert die Möglichkeit sowohl Autoren als auch Charaktere zu abonnieren.

In der ersten Variante des Facebook-RPs gibt es keine Übersichtsseite für unterschiedliche Beiträge. Alle RP-Beiträge vermischen sich mit den Statusmeldungen des Autors und dessen Freunde. Die zweite Variante liefert eine Übersichtsseite aller Beiträge des Charakters und aller abonnierten Charaktere und die Übersichtsseite aller eigenen Beiträge des Charakters. Weitere Übersichtsseiten (wie Geschichten oder Autor-Beiträge) gibt es auch hier nicht. Die dritte Variante liefert lediglich eine Übersicht aller

Beiträge des Autors und Charakters und deren abonnierten Autoren und Charaktere, sowie einzelne Übersichtsseiten der eigenen Autor- und Charakter-Beiträge. Getrennte Übersichtsseiten aller Charakter-Beiträge mit den Beiträgen der abonnierten Charaktere, oder aller Autor-Beiträge mit den Beiträgen der abonnierten Autoren, oder der Geschichten gibt es nicht.

In der ersten Variante können weder Gruppen- noch Event-Seiten angelegt werden. In der zweiten Variante können sowohl Gruppen- als auch Event-Seiten mit allen Funktionen angelegt werden (vgl. Kapitel 4.2.1.4 und 4.2.1.10). In der dritten Variante können zwar Gruppen- und Event-Seiten angelegt werden, es können aber lediglich Autoren (keine Charaktere) mit diesen Seiten verlinkt werden (zum Beispiel auf der Mitgliederliste).

Private Nachrichten können auf Facebook generell verfasst werden. In der ersten Variante können diese allerdings lediglich als Autor verfasst werden, während sie bei der zweiten Variante nur als Charakter verfasst werden können. Die dritte Variante liefert die Möglichkeit private Nachrichten sowohl als Charakter als auch als Autor zu verfassen.

Diese Analyse ergibt, dass Facebook nicht als Schreibrollenspiel-Anwendung geeignet ist, obwohl es durchaus als solche verwendet wird.

4.4.2.4 Foren

Auch Foren werden häufig für Schreibrollenspiele genutzt. Folgend werden mehrere Möglichkeiten, wie Foren zur Unterstützung von Schreibrollenspielen genutzt werden, aufgezeigt. Die hier aufgeführten Abbildungen stammen aus dem Forum **Carpe Noctem** (<http://tdvfantasy.siteboard.eu/>).

Eine weitverbreitete Art des Foren-Schreibrollenspiels sieht vor, dass ein Forum genau eine fiktionale Spielwelt abbildet. Die Regeln und Beschaffung dieser Spielwelt werden in einem Thread des Forums festgehalten.

Die Untergruppen bzw. Ordner des Forums repräsentieren dann unterschiedliche Bereiche dieser fiktionalen Welt, z.B. Stadtviertel, oder eine Schule, während dessen Unterforen bestimmte Teile dieser Bereiche repräsentieren, z.B. Klassenräume oder Straßen (siehe Abbildung 10).

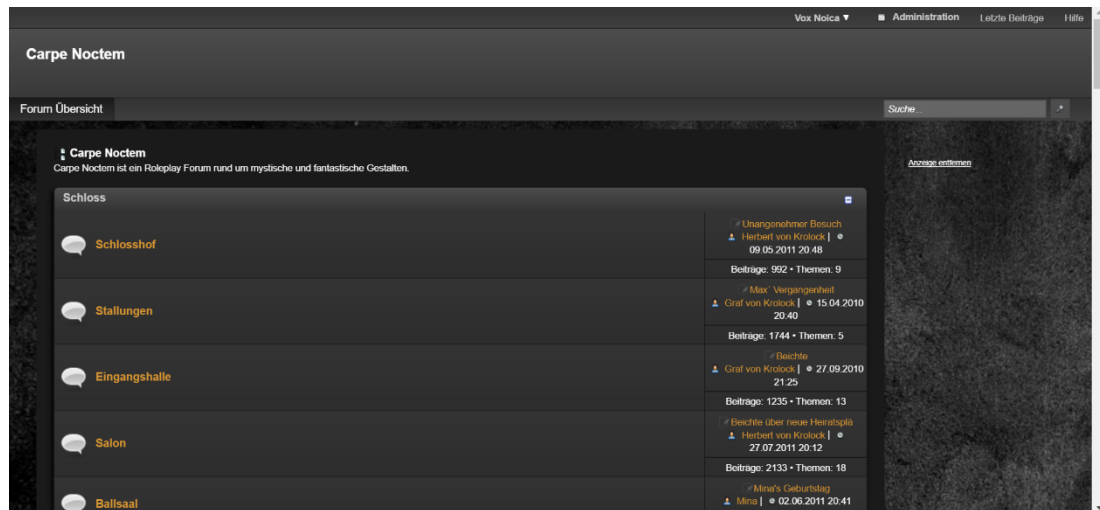


Abbildung 10: Typische Struktur eines Forum-Rollenspiels

In manchen dieser Foren wird live geschrieben. D.h. die Geschichte wird in dem entsprechenden Unterforum bzw. Thread fortgeführt, in dem sich der Charakter befindet. Befindet sich der Charakter zum Beispiel im Klassenzimmer 201, schreibt der Autor dieses Charakters im Thread **Klassenzimmer 201** im Unterforum **Klassenräume** des Forums **Schule**. Verlässt der Charakter diesen Raum, gibt er in seinem Beitrag an, wo er hinget und schreibt die Geschichte an dem entsprechenden Ort weiter.

In dieser Art des RPs wird nur eine einzige Geschichte abgebildet, die in dem Forum, wie in einer Art Computerspiel, ausgespielt wird.

In anderen Foren stellt jeder Thread eine eigene Geschichte dar, die zuvor in einem Planungsthread geplant und vorbereitet wurde (vgl. Carpe Noctem).

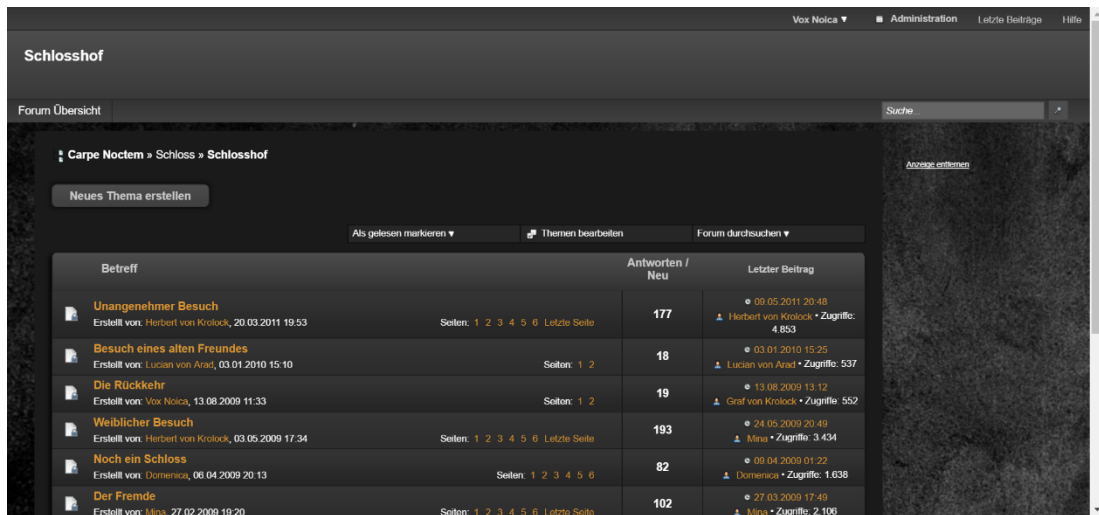


Abbildung 11: Beispiel für ein Forum-Rollenspiel; jeder Thread entspricht einer Geschichte

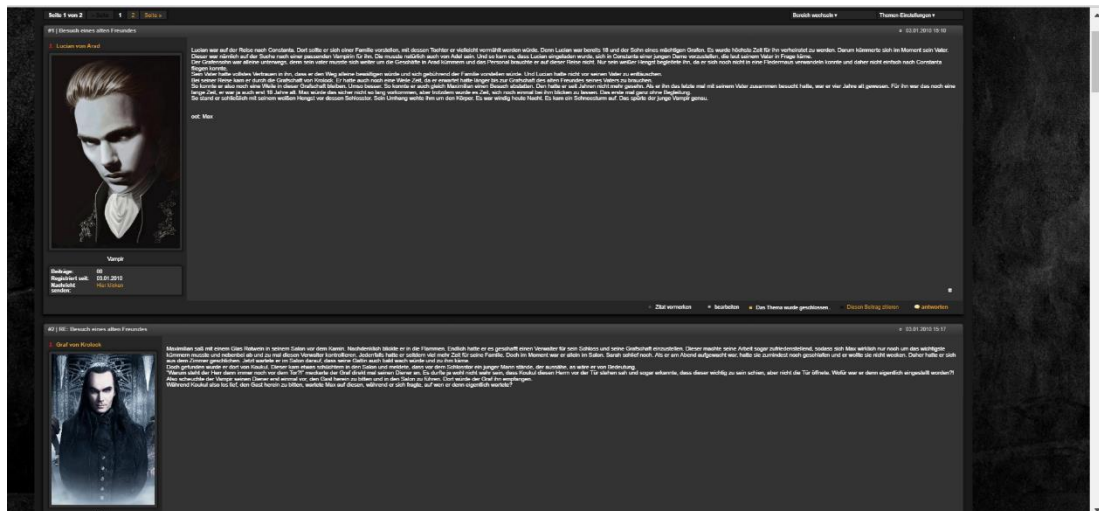


Abbildung 12: Aufbau eines Threads im Foren RP

In einer weiteren Form des Foren RPs repräsentiert jedes Unterforum eine eigene, neue fiktionale Spielwelt, sodass in einem Forum mehrere Geschichten in unterschiedlichen Spielwelten geschrieben werden können.

In allen Arten des Foren RPs repräsentiert jeder Account nur einen bestimmten Charakter, der sowohl im Account-Profil als auch in einem Unterforum – meistens **Charakterbiografien** genannt – beschrieben wird.

Obwohl Foren viele Möglichkeiten der Verwendung als Schreibrollenspiel-Anwendung bieten, lassen sich einige Mängel feststellen.

So gibt es keinen übergeordneten Account zur Verwaltung der Charaktere, sondern lediglich einzelne Charakter-Accounts mit jeweils eigenen Login-Daten.

In Foren gibt es keine Möglichkeit andere Charaktere oder Geschichten zu abonnieren, oder eine Übersichtsseite mit allen Beiträgen anzuzeigen.

Auch können einzelne Charaktere nicht miteinander verlinkt werden. Eventuelle Beziehungen zwischen den Charakteren können nur manuell in der Charakterbiografie eingetragen werden.

Zudem kann keine Altersbeschränkung für bestimmte Bereiche des Forums oder Accounts festgelegt werden, die vom System selbst geprüft würde. Es können lediglich entsprechende Hinweise gemacht werden.

Auch sonstige Berechtigungen – lesen, schreiben, löschen – können nur von Moderatoren oder Administratoren des Forums vorgenommen werden, nicht aber von allen Nutzern.

Gruppen- oder Eventseiten können nur bedingt als Unterforen oder Threads angelegt werden. Eine Verlinkung von Mitgliedern bzw. Teilnehmern ist allerdings nicht möglich.

Eine Archivierung bzw. Herunterladen der Geschichten ist in Foren ebenfalls nicht möglich, d.h. die Geschichten können nur online und im Forum gelesen werden.

Aus diesen Gründen lässt sich feststellen, dass Foren nicht als Schreibrollenspiel-Anwendung geeignet sind.

4.4.3 Für Schreibrollenspiele entwickelte Web-Applikationen

In diesem Kapitel werden Applikationen, die für Schreibrollenspiele entwickelt wurden erläutert und auf ihre Tauglichkeit analysiert.

Die folgende Tabelle liefert einen Überblick über den Vergleich der Funktionen, die WriRo unterstützt, und der Funktionen der analysierten Anwendungen.

WriRo	Roleplay.me	RolePlayGateway und OngoingWorlds
Autor-Account	JA	JA
Charakter-Profil	JA	JA
Nutzer abonnieren	JA	NEIN
Geschichts-Profil	NEIN	JA
Geschichte herunterladen (PDF)	NEIN	NEIN
Geschichte abonnieren	NEIN	NEIN
Random-Beiträge erstellen	JA	NEIN
Suchfunktion	JA	JA
Verschiedene Übersichtsseiten	JA	JA
Charakter-Metadaten	NEIN	NEIN
Berechtigungen	NEIN	NEIN
Altersbeschränkung	NEIN	NEIN
Nutzer blocken	JA	NEIN
Gruppen-Seiten	NEIN	JA
Beiträge favorisieren	JA	NEIN
Medien-Upload	JA	JA
Auf Beiträge antworten	JA	NEIN
Private Nachrichten	JA	JA
Events	NEIN	NEIN
Notifikationen	NEIN	NEIN

Tabelle 5: Funktionsvergleich von Web-Applikationen für Schreibrollenspiele

Alle Funktionen über dem Strich (dickere Linie) wurden im Prototyp von WriRo bereits implementiert. Die anderen Funktionen sind bisher lediglich geplant worden. In diesem Zusammenhang bedeutet **Ja**, dass die Funktion von der jeweiligen Applikation unterstützt wird, und **Nein**, dass sie nicht oder nur teilweise unterstützt wird.

4.4.3.1 Roleplay.me

Roleplay.me ist eine der wenigen Web-Applikationen, die für Schreibrollenspiele entwickelt wurde. Die Anwendung ist eine Mischung aus Blog, Forum, und sozialem Netzwerk. Jeder Nutzer hat die Möglichkeit einen oder mehrere Charakter-Accounts anzulegen. Diese Accounts haben ein Profil, ähnlich dem Profil bei Facebook und

Twitter, über das mit anderen Nutzern kommuniziert werden kann (siehe Abbildung 14). Außerdem kann jeder Nutzer Blogs anlegen, in denen sie die Charaktere vorstellen können (siehe Abbildung 15). Der Forum-Bereich der Anwendung dient dem Suchen von Mitschreibern und dem Vorstellen von Geschichtsideen. Die Geschichte selber wird entweder im Forum oder über private Nachrichten der beteiligten Charaktere geschrieben (siehe RolePlay.me | RolePlay Online).

Die Accounts, die in der Anwendung erstellt werden, sind Charakter-Accounts. Einen übergeordneten Autor-Account, der mehrere Charaktere verwaltet, gibt es nicht.

Charakter-Profile mit den wichtigsten Informationen sind in Form des Account-Profils vorhanden. Nähere Informationen über den Charakter können in einem Blog hinterlegt werden.

Geschichten werden entweder im Forum (siehe Abbildung 17) oder in privaten Nachrichten geschrieben, ein Geschichtsprofil gibt es nicht. Daher können andere Nutzer die Geschichten entweder gar nicht lesen (wenn sie in privaten Nachrichten geschrieben werden) oder nicht abonnieren oder herunterladen (wenn sie im Forum geschrieben werden).

Die Metadaten der Geschichte, wie eine Zeitachse oder ein Spannungsbogen, können zwar im Forum diskutiert werden, die Anwendung bietet aber keine Werkzeuge um diese anzulegen und zu speichern.

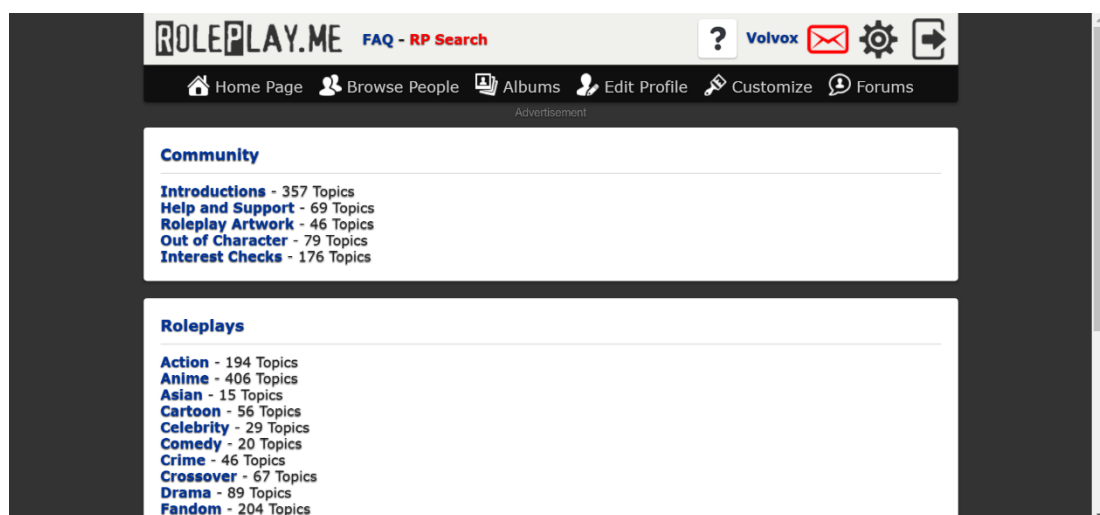


Abbildung 13: Forum-Übersicht auf Roleplay.me

Charakterseiten können miteinander befreundet sein. Dies entspricht einem Abonnieren der entsprechenden Seite. Über die Charakter-Profile können diese auch blockiert werden, sodass sie den blockierenden Nutzer nicht mehr kontaktieren können.

Sowohl das Verfassen von Solos (meistens im Blog), Random RP (meistens auf der Profilsseite), und SLs (meistens im Forum oder in privaten Nachrichten) ist möglich. Durch die Verwendung von Foren, Blogs, und Profilsseiten sind verschiedene Übersichtsseiten vorhanden. Im Profil eines Charakters werden stets dessen Profilbeiträge angezeigt. Auf der Startseite eines Accounts werden die eigenen Beiträge und die der befreundeten Nutzer angezeigt.

The screenshot shows a user profile for 'JosieElsyChapman' on the website Roleplay.me. At the top, there are navigation links: 'Add Friend', 'Block', 'Report', and 'Offline'. The profile features a large profile picture of a woman with long, wavy brown hair. To the right of the picture, the name 'JosieElsyChapman' is displayed in bold, with the alias '(josie saunders)' in red below it. Further down, her age and status are listed as '19 / Female / Single' and her location as 'Illinois - United States'. A 'Latest Status' section shows a post from August 16th, 2016, at 18:25, with the text: 'Hi I'm Josie age 19 ,Inmate at Litchfield, and sister to Piper Chapman OITNB RP'. Below the status, there are sections for 'Latest Comments' and 'Latest Status' with links to 'View All Updates' and 'View All Comments'. A statistics box shows 'Last Visited: August 17th, 2016', 'Member Since: August 16th, 2016', and 'Profile Views: 36'. A 'Characters' section lists 'Josie.Elsy.Chapman' as the character, 'Orange Is The New Black' as the series, and 'Any, Casual, Drama' as the genres. A 'Friends' section shows '4 Friends' with a link to 'View All Friends' and a small grid of friend avatars. A 'Photos' section shows '1 Photos' with a link to 'View All Photos'. At the bottom, there is a footer with links for 'Terms Of Use', 'Privacy', 'Copyright', 'Icons', 'Knowing God', 'Contact', and 'Vote For Us'.

Abbildung 14: Charakter-Profil auf Roleplay.me

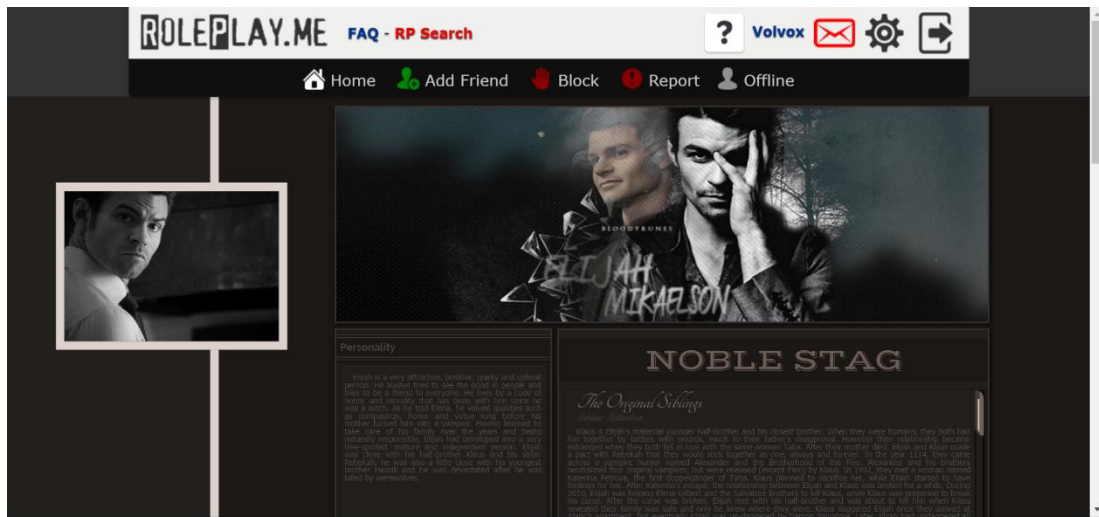


Abbildung 15: Ein Blog auf Roleplay.me

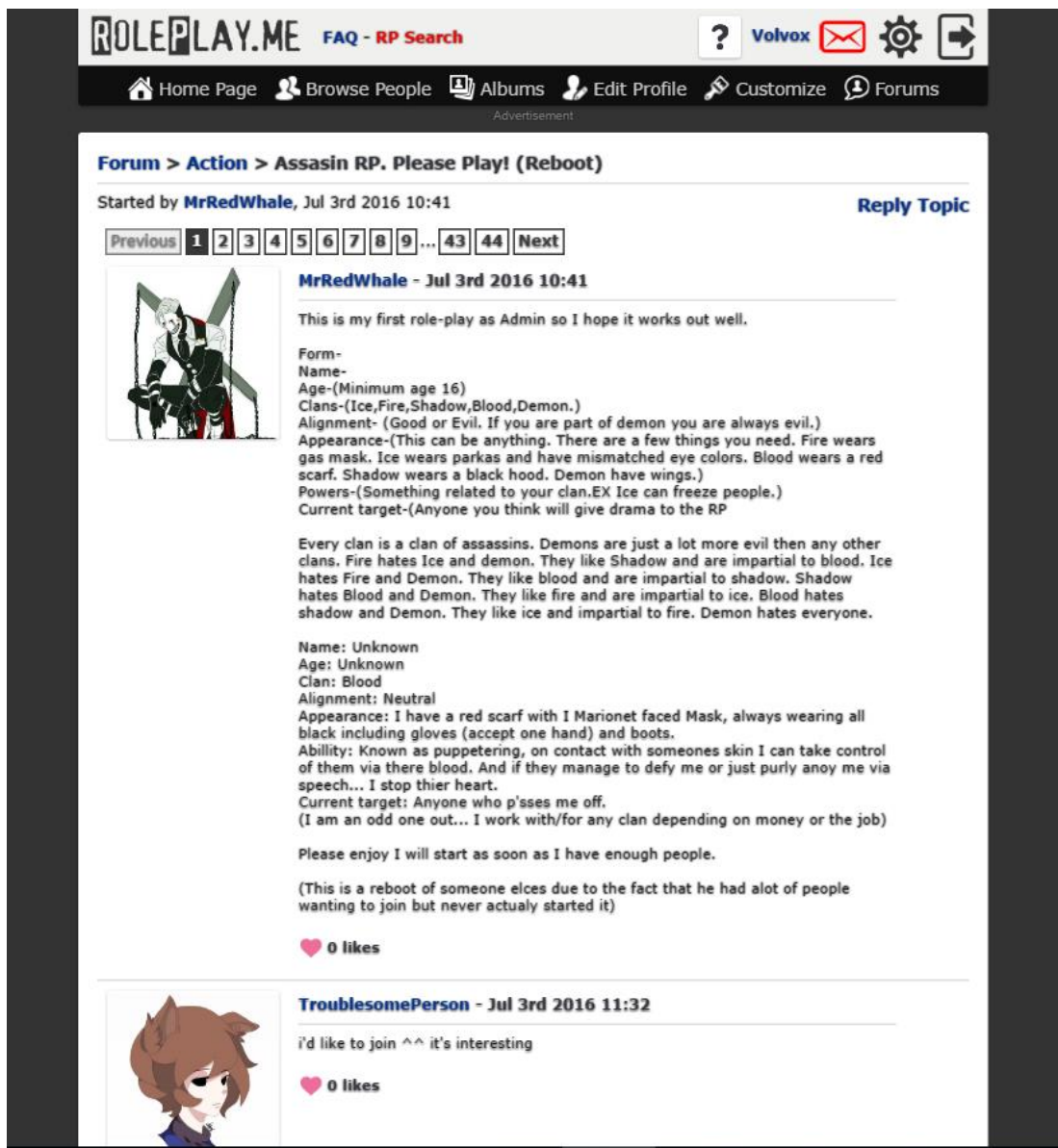


Abbildung 16: Vorstellung von Idee und Charakteren im Forum-Thread

ROLEPLAY.ME [FAQ - RP Search](#) [?](#) [Volvox](#) [✉](#) [⚙](#) [➡](#)


[Home Page](#) [Browse People](#) [Albums](#) [Edit Profile](#) [Customize](#) [Forums](#)

Advertisement

Forum > Action > Assassin RP. Please Play! (Reboot)


Started by **MrRedWhale**, Jul 3rd 2016 10:41 [Reply Topic](#)

[Previous](#) [1](#) [2](#) ... [30](#) [31](#) [32](#) [33](#) [34](#) [35](#) [36](#) ... [43](#) [44](#) [Next](#)

 **Ethelian** - Jul 9th 2016 16:38


"I'll go. My powers aren't super powerful." Aguri said.

♥ 0 likes

 **Ethelian** - Jul 9th 2016 16:43


Aguri blended in as an Ice girl, and walked out in distress. "Excuse me? Where can I find our leader? We got a message not from an ally, but from an enemy!" She expressed, acting exactly like they do. The guy pointed to a certain direction. "Thank you, hopefully I'll see you at a clan meeting." She winked at the Ice guy. And quicker than the speed of light, she went back to herself and pulled out a shadow dagger. She pierced the shadow dagger in his side.

♥ 0 likes

 **ShadedAssassinz** - Jul 9th 2016 16:46


Rin watched, as he flowed out, looking exactly like a shadow. The shadow wrapped around the guy, as he punctured him in his main vital spots. He laughed a little, and looked at Ash. "Your turn."

♥ 0 likes

 **Ethelian** - Jul 9th 2016 16:49

Aguri then blended into the wall behind, in case anyone came.

♥ 0 likes

 **ShadedAssassinz** - Jul 9th 2016 16:50

Rin followed exactly what she did, making the shadows invisible, so it looked like

Abbildung 17: Rollenspiel im gleichen Thread (siehe Abbildung 16)

Über die Suchfunktion der Anwendung kann nach Charakteren gesucht werden. Dabei kann nach bestimmten Merkmalen dieser Charaktere gefiltert werden. Die Suche nach Autoren oder Geschichten ist aber nicht möglich.

Außerdem können verschiedene Nutzerlisten angezeigt werden. Diese umfassen: Gesamtnutzer, neue Nutzer, männliche Charaktere, und weibliche Charaktere.

Die Anwendung unterstützt das Hochladen von verschiedenen Medien – Videos, Bilder und GIFs – in allen Bereichen der Anwendung.

Es gibt die Möglichkeit sowohl private als auch öffentliche Nachrichten zwischen Nutzern über die jeweiligen Profilseiten auszutauschen.

Zudem lassen sich Email-Notifikationen, aber keine Browser-Notifikationen, einstellen. Event-Seiten mit Teilnehmerlisten und Kalendereinträgen können nicht erstellt werden. Events sind in dieser Anwendung nicht vorgesehen.

Insgesamt lässt sich feststellen, dass die Anwendung bereits viele der benötigten Funktionen erfüllt, allerdings aufgrund ihrer mangelhaften Struktur und fehlender Funktionen zur Erstellung und Speicherung von Geschichten, und Verwaltung von mehreren Charakteren nicht für Schreibrollenspiele geeignet ist.

4.4.3.2 RolePlayGateway und OngoingWorlds

RolePlayGateway und OngoingWorlds sind Anwendungen, die für Schreibrollenspiele entwickelt worden sind. Beide Applikationen folgen demselben Prinzip und unterscheiden sich nicht in ihren wesentlichen Funktionen. Daher werden sie in diesem Kapitel nicht separat, sondern als eine Anwendung betrachtet.

Die Anwendung bietet die Möglichkeit unterschiedliche Spielwelten für unterschiedliche Rollenspiele zu entwickeln. Jeder Spielwelt werden verschiedene Charaktere zugewiesen, die nur innerhalb dieser Spielwelt verwendet werden. Die Spielwelt folgt einer Geschichte.

Die Nutzer melden sich ausschließlich mit einem Autor-Account an. Über diese Accounts können die Nutzer Spielwelten und dazugehörige Charaktere entwickeln, oder Charaktere zu Spielwelten anderer Nutzer entwickeln oder auswählen.

Innerhalb einer Spielwelt werden ein oder mehrere Charaktere von jeweils einem Autor repräsentiert. Über die Zuteilung der Charaktere erfolgt zuvor eine Absprache über das Spielwelt-Profil.

Diese Spielwelten entsprechen Foren. Daher verläuft das Verfassen der Geschichten innerhalb dieser Spielwelten, wie in einem Forum (siehe Kapitel 4.4.2.4).

Wie in einem Forum, das für Schreibrollenspiele verwendet wird, gibt es auch in den Spielwelten sogenannte OOC (out of character) Bereiche, in denen Absprachen über die Geschichte getroffen werden können (siehe Abbildung 18). Außerdem gibt es einen Chat-Bereich, über den sich die Autoren austauschen können.

Introduction				Characters				Places				Groups				Arcs				Quests				Posts				Chat				Forum			
The Multiverse: Out Of Character (OOC)																																			
NEWTOPIC*																																			
TOPICS																REPLIES				VIEWS				LAST POST											
The Multiverse: A Renewal by Remæus on Tue Jan 19, 2016 5:19 pm																9				1280				by Eloas on Thu Jun 23, 2016 7:35 am											
Requests for Sovereignty / New Locations by Remæus on Mon Oct 10, 2011 1:39 pm																3				2503				by Remæus on Sun Mar 03, 2013 1:25 am											
The Multiverse: The Official OOC (uber)Thread by Skallagrim on Wed Mar 11, 2009 10:52 am																328				13290				by Peaches-N-Cream on Sun Oct 07, 2012 12:14 am											
Is it Spam? Proper Use of OOC in the Multiverse by Ylanne on Mon Aug 02, 2010 12:16 am																14				2683				by Ylanne on Mon Dec 20, 2010 10:37 pm											
A little guidance...? by Umbrella on Tue Jun 21, 2016 10:41 am																5				62				by lil_kreen on Thu Jul 07, 2016 1:29 pm											
Alternatives to Wing City on Terra by Rulke on Mon Jun 27, 2016 1:57 pm																0				45				by Rulke on Mon Jun 27, 2016 1:57 pm											
Would North Korea exist in the Multiverse? by Cypher7850 on Wed Jun 22, 2016 1:06 pm																3				95				by Cypher7850 on Thu Jun 23, 2016 4:06 pm											
What was the very 1st Multiverse post? by Cypher7850 on Tue Jun 21, 2016 5:13 pm																2				164				by Cypher7850 on Tue Jun 21, 2016 5:56 pm											
A Guide to Roleplaying in the Multiverse by Patcharoo on Fri Jun 10, 2016 7:05 pm																0				139				by Patcharoo on Fri Jun 10, 2016 7:05 pm											

Abbildung 18: OOC-Forum-Bereich in RolePlayGateway

Bestimmte Metadaten, wie eine Vorgeschichte der Spielwelt, die Beschreibung der Spielwelt und eine Charakterliste, können über die Seite der Spielwelt angelegt werden. Werkzeuge für andere Metadaten, z.B. einen Spannungsbogen oder einen Zeitstrahl, gibt es allerdings nicht.

Die gespielten Charaktere und Geschichten eines Autors sind auf dessen Profilseite verlinkt (siehe Abbildung 19).

Die oben beschriebene Anwendung bietet die Möglichkeit einen Autor-Account und mehrere Charaktere anzulegen.

Zudem kann ein Geschichtsprofil, in Form einer Spielwelt, angelegt werden (siehe Abbildung 20). Diese Geschichten können abonniert werden, indem das entsprechende Forum abonniert wird. Die Geschichten können aber nicht archiviert bzw. heruntergeladen und extern gespeichert werden.

RPG Roleplay Forum Chat News Search... 0 Profile Logout [Volvox]

BASIC INFORMATION

Username: **Sovenric**
 Groups: Registered users

Favorite Role Playing Game: Earthbound
 Game Master: No

USER STATISTICS

Joined: Fri Feb 19, 2016 2:21 pm
 Last visited: Fri Aug 12, 2016 5:42 am
 Total posts: [Search user's posts](#)
 (0.00% of all posts / 0.20 posts per day)

CONTACT SOVENRIC

PM: [Send private message](#)


Roleplay Stats


2 created. **0** active. **2** inactive. **0** completed.
0% success ratio.

Writing Stats

24673 words written. **23** total posts.
1072.74 words per post. **6** roleplays played.
3.83 posts per roleplay.
29.17 average days in a roleplay.

Roleplays [View All »](#)

 **PLATELLA**
 One day, one who flies the farthest shall live in the cloud streets and feast on the candied water evaporated upon the sky. An advantage to having wings is that they could meander so far without leaning on something else, and yet they lean on the clouds.

 **OLISSES - CONFINED TO A LEMMATORIUM**
 Every scientist on Soul is confined to a Lemmatorium. Slowly the people of Soul are going insane; learning that immortality is real, and people are able to perform surgery on abstract concepts such as energy, life and cleverness.

Characters [View All »](#)



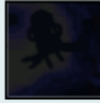


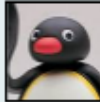





 HEATHIN CASSIE EMERSON English language decalibrator and creatively-minded Stoic	 ANDREW HARRISON THURSDAY A senseless man living a contented life on the streets.	 GUTHER HANDSEN SAM Wishes to go back to a simpler time. An inspector and a luddite.
 JOBE SIMPRA GWANNON "Poison to everyone's ears"; which she's expressed in her numerous Fencing matches	 JOOST CROMLYN OVIES A snappy, red-skinned, megalomania driven daemon	 PINGU THE PENGUIN "noot noot"
 COPPERNICUS OF SYNOPE Designer, planner and a poor listener. The wheel is cognitive which owes its respect to him.	 WOODWORMS One must imagine the woodworm abashed	 STU SWARTE JONEN Stu mends both buildings and forgotten souls
	 EARL ADDISON OPRAY Co-designer with double the leaves of paper	
	 OSCHENHAUER	

Abbildung 19: Profilsseite eines Autors auf RolePlayGateway

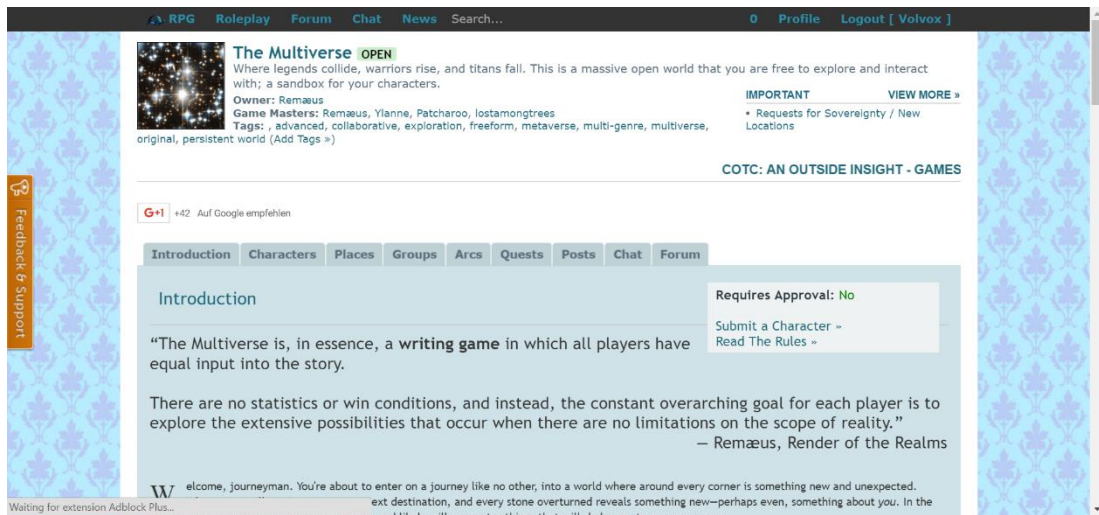


Abbildung 20: Übersichtsseite einer Geschichte auf RolePlayGateway

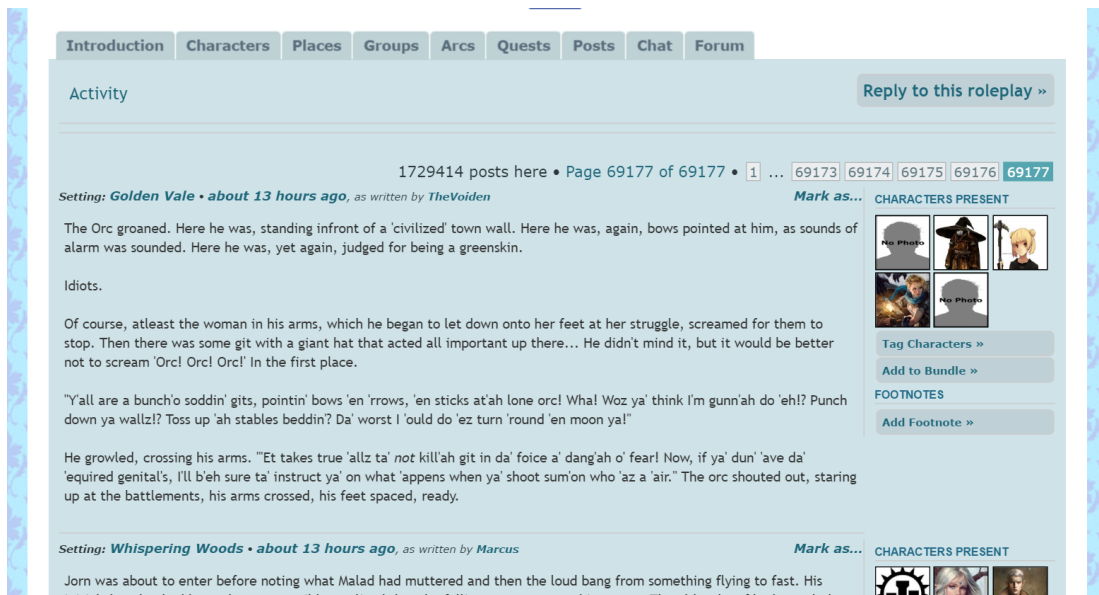


Abbildung 21: Struktur von Geschichtsbeiträgen auf RolePlayGateway

Nutzer können nicht abonniert oder befreundet, oder blockiert werden. Es können aber private Nachrichten zwischen Nutzern versendet werden.

Beiträge können nur im Forum der Spielwelt, in privaten Nachrichten, oder Chats gemacht werden. Daraus folgt, dass lediglich SLs mit der Anwendung verfasst werden können. Solos oder Random RP werden von dieser Anwendung nicht unterstützt.

Es gibt Übersichtsseiten über die Beiträge eines Charakters, aber nicht über die Beiträge eines Autors. Autoren können auch keine Beiträge, z.B. Statusmeldungen, verfassen.

Im Profil und in den Foren können Bilder und GIFs hochgeladen werden. Sie dienen der visuellen Unterstützung der Geschichte bzw. der Spielwelt oder des Charakters.

Eventseiten gibt es nicht. Die Durchführung von Events ist in dieser Anwendung nicht vorgesehen.

Es gibt eine Freitextsuche, die sowohl Charaktere als auch Beiträge und Spielwelten zurückliefert. Allerdings kann nicht nach verschiedenen Merkmalen bzw. Charakteren, und Geschichten, gefiltert werden.

Eine Übersichtsseite bzw. Liste aller registrierten Autoren oder Charaktere gibt es nicht. Allerdings gibt es eine Übersicht aller Spielwelten.

Sowohl Email-Notifikationen als auch Browser-Notifikationen (über Pop-ups) können über die Anwendung eingestellt werden. Diese informieren über neue private Nachrichten oder antworten in einer Geschichte.

Insgesamt lässt sich feststellen, dass diese Web-Anwendung bereits viele Funktionen für Schreibrollenspiele unterstützt, aber dennoch Mängel aufweist. Aufgrund der Foren-Struktur innerhalb der Spielwelten fehlen einige Grundfunktionen. So fehlt die Möglichkeit des Verfassens von Solos und Random RP. Außerdem können Geschichten nicht gespeichert werden.

5 Entwurf

5.1 Anwendungsfälle

Anwendungsfallbeschreibungen und -diagramme dienen der Visualisierung und Beschreibung von Interaktionsmöglichkeiten verschiedener Nutzergruppen mit der Anwendung. Außerdem wird beschrieben welche Folgen die jeweilige Interaktion hat und welche Abhängigkeiten zwischen den Interaktionsmöglichkeiten bzw. Anwendungsfällen existieren. In diesem Kapitel werden ein Anwendungsfalldiagramm der zu entwickelnden Client-Anwendung und die dazugehörigen Anwendungsfallbeschreibungen beschrieben.

5.1.1 Anwendungsfalldiagramm

In diesem Kapitel werden die Anwendungsfälle der zu programmierenden Web-Applikation beschrieben und in einem Diagramm aufgezeigt. Sie dienen der Konzeption von Interaktionsmöglichkeiten des Nutzers mit der Web-Applikation. Beschrieben

werden alle Anwendungsfälle, die sich aus dem Funktionsumfang (siehe Kapitel 4.3) ergeben. In dem Anwendungsfall-Diagramm können lediglich die Grundfunktionen der Anwendung berücksichtigt werden, da das Diagramm sonst zu unübersichtlich würde. Außerdem umfassen die Anwendungsfälle, ebenfalls aufgrund der Übersichtlichkeit, nur Funktionen des Client-Programms.

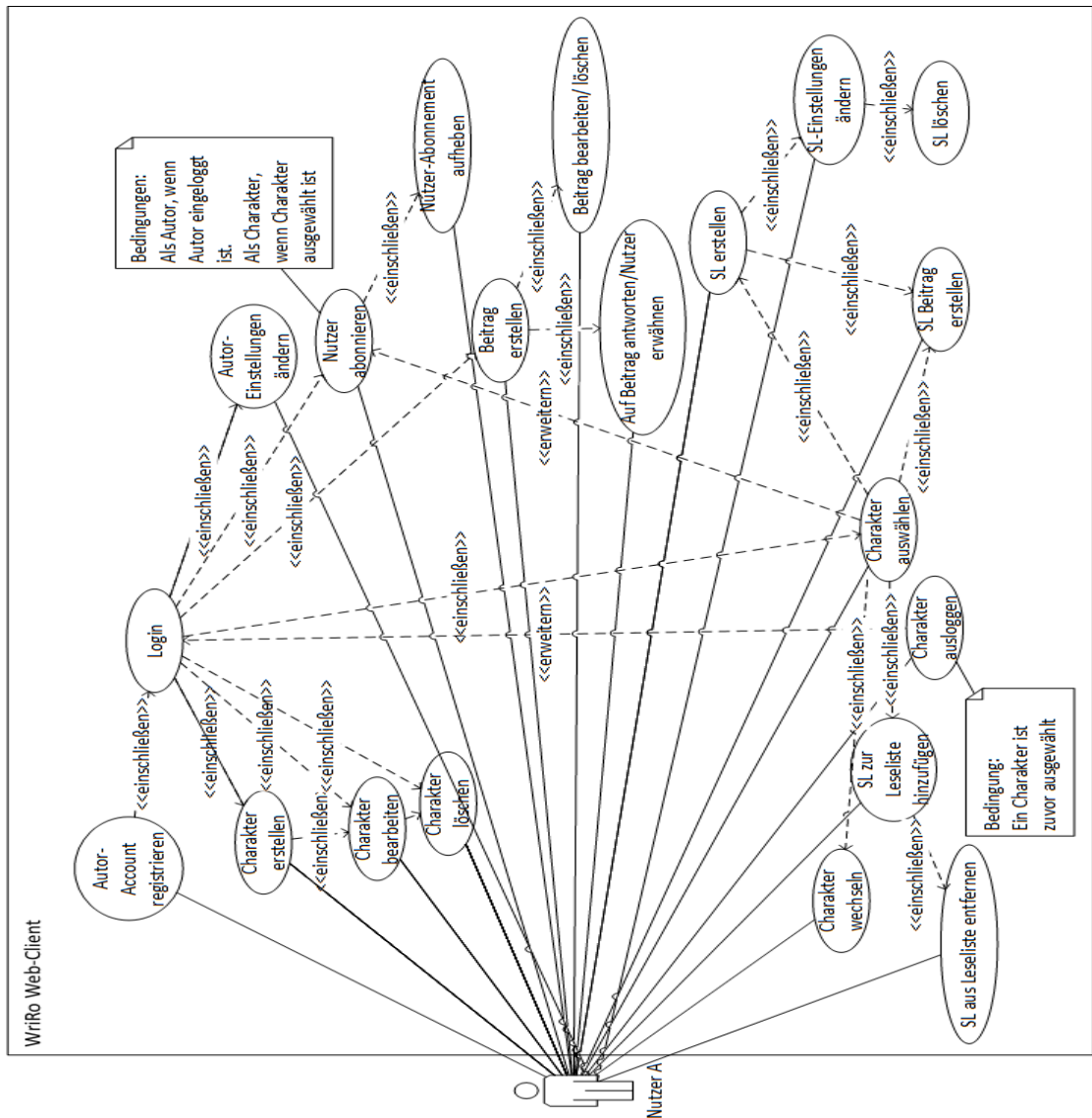


Abbildung 22: Use-Case-Diagramm der Client-Anwendung mit allen Grundfunktionen

5.1.2 Anwendungsfallbeschreibungen

Die Anwendungsfallbeschreibungen in diesem Kapitel ergänzen das im Kapitel 5.1.1 aufgeführte Anwendungsfall-Diagramm. Sie dienen der genauen Beschreibung der Vorgänge und ihrer Abhängigkeiten.

Nr	01
Name	Autor-Account erstellen
Ziel	Ein neuer Autor-Account wird erstellt
Ablauf	<ol style="list-style-type: none"> 1. Autor registrieren Der Nutzer benutzt die "Registrieren"-Schaltfläche der Anwendung. Das System lädt das Registrierungsformular. 2. Formular ausfüllen und abschicken Folgende Felder müssen ausgefüllt werden: Nutzername, Name, Geburtsdatum, Email, Passwort Das System prüft die eingegebenen Daten. Bei success: Die Daten werden in die Datenbank übernommen. Eine Bestätigung wird (auch per Email) gesendet. Der neue Nutzer ist im Autor-Account eingeloggt. (weiter mit 3) Bei failure: Eine Fehlermeldung wird ausgegeben. Das falsche Feld wird markiert. 3. Autor-Profil ausfüllen Folgende Daten können eingegeben werden: Autor-Bild, Header-Bild, Kurzbiografie, Herkunft Das System speichert die Daten in der Datenbank. 4. Account-Einstellungen Folgende Einstellungen können vorgenommen werden: Sprache, Privatsphäre (Öffentlich, nur abonnierte Autoren-Accounts, Privat), Email-Benachrichtigungen, Notifikationen, Altersbeschränkung Das System speichert die Einstellungen.
Vorbedingung	Keine
Ergebnis	Die Anmelde-Daten des neuen Nutzers wurden gespeichert. Das Profil des Autors wurde gespeichert. Alle Einstellungen wurden übernommen.

Tabelle 6: Use-Case 1: Autor-Account erstellen

Nr	02
Name	Autor-Einstellungen ändern
Ziel	Die Einstellungen des Autor-Accounts ändern.
Ablauf	<ol style="list-style-type: none"> Schaltfläche "Bearbeiten" im Autorprofil wählen Das System lädt die Einstellungs-View des Autors. Einstellungen anpassen und speichern Folge Einstellungen können geändert werden: Name, Geburtsdatum, Email, Passwort, Sprache, Privatsphäre (Öffentlich, nur abonnierte Autoren-Accounts, Privat), Email-Benachrichtigungen, Notifikationen, Altersbegrenzung, Profil mit: Autor-Bild, Header-Bild, Kurzbiografie, Herkunft Das System speichert die Einstellungen und sendet eine Bestätigung.
Vorbedingung	Autor ist eingeloggt.
Ergebnis	Die Einstellungen des Autor-Accounts wurden geändert.
Anmerkung	Keine

Tabelle 7: Use-Case 2: Autor-Einstellungen ändern

Nr	03
Name	Charakter erstellen
Ziel	Ein neuer Charakter wird erstellt und dem Autor-Account hinzugefügt
Ablauf	<ol style="list-style-type: none"> Charakter hinzufügen Im Charaktermenü wird "Hinzufügen" gewählt. Das System lädt ein leeres Charakterprofil. Profil ausfüllen Folgende Felder müssen ausgefüllt werden: Name, Charakter-ID (eindeutige Charakterbezeichnung), Status (aktiv/inaktiv), Avatar, Autor-Sichtbarkeit (sichtbar, unsichtbar) Folgende Felder können ausgefüllt werden: Beschreibung, Biografie, Wohnort, Header-Bild, Fandom, Familienmitglieder, Lebensgefährte, Altersbeschränkung Das System speichert alle Einträge in der Datenbank und sendet eine Bestätigung. Nach Anzeige der Bestätigung wird das neue Charakterprofil angezeigt. Der Autor ist mit dem neuen Charakter eingeloggt.
Vorbedingung	Autor ist eingeloggt.
Ergebnis	Ein neuer Charakter wurde erstellt und gespeichert. Der Autor ist mit diesem Profil eingeloggt.

Tabelle 8: Use-Case 3: Charakter erstellen

Nr	04
Name	Charakter bearbeiten
Ziel	Ein Charakterprofil wird nachträglich angepasst.
Ablauf	<ol style="list-style-type: none"> 1. Charakter wählen Charakter wird im Charaktermenü des Autor-Accounts ausgewählt. Das System zeigt das Charakter-Profil an. 2. Bearbeitbares Profil anzeigen Die Edit-Schaltfläche des Profils wird gewählt. Das System zeigt das bearbeitbare Profil an. 3. Profil bearbeiten und speichern Folgende Einstellungen können bearbeitet werden: Name, Status (aktiv/inaktiv), Avatar, Autor-Sichtbarkeit (sichtbar, unsichtbar), Beschreibung, Biografie, Wohnort, Header-Bild, Fandom, Familienmitglieder, Lebensgefährte, Altersbeschränkung Das System speichert alle Änderungen in der Datenbank und sendet eine Bestätigung bei erfolgreicher Anpassung. Das Charakterprofil wird angezeigt.
Vorbedingung	Autor ist eingeloggt.
Ergebnis	Alle vorgenommenen Änderungen wurden gespeichert.
Anmerkung	Die Edit-Schaltfläche ist nur für den Autor des Charakters sichtbar.

Tabelle 9: Use-Case 4: Charakter bearbeiten

Nr	05
Name	Charakter löschen
Ziel	Ein Charakter wird gelöscht
Ablauf	<ol style="list-style-type: none"> 1. Charakter auswählen Charakter wird im Charaktermenü des Autor-Accounts ausgewählt. Das System zeigt das Charakter-Profil an. 2. Löschen anfordern Die "Delete"-Schaltfläche im Header des Charakters wird ausgewählt. Das System sendet eine Bestätigungsanforderung. 3. Bestätigen Das System löscht den Charakter und alle seine Beiträge. Das System aktualisiert alle abonnierten Seiten.

Vorbedingung	Autor ist eingeloggt.
Ergebnis	Charakterseite und alle Einträge des Charakters wurden gelöscht. Alle abonnierten Boards wurden aktualisiert.
Anmerkung	Die "Delete"-Schaltfläche ist nur für den Autor des Charakters sichtbar

Tabelle 10: Use-Case 5: Charakter löschen

Nr	06
Name	Charakter auswählen
Ziel	Ein Charakter wird aus der Liste der Charaktere ausgewählt.
Ablauf	1. Charakter auswählen Charakter wird im Charaktermenü des Autor-Accounts ausgewählt. Das System zeigt die Charakterprofilseite an und loggt den Nutzer in diesen Charakter-Account ein. Die Charakteransichten werden geladen.
Vorbedingung	Autor ist eingeloggt
Ergebnis	Der Nutzer ist als Charakter eingeloggt. Die Charakteransichten wurden geladen.
Anmerkung	Der Nutzer agiert jetzt als Charakter; nicht mehr als Autor. Von hier aus kann er zurück zum Autor-Account oder in einen anderen Charakter wechseln.

Tabelle 11: Use-Case 6: Charakter auswählen

Nr	07
Name	Charakter wechseln
Ziel	Ein anderer Charakter wird gewählt.
Ablauf	1. Charaktermenü öffnen Das System lädt alle Charaktere des Autors. 2. Charakter auswählen (siehe Nr. 06)
Vorbedingung	Charakter ist eingeloggt
Ergebnis	Ein neuer Charakter wurde gewählt. Nutzer ist jetzt als dieser Charakter eingeloggt.
Anmerkung	Keine

Tabelle 12: Use-Case 7: Charakter wechseln

Nr	08
Name	Aus Charakter ausloggen
Ziel	Der Nutzer wird aus allen Charakter-Accounts ausgeloggt.
Ablauf	<ol style="list-style-type: none"> 1. Schaltfläche "Autor" wählen Das System fordert eine Bestätigung. 2. Bestätigen Das System kehrt zum Autor-Account zurück und loggt den Nutzer aus dem aktuellen Charakter aus. Das System lädt die Autor-Ansichten.
Vorbedingung	Charakter ist eingeloggt.
Ergebnis	Der Nutzer ist als Autor eingeloggt.
Anmerkung	Keine

Tabelle 13: Use-Case 8: Aus Charakter ausloggen

Nr	09
Name	Nutzer abonnieren
Ziel	Alle Beiträge des Nutzers (Autor/Charakter) erscheinen zukünftig auf dem Main-Board des Nutzers
Ablauf	<p>1a) Nutzer suchen und anzeigen Folgende Suchkriterien sind möglich: Name, Charakter-ID, Fandom, Status (aktiv/inaktiv). Auch Kombinationen. Oder: Autorname, Username (weiter mit 2)</p> <p>1b) Nutzer auswählen In einer beliebigen Konversation/Story wird die Charakter-ID des Charakters bzw. der Username des Autors angeklickt. Das System zeigt das Charakter- / Autorprofil an. (weiter mit 3)</p> <p>2) Nutzer wählen Aus der Liste von Charakteren bzw. Autoren wird einer gewählt. Das System zeigt das Charakter-/Autorprofil an. (weiter mit 3)</p> <p>3)Nutzer abonnieren Auf der Profilseite wird der Button "abonnieren" geklickt. Das System trägt den Charakter/Autor als abonnierten Charakter/Autor vom Nutzer in der Datenbank ein.</p>
Vorbedingung	Der Nutzer ist mit einem Charakter /Autor eingeloggt
Ergebnis	Der Charakter/Autor wurde abonniert. Alle Beiträge des Charakters/Autors werden künftig im Main-Board des Nutzers angezeigt
Anmerkung	Nur Charaktere/Autoren auf die der Nutzer Leseberechtigung hat werden angezeigt. Nur Autoren können Autoren abonnieren; Nur Charaktere können Charaktere abonnieren.

Tabelle 14: Use-Case 9: Nutzer abonnieren

Nr	10
Name	Nutzer-Abonnement aufheben
Ziel	Einen Nutzer (Charakter / Autor) nicht länger abonnieren. Beiträge des Nutzers nicht länger im Main-Board erhalten.
Ablauf	<ol style="list-style-type: none"> 1. Nutzerprofilseite auswählen Aus der Liste der abonnierten Charaktere / Autoren wird ein Nutzer ausgewählt. Das System lädt die Nutzerprofilseite. 2. Schaltfläche "Abonnieren aufheben" auswählen Das System fordert eine Bestätigung. 3. Bestätigen Das System löscht den Eintrag aus der Datenbank. Der Nutzer wird aus der Liste der abonnierten Nutzer entfernt.
Vorbedingung	Autor / Charakter ist eingeloggt
Ergebnis	Beiträge des Nutzers werden nicht mehr im Main-Board angezeigt. Der Nutzer wird nicht mehr als abonnierter Nutzer angezeigt.
Anmerkung	keine

Tabelle 15: Use-Case 10: Nutzer-Abonnement aufheben

Nr	11
Name	SL zur Leseliste hinzufügen
Ziel	Eine Geschichte zum einfachen Lesen zur Leseliste hinzufügen
Ablauf	<p>1a) Geschichte suchen und auswählen Folgende Suchkriterien sind möglich: Name, mitwirkende Charaktere, Fandom, Altersbegrenzung. Kombinationen sind möglich.</p> <p>Das System zeigt alle Geschichten an, die den Suchkriterien entsprechen. Der Nutzer wählt eine Geschichte aus der Liste aus. Das System zeigt das Geschichts-Board an.</p> <p>1b) Geschichte auswählen Eine Geschichte wird im Charakterprofil ausgewählt. Das System zeigt das Geschichts-Board an.</p> <p>2) Geschichte abonnieren Die Abonnieren-Schaltfläche des Geschichts-Boards wird ausgewählt. Das System fügt die Geschichte zur Leseliste des Charakters hinzu. Das Geschichts-Board wird angezeigt.</p>

Vorbedingung	Charakter ist eingeloggt.
Ergebnis	Die Geschichte wird in der Leseliste angezeigt und ständig aktualisiert.
Anmerkung	In der Suche und auf Charakterprofilen werden nur Geschichten angezeigt, zu denen der eingeloggte Charakter eine Leseberechtigung hat.

Tabelle 16: Use-Case 11: SL zur Leseliste hinzufügen

Nr	12
Name	SL aus Leseliste entfernen
Ziel	Die SL erscheint nicht länger in der Leseliste.
Ablauf	<ol style="list-style-type: none"> 1. Geschichts-Board auswählen Eine Geschichte wird aus der Leseliste ausgewählt. Das System lädt das Geschichts-Board 2. Geschichts-Profil öffnen Das System lädt das Geschichts-Profil. 3. Schaltfläche "Abonnieren aufheben" auswählen Das System fordert eine Bestätigung. 4. Bestätigung Das System löscht den Eintrag aus der Datenbank. Das Geschichts-Board wird aus der Leseliste entfernt.
Vorbedingung	Charakter ist eingeloggt
Ergebnis	Die SL wurde aus der Leseliste gelöscht.
Anmerkung	Keine

Tabelle 17: Use-Case 12: SL aus Leseliste entfernen

Nr	13
Name	SL erstellen
Ziel	Ein neues Geschichts-Board wird für eine Geschichte erstellt
Ablauf	<ol style="list-style-type: none"> 1. SL hinzufügen Auf der Charakterseite wird die SL hinzufügen - Schaltfläche ausgewählt. Das System öffnet ein leeres Geschichts-Profil. 2. Geschichts-Profil ausfüllen und speichern Folgende Felder müssen ausgefüllt werden: Titel, mitwirkende Charaktere (Ids), Status (Fortlaufend, suchend), Sichtbarkeit (privat, öffentlich) Folgende Felder können zusätzlich ausgefüllt werden: Fandom, Kurzbeschreibung, Altersbeschränkung

	Das System speichert die neue SL mit allen Einstellungen in der Datenbank. Als Bestätigung wird das neue Geschichts-Board angezeigt. Das Geschichts-Board wird zu allen Profilen der mitwirkenden Charaktere hinzugefügt.
Vorbedingung	Charakter ist eingeloggt.
Ergebnis	Ein neues Geschichts-Board wurde bei allen beteiligten Charakteren erstellt.
Anmerkungen	Fortlaufend: Die Geschichte ist in Bearbeitung; Suchend: Es werden noch Charaktere gesucht, die sich an der Geschichte beteiligen wollen; privat: Nur mitwirkende Charaktere können die Geschichte sehen; öffentlich: Alle Charaktere (abhängig von den Privatsphäre-Einstellungen der beteiligten Charaktere) können die Geschichte lesen

Tabelle 18: Use-Case 13: SL erstellen

Nr	14
Name	SL- Einstellungen ändern
Ziel	Einstellungen zur Geschichte werden geändert.
Ablauf	<ol style="list-style-type: none"> 1. Geschichts-Board auswählen Im Charakterprofil wird eine Geschichte ausgewählt. Das System zeigt das Geschichts-Board an. 2. Geschichts-Profil öffnen Im Headerbereich des Geschichts-Boards wird die Schaltfläche "Edit" ausgewählt. Das System öffnet das Geschichts-Profil. 3. Geschichts-Profil bearbeiten und speichern Folgende Optionen sind bearbeitbar: Titel, Mitwirkende Charaktere, Status (fortlaufend, suchend, abgeschlossen), Sichtbarkeit, Altersbeschränkung, Fandom, Kurzbeschreibung <p>Das System speichert alle geänderten Daten in der Datenbank und sendet eine Bestätigung. Das System kehrt zur Geschichts-Board View zurück.</p>
Vorbedingung	Charakter ist eingeloggt und als mitwirkender Charakter eingetragen.
Ergebnis	Einstellungen der Geschichte wurden geändert.
Anmerkung	Die Edit-Schaltfläche ist nur für berechtigte Nutzer sichtbar

Tabelle 19: Use-Case 14: SL-Einstellungen ändern

Nr	15
Name	SL löschen
Ziel	Entfernen eines Geschichts-Boards.
Ablauf	<ol style="list-style-type: none"> 1. Geschichts-Board auswählen Im Charakterprofil wird eine Geschichte ausgewählt. Das System zeigt das Geschichts-Board an. 2. Geschichts-Profil öffnen Im Headerbereich des Geschichts-Boards wird die Schaltfläche "Edit" ausgewählt. Das System öffnet das Geschichts-Profil. 3. Geschichte löschen Die Schaltfläche "Delete" wird ausgewählt. Das System fordert eine Bestätigung. 4. Bestätigen Das System löscht das Geschichts-Board aus allen Charakterprofilen und aus der Datenbank. Das System sendet eine Bestätigung und kehrt zur Charakterprofil View zurück.
Vorbedingung	Charakter ist eingeloggt und als mitwirkender Charakter eingetragen.
Ergebnis	Das Geschichts-Board wurde gelöscht. Alle abonnierten Boards wurden aktualisiert.
Anmerkung	Die Delete-Schaltfläche ist nur für berechtigte Nutzer sichtbar

Tabelle 20: Use-Case 15: SL löschen

Nr	16
Name	Neuer Beitrag zu einer vorhandenen SL
Ziel	Ein neuer Beitrag auf einem Geschichts-Board wird verfasst.
Ablauf	<p>1a) Geschichts-Board wählen Das System lädt das Geschichts-Board. (weiter mit 2)</p> <p>2) Schaltfläche "Neuer Beitrag" wählen Das System öffnet ein leeres Textfeld. (weiter mit 3)</p> <p>1b) Auf den letzten Beitrag des Geschichts-Boards antworten Der Nutzer wählt die Schaltfläche "Antworten" in den Beitragsdetails des letzten Beitrags des Geschichts-Boards (z.B. von seinem Main-Board und Charakterprofil aus). Das System öffnet ein leeres Textfeld. (weiter mit 3)</p> <p>3) Beitrag eingeben und abschicken Der Nutzer gibt einen neuen Texteintrag ein, und / oder lädt ein Bild-Medium hoch (siehe Medien-Upload). Das System prüft den Beitrag, speichert ihn und verlinkt ihn mit der Geschichte.</p>

	Der Beitrag erscheint im Geschichts-Board. Alle Main-Boards und Leselisten der Abonnenten werden aktualisiert.
Vorbedingung	Charakter ist eingeloggt und als mitwirkender Charakter im Geschichts-Board eingetragen.
Ergebnis	Ein neuer Beitrag zur SL wurde erstellt. Alle Boards der Abonnenten wurden aktualisiert.
Anmerkung	Die Schaltflächen "Neuer Beitrag" oder "Antworten" sind nur für berechtigte Nutzer sichtbar.

Tabelle 21: Use-Case 16: Neuer Beitrag zu einer vorhandenen SL

Nr	17
Name	SL archivieren
Ziel	Eine Geschichte wird als PDF heruntergeladen.
Ablauf	<ol style="list-style-type: none"> 1. Geschichts-Board wählen Der Nutzer wählt ein Geschichts-Board aus der Lister seiner Geschichten aus. Das System lädt das Geschichts-Board 2. Geschichts-Profil öffnen Das System lädt das Geschichts-Profil. 3. Schaltfläche "Archivieren" wählen Das System schreibt alle Beiträge des Geschichts-Board in ein PDF-Dokument und lädt dieses auf den Rechner des Nutzers.
Vorbedingung	Charakter eingeloggt
Ergebnis	Die gesamte Geschichte wurde heruntergeladen.
Anmerkung	Die Schaltfläche "Archivieren" ist nur für berechtigte Nutzer sichtbar

Tabelle 22: Use-Case 17: SL archivieren

Nr	18
Name	Beitrag erstellen
Ziel	Ein neuer Beitrag wird erstellt
Ablauf	<ol style="list-style-type: none"> 1. Board auswählen Das Board wird ausgewählt, auf welchem der Beitrag erscheinen soll Zur Auswahl stehen: News-Board (Autor), Random-Board (Charakter), Event-Board (Charakter) und ein beliebiges Story-Board (Charakter)

	<p>Das System zeigt das gewählte Board an.</p> <p>2. Eintrag hinzufügen Der Nutzer fügt mit "neuer Beitrag" einen neuen Beitrag hinzu. Das System öffnet ein leeres Textfeld.</p> <p>3. Eintrag erstellen und abschicken Nutzer gibt den Textbeitrag ein, und / oder lädt ein Bild-Medium hoch(siehe Medium-Upload). Mit Senden des Beitrags wird dieser abgeschickt. Das System prüft den Eintrag und speichert ihn in der Datenbank. Bei success gibt das System eine Bestätigung zurück und führt ein Update auf allen Main-Boards der Abonnenten durch, bei failure gibt das System eine Fehlermeldung zurück.</p>
Vorbedingung	Der Nutzer ist eingeloggt.
Ergebnis	Bei success: Ein neuer Eintrag ist im entsprechenden Board erstellt. Alle Boards der Abonnenten wurden aktualisiert.
Anmerkungen	Die "neuer Beitrag"-Schaltfläche ist nur für berechtigte Nutzer sichtbar

Tabelle 23: Use-Case 18: Beitrag erstellen

Nr	19
Name	Beitrag editieren
Ziel	Ein Beitrag wird bearbeitet / nachträglich geändert
Ablauf	<p>1. Beitrag öffnen Die Edit-Schaltfläche des entsprechenden Beitrags wird ausgewählt. Das System öffnet das Textfeld des Beitrags.</p> <p>2. Beitrag bearbeiten und speichern Das System prüft die Änderungen und überträgt sie auf die Datenbank. Das System aktualisiert alle Boards der Abonnenten.</p>
Vorbedingung	Der Autor / Charakter ist eingeloggt und berechtigt zum Editieren des Beitrags
Ergebnis	Beitrag wurde geändert; Alle Boards der Abonnenten wurden aktualisiert.
Anmerkungen	Die Edit-Schaltfläche ist nur für berechtigte Nutzer sichtbar

Tabelle 24: Use-Case 19: Beitrag editieren

Nr	20
Name	Beitrag löschen
Ziel	Beitrag wird aus allen Boards entfernt
Ablauf	<ol style="list-style-type: none"> 1. Beitrag löschen anfordern Die Delete-Schaltfläche des entsprechenden Beitrags wird ausgewählt. Das System fordert eine Bestätigung des Nutzers. 2. Bestätigen Das System entfernt den Beitrag vom Board und löscht den entsprechenden Eintrag in der Datenbank. Alle Boards der Abonnenten werden aktualisiert.
Vorbedingung	Autor / Charakter ist eingeloggt und berechtigt zum Löschen des Beitrags
Ergebnis	Der Beitrag wurde aus allen Boards entfernt.
Anmerkungen	Die Delete-Schaltfläche ist nur für berechtigte Nutzer sichtbar

Tabelle 25: Use-Case 20: Beitrag löschen

Nr	21
Name	Medien-Upload
Ziel	Upload eines Bildes oder GIFs
Ablauf	<ol style="list-style-type: none"> 1. Uploadfenster öffnen Die Upload-Schaltfläche wird in einem Beitrag oder im Profil / Einstellungen ausgewählt. Das System öffnet das Dateiverzeichnis des Nutzers. 2. Datei wird ausgewählt Der Nutzer wählt eine Datei aus dem Dateiverzeichnis und bestätigt seine Auswahl. Das System lädt eine Vorschau des Bildes / GIFs. 3. Upload bestätigen Das System lädt die Datei auf den Server und macht einen Eintrag in der Datenbank. Das Bild / GIF wird im Profil oder im Beitrag angezeigt.
Vorbedingung	Autor / Charakter ist eingeloggt
Ergebnis	Ein Bild oder GIF wurde hochgeladen.
Anmerkungen	Bei Abbruch des Upload-Prozesses kehrt das System zur vorherigen View zurück und keine Datei wird gespeichert.

Tabelle 26: Use-Case 21: Medien-Upload

Nr	22
Name	Einen anderen Nutzer erwähnen
Ziel	Einen anderen Nutzer (Autor / Charakter) erwähnen
Ablauf	<ol style="list-style-type: none"> 1. Neuer Beitrag Schaltfläche „neuer Beitrag“ wählen. Das System öffnet ein leeres Textfeld. 2. Nutzerkennung eingeben Im Laufe des Textes wird mit @Username bzw. @CharakterID auf den jeweiligen Nutzer verwiesen. Das System schlägt Nutzernamen bzw. CharakterIDs aus der Liste der Abonnierten vor, sowie Nutzernamen, die ähnlich den bereits eingegebenen Namen sind. 3. Text eingeben und abschicken Das System prüft die Nachricht und speichert sie. Das System sendet die Nachricht an den erwähnten Nutzer. Die Nachricht erscheint im Mitteilungs-Board des Nutzers.
Vorbedingung	Autor / Charakter ist eingeloggt
Ergebnis	Es wurde eine (öffentliche) Nachricht an den Autor bzw. Charakter gesendet.
Anmerkung	Nutzererwähnungen sind nur im Main-Board / News-Board und Random-Board möglich.

Tabelle 27: Use-Case 22: Einen anderen Nutzer erwähnen

Nr	23
Name	Auf einen Beitrag antworten
Ziel	Auf einen beliebigen Beitrag im Main-Board / Random-Board / News-Board /Mitteilungs-Board antworten
Ablauf	<ol style="list-style-type: none"> 1. Beitrag auswählen Der Nutzer (Autor / Charakter) wählt den Beitrag aus dem entsprechenden Board aus. Das System lädt die Beitragsdetails. 2. Schaltfläche "Antworten" auswählen Das System lädt ein neues Textfeld, in welchem der Nutzer des Beitrags mit @Username oder @CharakterID erwähnt wird. 3. Nachricht eingeben und abschicken Das System speichert den neuen Beitrag und verlinkt ihn mit dem alten Beitrag. Das System sendet die Nachricht an das Mitteilungs-Board des erwähnten Nutzers, an das Random-Board des antwortenden Users und an das Main-Board aller Abonnenten des antwortenden Nutzers. Das System aktualisiert alle Main-Boards der Abonnenten.

Vorbedingung	Autor / Charakter ist eingeloggt
Ergebnis	Die Antwort zu einem Beitrag wurde gespeichert und verlinkt. Alle Abonnenten haben den aktuellen Stand.
Anmerkung	Keine

Tabelle 28: Use-Case 23: Auf einen Beitrag antworten

Nr	24
Name	Private Nachricht senden
Ziel	Einem anderen Nutzer (Autor/Charakter) eine private Nachricht senden
Ablauf	<p>1a) Das eigene Postfach öffnen Das System öffnet das Postfach mit allen alten Nachrichtendialogen. Wenn es bereits einen alten Dialog mit dem Nutzer gibt, weiter mit 2a. Wenn nicht, weiter mit 2b.</p> <p>2a) Nachrichtendialog auswählen Es wird ein Nachrichtendialog aus der Liste ausgewählt. Das System lädt den Nachrichtendialog mit allen alten Nachrichten. (weiter mit 3)</p> <p>2b) Neue Nachricht erstellen und Nutzer adressieren Im Adressfeld auf den Nutzer verweisen mit @Username bzw. @CharakterID. Das System öffnet einen neuen Nachrichtendialog und schlägt Nutzernamen bzw. CharakterIDs aus der Liste der Abonnierten vor. (weiter mit 3)</p> <p>1b) Schaltfläche "private Nachricht senden" im Profil des Nutzers wählen Das System öffnet einen neuen Nachrichtendialog, in dem der Nutzername bereits im Adressfeld eingetragen ist. Wenn es bereits einen Nachrichtendialog gibt, wird dieser geladen. (weiter mit 3)</p> <p>3) Nachricht eingeben und absenden Das System prüft die Nachricht und speichert sie. Das System sendet die Nachricht ins Postfach des erwähnten Nutzers.</p>
Vorbedingung	Autor / Charakter ist eingeloggt. Beide Nutzer haben sich gegenseitig abonniert.
Ergebnis	Eine private Nachricht wurde zwischen zwei Nutzern versendet. Ein neuer Nachrichtendialog wurde erstellt.
Anmerkung	Es können nur private Nachrichten gesendet werden, wenn sich beide Nutzer gegenseitig abonniert haben.

Tabelle 29: Use-Case 24: Private Nachricht senden

Nr	25
Name	Freitextsuche
Ziel	Mit Hilfe der Freitextsuche wird ein Charakter, ein Autor oder eine Geschichte gesucht.
Ablauf	<ol style="list-style-type: none"> Freitextsuche öffnen Der Nutzer öffnet die Freitextsuche über die entsprechende Schaltfläche auf einer beliebigen Seite. Das System lädt das Suchfeld. Suchbegriffe eingeben und abschicken Das System lädt alle Stories, Autoren und Charaktere, die öffentlich sind und den Suchbegriffen entsprechen. Suche verfeinern Filter: Charaktere, Geschichten, Beiträge, Autoren Mögliche Suchkriterien einstellen: Charaktername, CharakterID, Charakter-Eigenschaften, Autorname, Autor-Username, Geschichts-Titel, Beitragsinhalt, Altersbegrenzung, mitwirkende Autoren, Geschichts-Status, Account-Status, Geschichts-Beschreibung, etc. Das System lädt alle Einträge, die den Sucheinstellungen in Kombination mit dem Freitext entsprechen.
Vorbedingung	Autor / Charakter eingeloggt.
Ergebnis	Eine Liste von Einträgen, die der Suche entsprechen wurde geladen.
Anmerkung	Im Autor-Account kann der Nutzer nur nach Autorspezifischen Inhalten (z.B. andere Autoren) suchen. Im Charakter-Account kann der Nutzer nur nach Charakterspezifischen Inhalten (also keine Autoren, etc., aber z.B. Geschichten) suchen.

Tabelle 30: Use-Case 25: Freitextsuche

Nr	26
Name	Event anlegen
Ziel	Eine neue Eventseite wird erstellt und ein Event eröffnet.
Ablauf	<ol style="list-style-type: none"> Event hinzufügen Die Hinzufügen-Schaltfläche im Eventbereich wird ausgewählt. Das System lädt ein leeres Eventprofil Profil ausfüllen und speichern Folgende Felder müssen ausgefüllt werden: Eventname, Datum und Uhrzeit des Events, Organisator/en (CharakterID), Event-Art (offen, geschlossen) Folgende Felder können ausgefüllt werden: Event-Ort, Dresscode, persönliche Einladung

	Das System speichert die Daten. Datum und Uhrzeit werden mit Zeitzone des Benutzers gespeichert. Das System erstellt eine neue Eventseite und listet sie im Profil des Organizers auf.
Vorbedingung	Charakter ist eingeloggt
Ergebnis	Ein neues Event wurde erstellt
Anmerkung	Geschlossene Events können nur von geladenen Gästen besucht werden. Nur sie können das Event sehen.

Tabelle 31: Use-Case 26 Event anlegen

Nr	27
Name	Event bearbeiten
Ziel	Die Daten eines Events werden nachträglich geändert.
Ablauf	<ol style="list-style-type: none"> 1. Event auswählen Aus der Liste von Events im Eventbereich wird ein Event ausgewählt. Das System zeigt die Profilseite des Events 2. Profil bearbeiten und speichern Folgende Felder können bearbeitet werden: Eventname, Datum und Uhrzeit des Events, Event-Art (offen, geschlossen), Event-Ort, Dresscode, persönliche Einladung <p>Das System speichert alle Änderungen und gibt eine Bestätigung zurück.</p>
Vorbedingung	Charakter ist eingeloggt und Organisator des Events
Ergebnis	Das Event wurde bearbeitet
Anmerkung	Nur berechtigte Nutzer können die Schaltfläche "Bearbeiten" im Profil sehen.

Tabelle 32: Use-Case 27: Event bearbeiten

Nr	28
Name	Gäste zu einem Event einladen
Ziel	Es wird eine Gästeliste für ein Event erstellt bzw. erweitert.
Ablauf	<ol style="list-style-type: none"> 1. Event auswählen Aus der Liste von Events im Eventbereich wird ein Event ausgewählt. Das System zeigt die Profilseite des Events 2. Gästeliste anzeigen Die Schaltfläche "Gästeliste" wird ausgewählt Das System lädt die Gästeliste.

	<p>3. Gäste suchen Es kann nach Charakternamen oder CharakterID gesucht werden. Das System lädt eine Liste von Charakteren, die den Suchbegriffen entsprechen.</p> <p>4. Gäste hinzufügen Aus der Liste von Charakteren werden alle Charaktere ausgewählt, die eingeladen werden sollen. Das System fordert eine Bestätigung.</p> <p>5. Bestätigen Das System fügt die Charaktere zur Gästeliste hinzu und sendet ihnen eine Einladung.</p>
Vorbedingung	Charakter ist eingeloggt.
Ergebnis	Ein Gast / Gäste wurde/n zum Event eingeladen. Die Gästeliste wurde aktualisiert.
Anmerkung	Bei geschlossenen Events können nur Organisatoren Gäste hinzufügen. Bei offenen Events kann jeder Gäste hinzufügen.

Tabelle 33: Use-Case 28: Gäste zu einem Event einladen

Nr	29
Name	Gäste aus der Gästeliste entfernen
Ziel	Ein Gast /Gäste wird/werden aus der Gästeliste eines Events entfernt.
Ablauf	<p>1. Event auswählen Aus der Liste von Events im Eventbereich wird ein Event ausgewählt. Das System zeigt die Profilseite des Events</p> <p>2. Gästeliste anzeigen Die Schaltfläche "Gästeliste" wird ausgewählt Das System lädt die Gästeliste.</p> <p>3. Gast auswählen Aus der Liste wird/werden ein Gast/mehrere Gäste ausgewählt. Das System markiert die ausgewählten Gäste.</p> <p>4. Gast löschen Die Schaltfläche "Löschen" wird ausgewählt. Das System fordert eine Bestätigung.</p> <p>5. Bestätigen Das System löscht die Charaktere aus der Gästeliste und sendet ihnen eine Benachrichtigung.</p>
Vorbedingung	Charakter ist eingeloggt und Organisator des Events
Ergebnis	Gäste wurden von der Gästeliste gelöscht.
Anmerkung	Nur die Organisatoren können Gäste löschen. Nur sie können die Schaltfläche "Löschen" sehen.

Tabelle 34: Use-Case 29: Gäste aus der Gästeliste entfernen

Nr	30
Name	Beitrag als Favorit markieren
Ziel	Ein beliebiger Beitrag wird als Favorit markiert und in der Favoritenliste des Charakters aufgeführt.
Ablauf	<ol style="list-style-type: none"> 1. Beitrag auswählen Ein Beitrag wird aus einer Liste von Beiträgen in einem Board ausgewählt. Das System zeigt die Beitragsdetails an. 2. Stern-Schaltfläche Die Stern-Schaltfläche wird ausgewählt. Das System markiert den Beitrag als Favoriten (Stern wird gelb). Das System speichert den Beitrag in der Liste der Favoriten. Das System benachrichtigt den Autor des Beitrags.
Vorbedingung	Charakter ist eingeloggt.
Ergebnis	Ein Beitrag wurde zur Favoritenliste des Charakters hinzugefügt und als Favorit markiert. Der Autor des Beitrags wurde darüber informiert.
Anmerkung	Keine

Tabelle 35: Use-Case 30: Beitrag als Favorit markieren

5.2 Sequenzdiagramme

Die folgenden Sequenzdiagramme stellen die technischen Abläufe innerhalb der Web-Applikation, sowie die Kommunikation zwischen dem Client und dem Server, dar. Da die Aufführung aller Anwendungsfälle als Sequenzdiagramme den Rahmen dieser Arbeit sprengen würde, wurden nur die wichtigsten Anwendungsfälle betrachtet. Aufgrund der Client-Server-Architektur, der Kommunikation über REST, und den modularen Strukturen des Frontend und Backend, können diese Sequenzdiagramme repräsentativ für alle Anwendungsfälle aufgefasst werden. Die Abläufe sind stets ähnlich.

Das wichtigste Sequenzdiagramm (siehe Abbildung 23) stellt die gesamte Kommunikation – vom Nutzer bis zur Datenbank – dar. In den folgenden Sequenzdiagrammen wurde auf eine genauere Betrachtung der Abläufe im Backend verzichtet, um eine bessere Übersicht zu schaffen. Der Ablauf im Backend ist stets ähnlich – es werden lediglich andere Klassen verwendet – und kann daher im Sequenzdiagramm zur Erstellung eines Autor-Accounts (siehe Abbildung 23) als repräsentativ betrachtet werden.

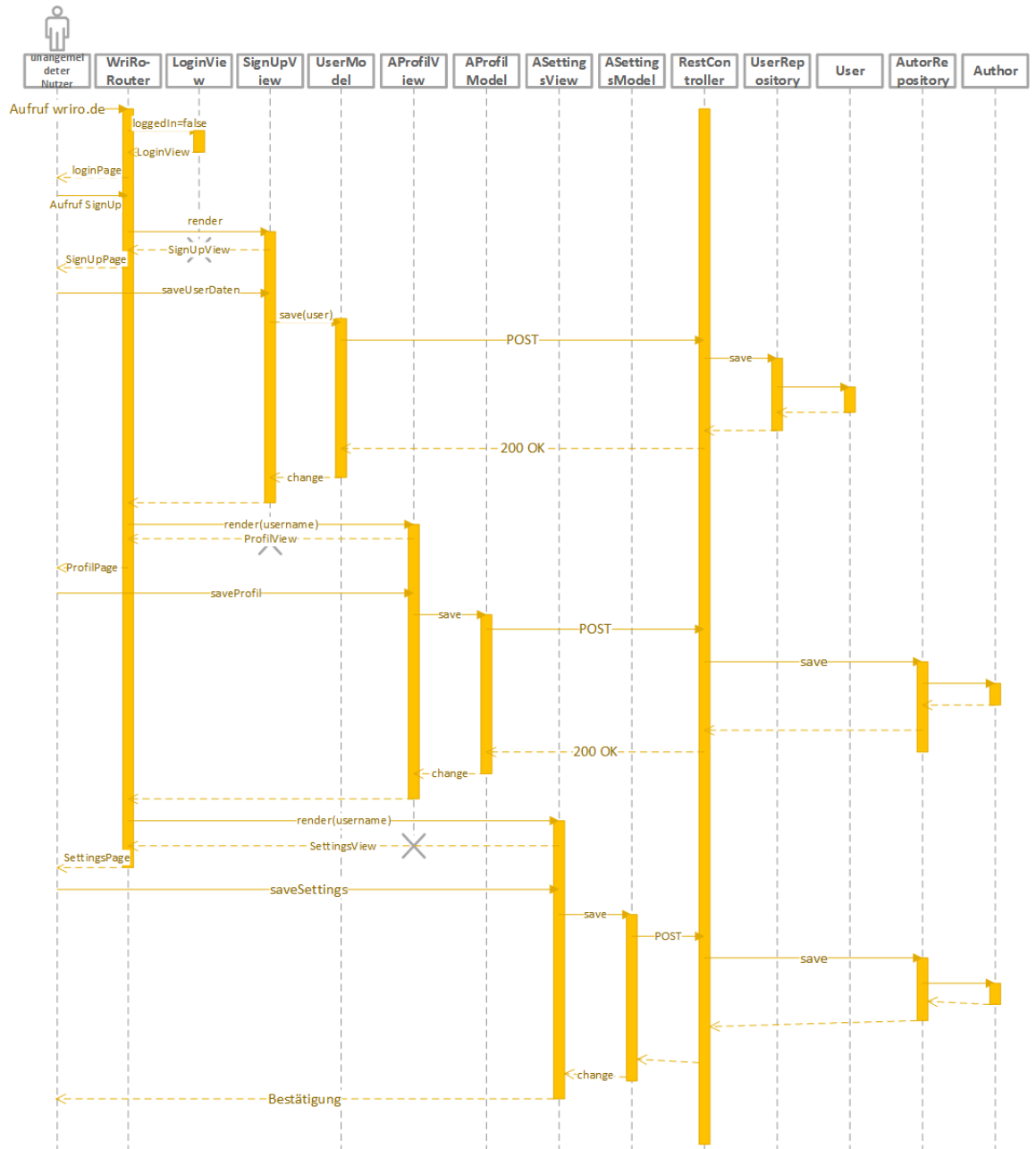


Abbildung 23: Sequenzdiagramm zum Anlegen eines neuen Accounts

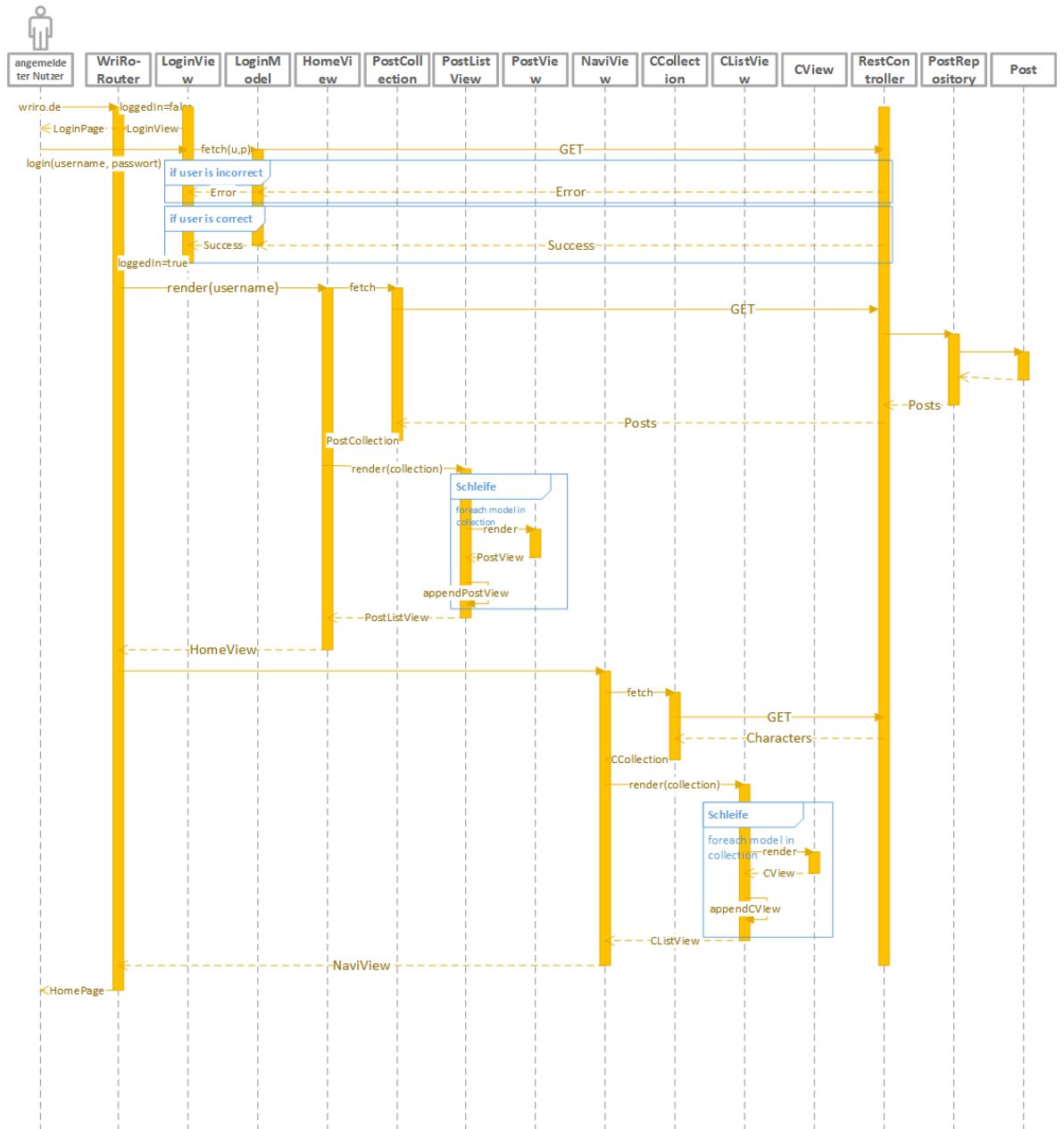


Abbildung 24: Sequenzdiagramm zum Login eines Nutzers

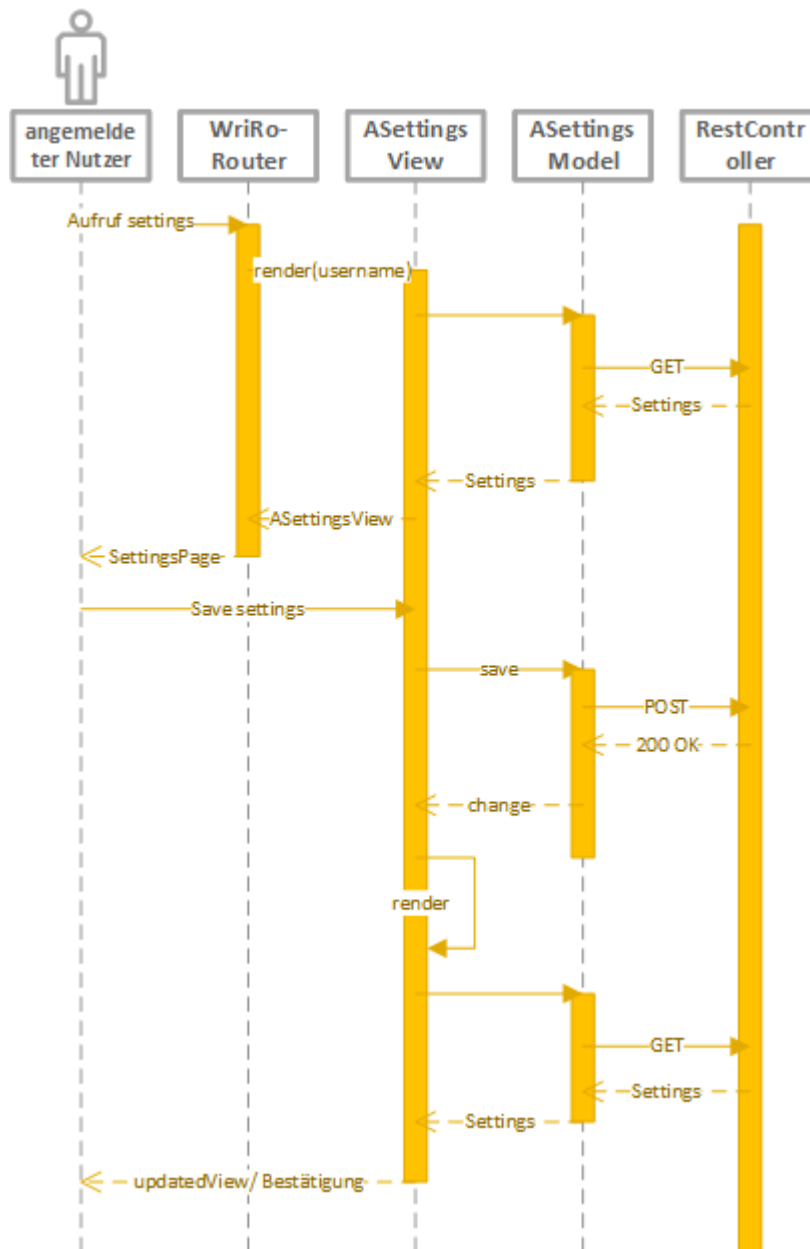


Abbildung 25: Sequenzdiagramm zum Ändern von Autor-Einstellungen

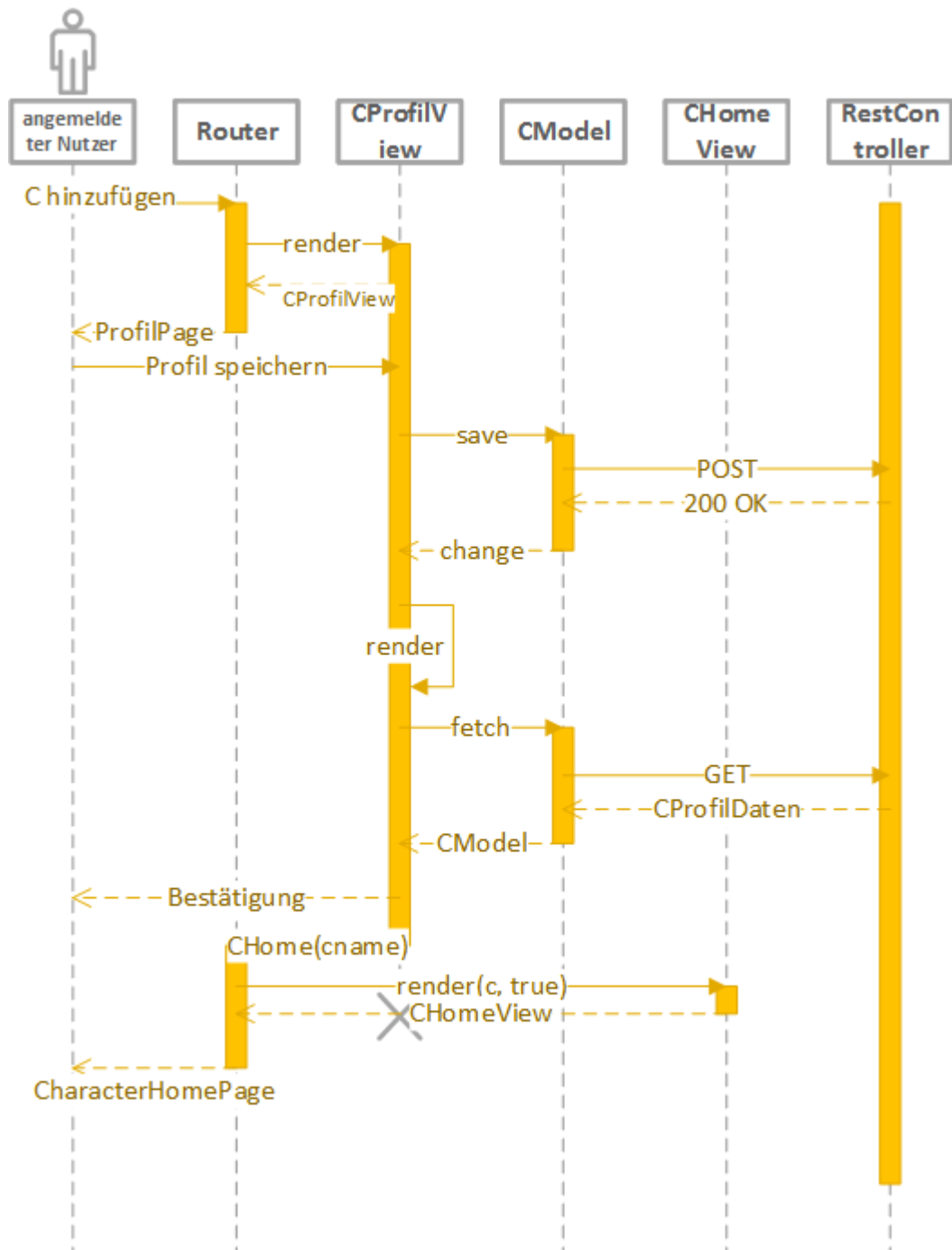


Abbildung 26: Sequenzdiagramm zum Erstellen eines neuen Charakters

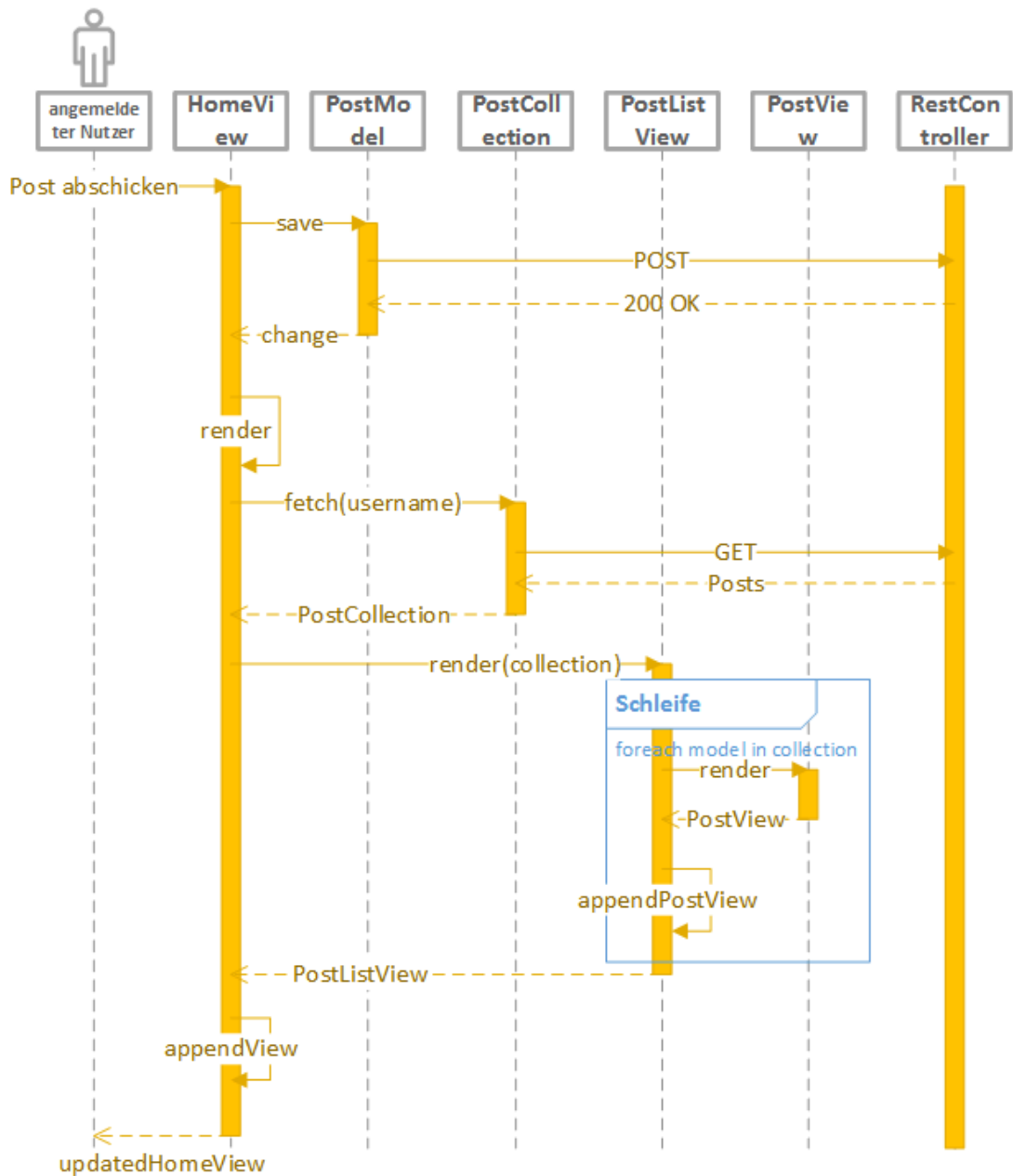


Abbildung 28: Sequenzdiagramm zum Erstellen eines neuen Beitrages

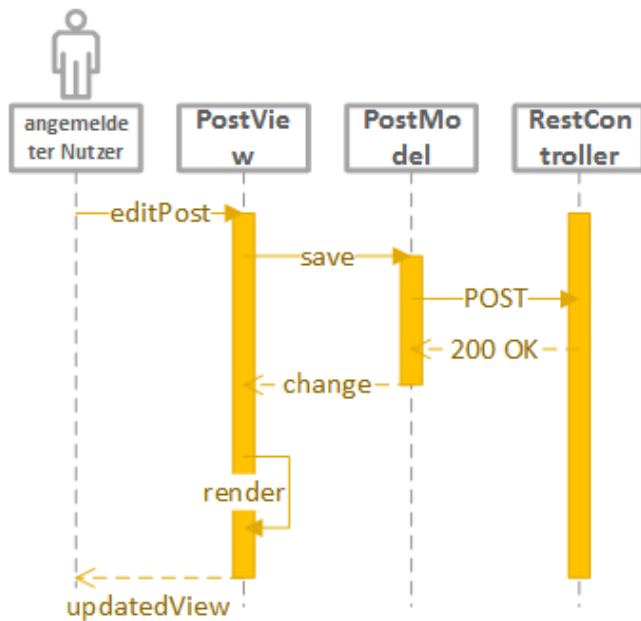


Abbildung 29: Sequenzdiagramm zum Editieren eines Beitrages

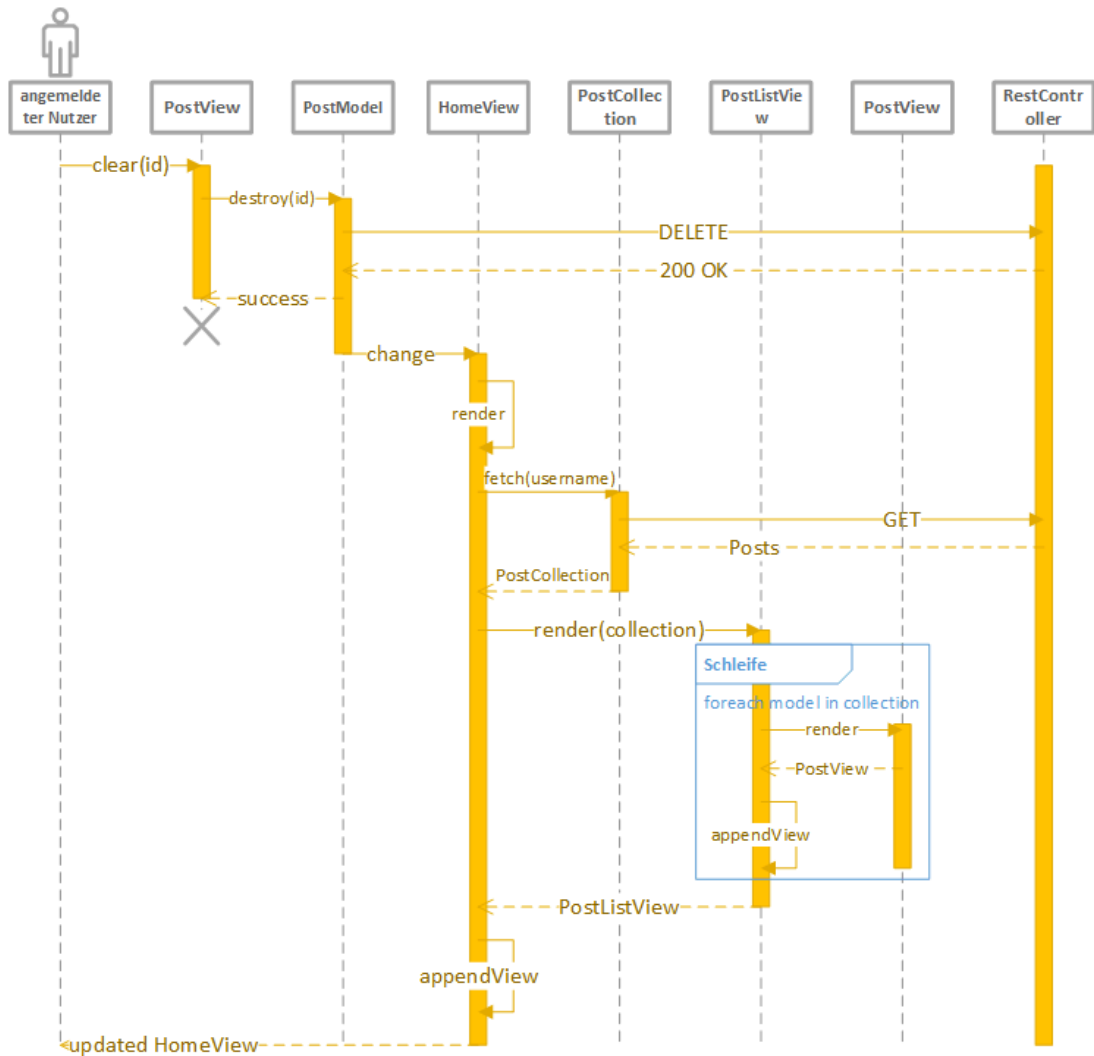


Abbildung 30: Sequenzdiagramm zum Löschen eines Beitrages

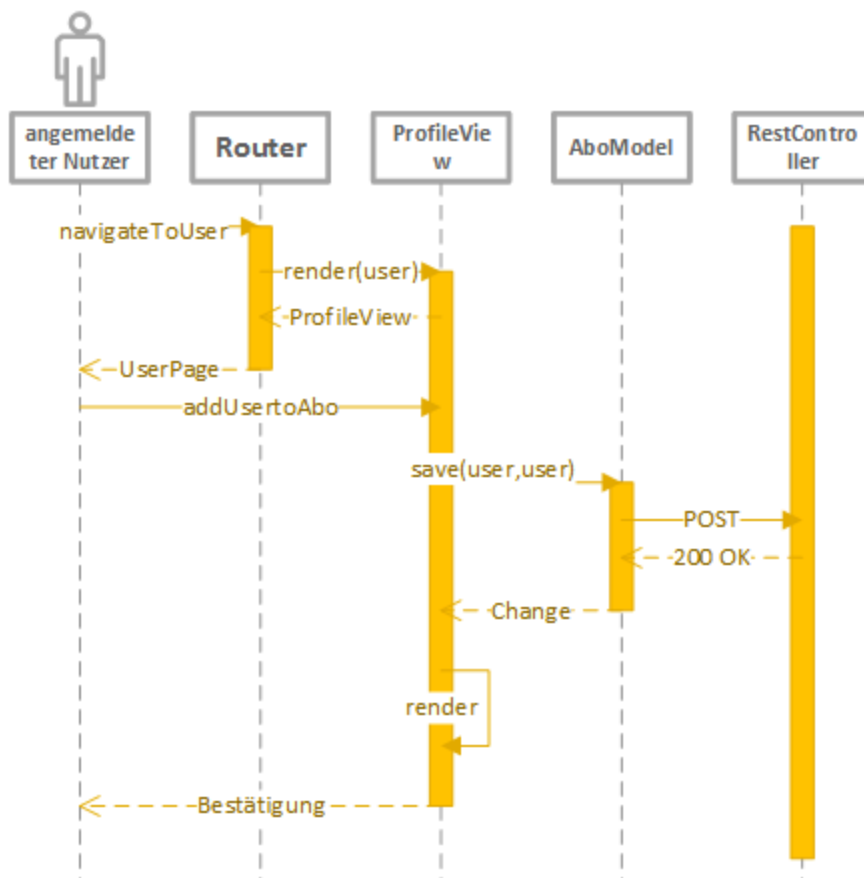


Abbildung 31: Sequenzdiagramm zum Abonnieren eines anderen Nutzers

Die in den Abbildungen 23 bis 31 gezeigten Sequenzdiagramme berücksichtigen keine Fehlerfälle, da diese aufgrund der Übersichtlichkeit nicht eingezeichnet wurden. Bei einem Fehler soll anstelle der Nutzdaten eine Fehlermeldung zurückgegeben werden.

5.3 Datenbankstruktur

Die Datenbank ist eine relationale Datenbank und umfasst 22 Tabellen. Diese lassen sich in sechs Kategorien unterteilen: Nutzerdaten, Autor-Account-Daten, Charakterdaten, Geschichtsdaten, Eventdaten, und Enumerationen.

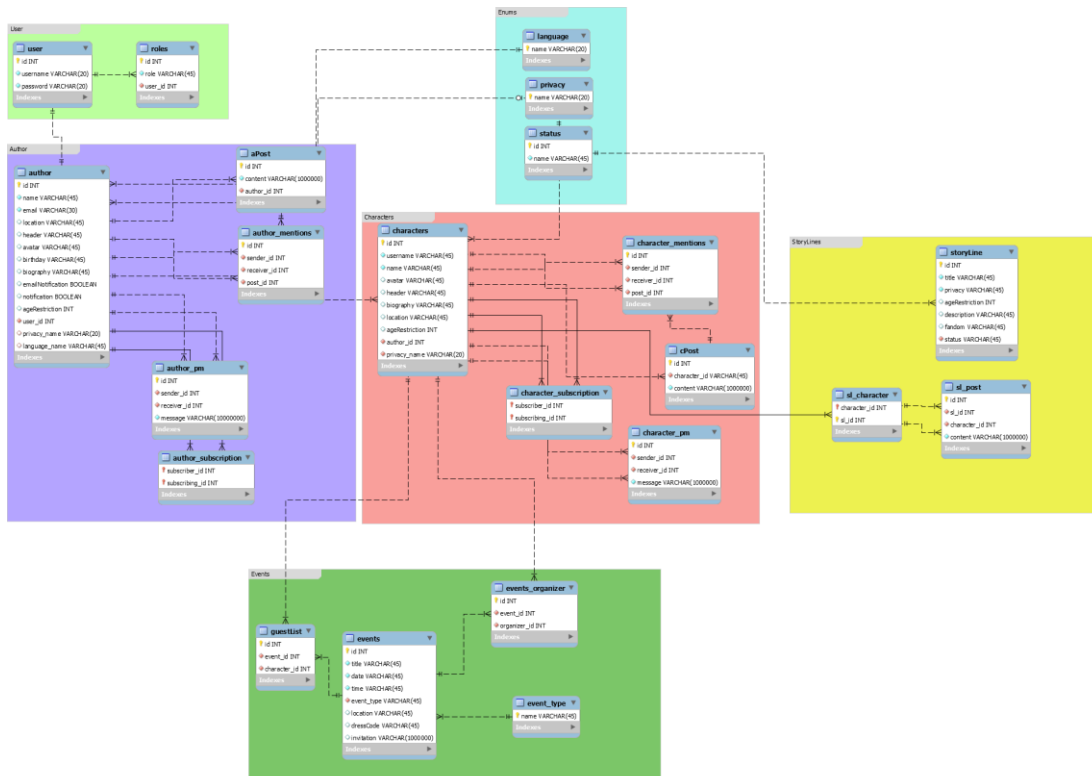


Abbildung 32: Übersicht des vollständigen ER-Modells der Datenbank

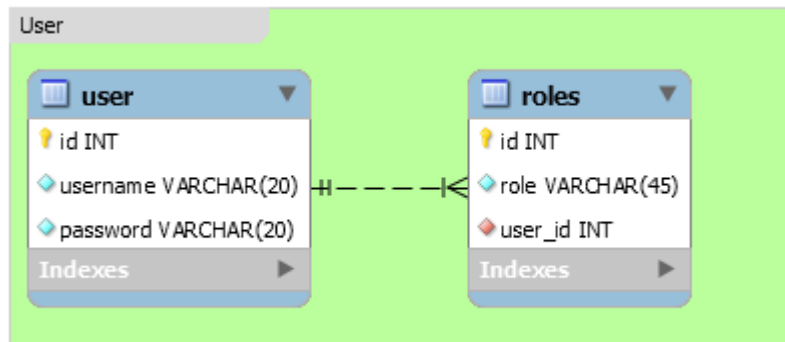


Abbildung 33: ER-Diagramm der User-Tabellen

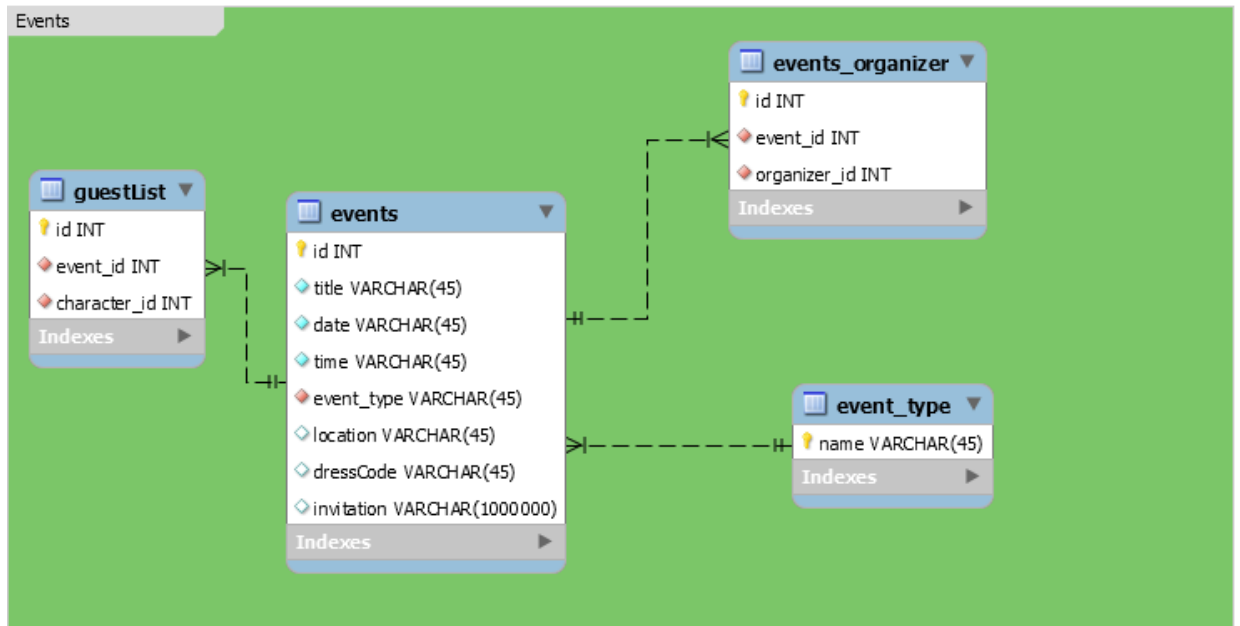


Abbildung 34: ER-Diagramm der Event-Tabellen

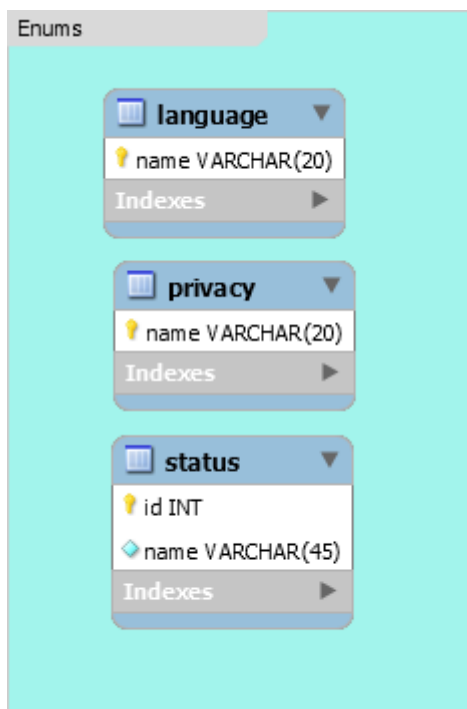


Abbildung 35: ER-Diagramm der Enumerations-Tabellen

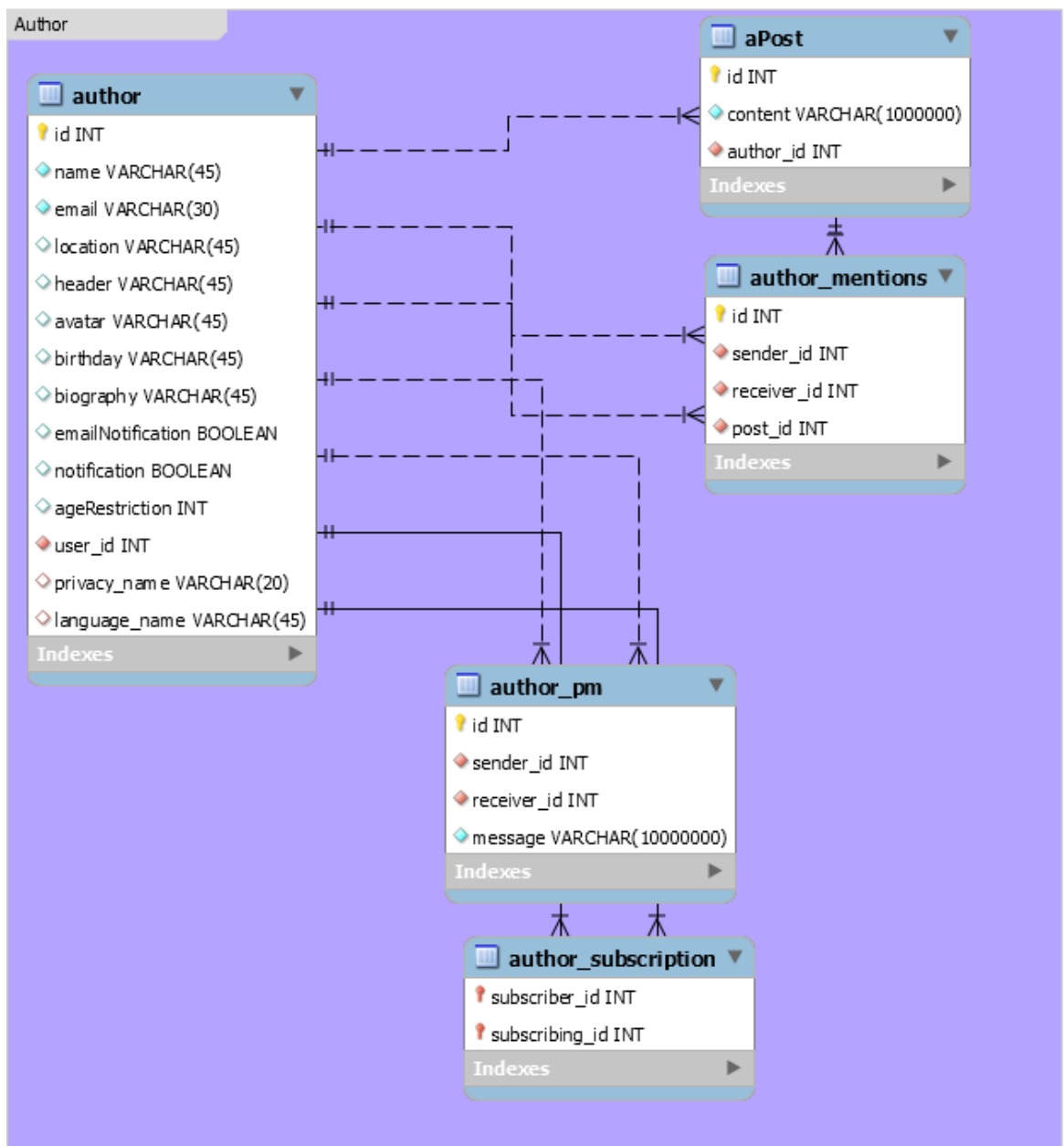


Abbildung 36: ER-Diagramm der Autor-Tabellen

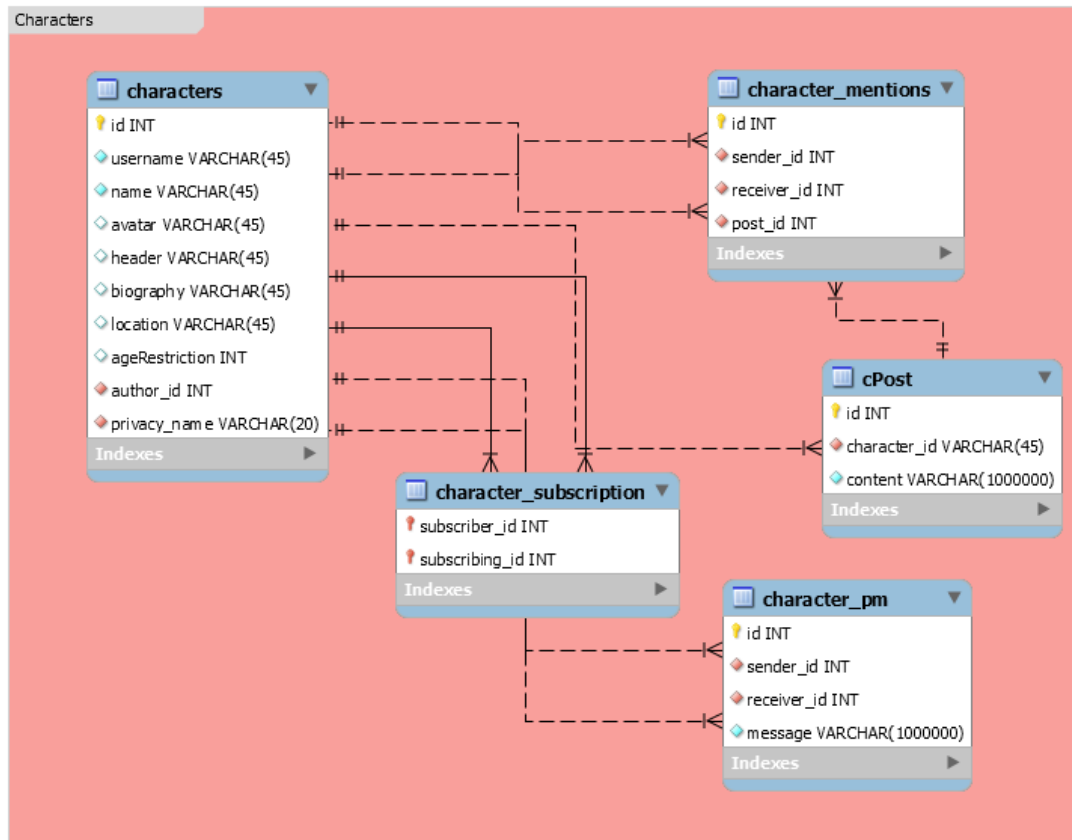


Abbildung 37: ER-Diagramm der Charakter-Tabellen

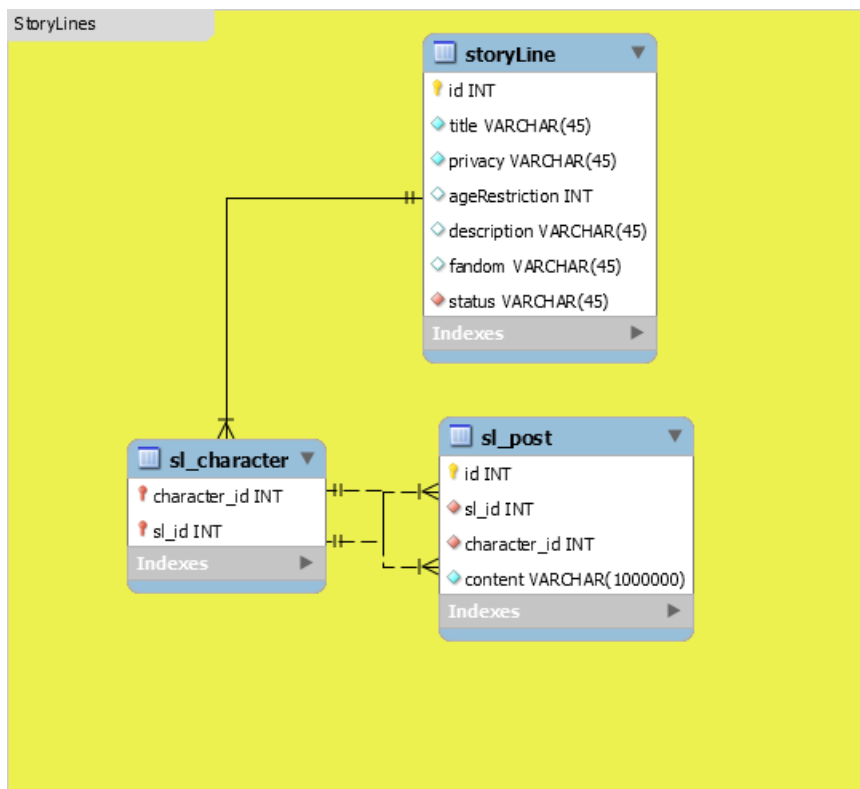


Abbildung 38: ER-Diagramm der Geschichts-Tabellen

Wie die Entity-Relationship-Modelle (ER-Model) zeigen, bestehen mehrere Schlüssel-Fremdschlüssel-Beziehungen zwischen den Tabellen innerhalb der Datenbank. Mit Hilfe dessen werden die jeweiligen Daten, auch Kategorie übergreifend, miteinander verknüpft.

Bei der Erstellung der Datenbank wurde darauf geachtet, dass die Tabellen der 3.Normalform entsprechen.

„Ein Relationentyp T genügt genau dann der 3. Normalform,

- *wenn sie der 1. Normalform genügt und*
- *wenn für jedes Attribut e und für jede volle funktionale Beziehung der Form $d \rightarrow e$ eine der folgenden Bedingungen erfüllt ist:*

- *Für die Determinante gilt $d = e$*
- *Das Attribut e ist Teil eines Schlüsselkandidaten.*
- *Die Determinante d ist ein Schlüsselkandidat.“*

(Piepmeyer 2011: 159)

Das heißt eine Tabelle genügt genau dann der 3.Normalform, wenn sie der 1.Normalform folgt und es weder transitive, noch partielle funktionale Abhängigkeiten zwischen den Nicht-Schlüsselattributen bzw. den Schlüsselkandidaten und Nicht-Schlüsselattributen gibt.

„Ein Relationentyp T genügt genau dann der ersten Normalform,

wenn

- *er keine Wiederholungsgruppen enthält und*
- *die Attributwerte in allen Relationen atomar sind.“*

(Piepmeyer 2011: 147)

Dies ist für alle Tabellen der vorliegenden Datenbank, bis auf die Tabelle **characters**, erfüllt.

Die Tabelle **characters** genügt nicht der 3.Normalform, da es eine transitive Abhängigkeit zwischen den Attributen **username** und allen anderen Attributen gibt. Dieser Sachverhalt liegt vor, weil in der Tabelle auf die Verwendung des natürlichen Schlüs-

sels **username** verzichtet wurde. Stattdessen wird der künstliche Schlüssel **id** verwendet, da die Tabelle mehrere Schlüssel-Fremdschlüssel-Beziehungen mit anderen Tabellen aufweist. Aufgrund der Beziehungen würde die Verwendung des natürlichen Schlüssels einen erheblichen Mehraufwand bei einer möglichen Änderung des Attributwertes – d.h. wenn der Nutzer den Usernamen seines Charakters ändert – bedeuten.

Die Tabelle **characters** genügt der 2.Normalform.

„Ein Relationentyp T genügt genau dann der 2. Normalform, wenn

- *T der 1. Normalform genügt und*
- *jedes Attribut a aus T eine der folgenden Bedingungen erfüllt:*
 - *a gehört zu einem Schlüsselkandidaten aus T .*
 - *a hängt voll von jedem Schlüsselkandidaten aus T ab“*

(Piepmeyer 2011: 154)

Das heißt die 2.Normalform liegt vor, wenn die 1.Normalform erfüllt ist und es keine partiellen funktionalen Abhängigkeiten in der Tabelle gibt, d.h. jedes Attribut ist entweder Teil des Schlüsselkandidaten oder hängt voll funktional von diesem ab.

Die anderen Tabellen weisen keinen natürlichen Schlüssel auf. Daher genügen sie auch trotz der Verwendung eines künstlichen Schlüssels der 3.Normalform.

6 Implementierung

Nach der Analyse und dem Entwurf der Web-Applikation wurde der Prototyp der Applikation WriRo (Abk. für Writing Role Play) entwickelt.

Der Prototyp der Web-Anwendung wurde als Client-Server-Programm, welches aus drei Schichten (Client, Server, Datenbank) besteht, entwickelt. Das Client-Programm wurde auf Basis von HTML5, CSS3, und JavaScript geschrieben. Insbesondere wurde die JavaScript-Bibliothek Backbone.js verwendet.

Das Server-Programm wurde als REST-Web-Service mit Hilfe von Java, insbesondere dem Spring Framework, entwickelt. Als Datenbankmanagementsystem wurde eine relationale Datenbank, die H2, ausgewählt.

In diesem Kapitel werden diese Entscheidungen hinsichtlich der Implementierung – Programmierumgebung und Datenbankmanagementsystem – erläutert.

Anschließend wird die Implementierung der Webanwendung beschrieben.

6.1 Wahl der Programmierumgebung

6.1.1 Wahl der Web-Technologien: Frontend

Das Client-Programm wurde auf Basis von JavaScript (JS), vor allem unter Verwendung der Bibliothek Backbone.js, entwickelt.

Backbone.js ist eine JS Bibliothek, die eine Model View Controller (MVC)-ähnliche Struktur im Frontend ermöglicht.

Die JS Model speichern relevante Daten der Single Page Application (SPA) und synchronisieren über CRUD (create, read, update, delete) Anweisungen mit dem Web-Service. Die Kommunikation findet über HTTP-Messages statt. Alle relevanten Daten werden im HTTP-Request-Body in Form von JSON (JavaScript Object Notation) übermittelt (vgl. Backbone.js 06-May-16).

Werden die Daten eines JS Models verändert, sendet dies eine **change**-Nachricht an alle JS Views, die das Model verwenden. Daraufhin werden die JS Views neu gerendert. Dadurch bleiben die Views stets aktuell, auch ohne dass der Anwender die Seite neu laden muss (vgl. Backbone.js 06-May-16).

Die JS Views rendern alle Daten aus den Models in HTML-Templates, wodurch diese auf der Seite angezeigt werden. Außerdem verarbeiten die Views alle Benutzer-Eingaben.

Einen klassischen Controller gibt es nicht. Dafür gibt es einen sogenannten Router, der abhängig von der URL bestimmte Funktionen der Model und Views aufruft (vgl. Backbone.js 06-May-16).

Zusätzlich zu den Models, Views und dem Router gibt es eine Collection. Die Collection ist eine Sammlung von Models, vergleichbar mit einem Array.

Durch den modularen Aufbau des Clients ist der Aufwand einer späteren Weiterentwicklung des Frontend möglichst gering. Es müssen lediglich bestimmte Models und Views geändert, oder hinzugefügt werden. Bei einer Erweiterung müssen bereits bestehende Model nicht bearbeitet werden. Außerdem können Models und Views jederzeit ausgetauscht, gelöscht, oder hinzugefügt werden, ohne dass andere Models oder Views geändert werden müssen.

Aufgrund der einfachen Struktur von Backbone.js, die eine schnelle Implementierung ermöglicht, der modularen Struktur, und der automatischen Aktualisierung von Inhalten wurde diese Bibliothek in diesem Projekt gewählt.

Weitere JavaScript-Bibliotheken und Templates, wie jQuery und Underscore.js wurden lediglich zur Vereinfachung von DOM-Operationen und dem Einbinden von JavaScript-Bibliotheken verwendet.

6.1.2 Wahl der Web-Technologien: Backend

Das Backend der vorliegenden Web-Applikation wurde mit Hilfe von Java, hauptsächlich dem Spring Framework, und einer relationalen Datenbank implementiert.

6.1.2.1 Wahl der Programmierumgebung für den Web-Service

Ein Großteil der Programmierumgebungen und -sprachen für Web-Anwendungen (Node.js, PHP, Ruby, Perl, Python und Java) eignet sich zur Entwicklung eines REST Web-Services mit Anbindung an eine Datenbank.

Eine der einfachsten Methoden ist PHP, insbesondere unter der Verwendung des PHP Yii Frameworks, das die Erstellung von Model- und Controller-Klassen automatisch übernimmt und die Wahl zwischen der Verwendung einer vom Framework erstellten oder eigenen View lässt (vgl. Getting Started: Creating First Yii Application | The Definitive Guide to Yii | Yii PHP Framework). Anstelle eines einfachen CRUD Codes können hier auch REST Web-Clients angebunden werden (vgl. How-To: Create a REST API | Wiki | Yii PHP Framework). Auf diese Weise lässt sich relativ schnell ein REST Web-Service realisieren.

Dennoch wurde für die Programmierung im Backend Java als Programmiersprache bzw. Programmierumgebung gewählt.

Java beschreibt eine einfache, robuste, objekt-orientierte, plattform-unabhängige, multi-threaded, und dynamische Programmierumgebung, die aus der Programmiersprache Java, den Java Klassen-Dokumenten, APIs, und der Java Virtual Machine (JVM) besteht (siehe Spell 2015: 1).

Als Nicht-Skript-Sprache bietet Java, im Gegensatz zu PHP, eine ausgereifte Objektorientierung und zahlreiche, ebenso ausgereifte und kontrollierte Bibliotheken und Frameworks, wie zum Beispiel JDBC (Java Database Connectivity), JSP (Java Server

Pages), EJB (Enterprise Java Beans), und JNDI (Java Naming and Directory Interface) (vgl. Walter 2008: 493).

Der dadurch entstehende Nachteil der hohen Komplexität Javas, wird durch die Verwendung des Spring Frameworks aufgehoben. Das Spring Framework, das von Pivotal Software Inc. entwickelt wurde, liefert zahlreiche Funktionen, die die Programmierung erleichtern und den Fokus auf die Entwicklung der eigentlichen Applikation ermöglichen. Dazu unterstützt das Spring Framework viele Anwendungen, wie REST, SOAP, und LDAP, und die Entwicklung auf verschiedenen Plattformen, wie Android und iOS, sowie verschiedene Sicherheitsmechanismen, wie Authentifizierung und Autorisierung, und dem Schutz vor CSRF (Cross Site Request Forgery), Session Fixation, Clickjacking, SQL Injection, und vielem mehr (vgl. Spring Projects).

Durch die Verwendung des Spring Frameworks konnten relativ schnell ein sicherer und performanter Rest-Controller und dem Datenbankschema entsprechende Repository-Klassen (siehe Kapitel 6.2.2), sowie eine Anbindung an die Datenbank über JDBC, realisiert werden, ohne auf die Vorteile einer Hochsprache verzichten zu müssen.

Der Programmcode des so entstandenen Web-Service ist einfach, strukturiert, objektorientiert und modular, und kann durch zahlreiche bereits bestehende Methoden (aus Bibliotheken und Frameworks) ergänzt werden.

6.1.2.2 Wahl der Datenbank

Die Wahl der Datenbank war eine wichtige Entscheidung im Zuge der Konzeption der Web-Applikation. Dabei wurden die beiden essentiellen Systeme, relationale Datenbanken und NoSQL-Datenbanken, auf ihre Stärken und Schwächen in Bezug auf dieses Projekt untersucht.

Relationale Datenbanksysteme zeichnen sich durch ihr relationales Datenmodell, einer Architektur, die durch das ANSI-SPARC-Modell Datenunabhängigkeit gewährleistet, einem festen Datenschema, welches Regeln für die Gewährung der Integrität enthält, einer standardisierten deskriptiven Abfragesprache, der Unterstützung des Mehrbenutzerbetriebs, und einer Konsistenzerhaltung (strong consistency) aus (vgl. Kaufmann und Meier 2016: 11, vgl. auch Piepmeyer 2011: 13). Zudem liefern sie Funktionen zur fehlerfreien und korrekten Speicherung von Daten (Transaktionen) und zum Schutz der Daten vor Zerstörung, Verlust und unbefugtem Zugriff (siehe Kaufmann und Meier

2016: 11). Komplexe Abfragen, die über Abfragen über den Primärschlüssel der Tabelle hinausgehen, können effizient mit Joins und Unterabfragen durchgeführt werden.

Diese Datenbanksysteme sind weit verbreitet, stoßen allerdings bei der Verwaltung von Big Data und bei massiv verteilten Anwendungen schnell an ihre Grenzen. Big Data beschreibt Daten, die folgende Eigenschaften aufweisen:

- Der Datenbestand ist im Terrabytebereich oder größer (Volume).
- Es werden unterschiedliche, multi-mediale Daten strukturiert, semi-strukturiert und bzw. oder unstrukturiert gespeichert (Variety).
- Die Datenverarbeitung erfolgt in Echtzeit (Velocity).

(siehe Kaufmann und Meier 2016: 13)

Relationale Datenbanken sind nicht oder nur schwer skalierbar und vergleichsweise langsam, das heißt sie können nur schwer an große Datenmengen (wie Big Data) angepasst werden und werden mit zunehmender Datenmenge langsamer (vgl. Parker et al.)

Für die Bedürfnisse von Big Data und massiv verteilten Anwendungen wurden NoSQL-Datenbanken entwickelt.

Der Begriff NoSQL bedeutet „Not Only SQL“ und ist ein Sammelbegriff für nicht-relationale Datenbanken (vgl. Lourenço et al. 2009: 741, vgl. auch Kaufmann und Meier 2016: 14). Daher umfasst er viele verschiedene Datenbanksysteme, die alle unterschiedliche Vor- und Nachteile aufweisen.

Eine Datenbank zählt dann zu den NoSQL-Datenbanken, wenn kein relationales Datenmodell vorliegt, die Datenarchitektur massive verteilte Webanwendungen und horizontale Skalierung unterstützt, umfangreiche Datenbestände, flexible Datenstrukturen und eine Echtzeitverarbeitung unterstützt werden, die Datenbankstruktur schemafrei ist, Datenreplikation und ein Mehrbenutzerbetrieb unterstützt werden, und eine bedingte Konsistenz (weak consistency oder eventually consistency) vorliegt (vgl. Kaufmann und Meier 2016: 419-420). Im Allgemeinen unterstützen NoSQL-Datenbanken alle Bedürfnisse von Big Data.

Die Verknüpfung von Daten – z.B. die Verknüpfung von mehreren Dokumenten in dokumentbasierten Datenbanken – ist nicht vorgesehen. Komplexe Abfragen können also nur schwer oder gar nicht ausgeführt werden.

Die Masse an NoSQL-Datenbanken und deren unterschiedlichen Werkzeuge, verlangt eine intensive Einarbeitung in diese. Das alleinige Verstehen der NoSQL-Datenbank-Strukturen und Verwendung reicht in der Regel nicht aus, um diese Systeme effizient zu nutzen (vgl. Meier 2016: 420). Die oben beschriebenen Funktionen der NoSQL-Datenbanken können nur bei einem effizienten Umgang mit diesen gewährleistet werden.

In diesem Projekt lag der Fokus auf die möglichst schnelle und effiziente Entwicklung eines Prototyps für eine Web-Anwendung. Eine intensive Einarbeitung in NoSQL-Datenbanken, die ein professionelles Umgehen mit diesen gewährleisten würde, war aufgrund der vorgegebenen Zeit nicht möglich.

Die Konsistenz der Daten ist ebenfalls in diesem Projekt besonders wichtig. Die zu entwickelnde Anwendung soll Benutzern die Möglichkeit des kollaborativen Schreibens ermöglichen. Dabei ist es wichtig, dass alle Benutzer bei einer Abfrage dieselben Daten erhalten. Wenn ein Nutzer zum Beispiel einen anderen Geschichtsbeitrag sieht als ein anderer Nutzer, kann er nicht entsprechend auf diesen Beitrag reagieren.

Zudem bestehen viele Beziehungen zwischen den Daten, die jederzeit abgefragt werden können. So werden Nutzerdaten benötigt, um passende Beitragsdaten abzurufen. Bei einer relationalen Datenbank können diese Beziehungen durch Schlüssel-Fremdschlüssel-Beziehungen ausgedrückt, und über Joins abgefragt werden. Bei NoSQL-Datenbanken – z.B. Dokumentbasierten Datenbanken – müssten die Nutzerdaten bei jedem Beitrag hinterlegt werden, damit sie später mit abgefragt werden können. Auf diese Weise würden redundante Daten – die Nutzerdaten würden bei jedem Beitrag kopiert und wiederholt in der Datenbank hinterlegt – entstehen, die eine Gefahr der Inkonsistenz bilden.

Da das Projekt keine Big Data oder massiv verteilte Anwendung ist, eine hohe Konsistenz gewährleistet sein muss, Beziehungen zwischen Daten dargestellt und abgefragt werden müssen, und eine intensive Einarbeitung in NoSQL-Systeme den Rahmen dieser Arbeit sprengen würde, wurde eine relationale Datenbank gewählt.

Die verwendete relationale Datenbank ist eine H2 Datenbank, die mit Standard SQL angesprochen wird.

6.2 Implementierung

Nach dem Entwurf der Software und der Entscheidung über die Implementierungswerkzeuge, folgt die Programmierung des Prototyps und die Einbettung des Prototyps in eine Server-Umgebung.

In diesem Kapitel wird beschrieben auf welche Weise die Entwürfe (siehe Kapitel 5) umgesetzt wurden. Dabei handelt es sich um eine kurze Zusammenfassung der entwickelten Klassen und deren Beziehungen untereinander.

6.2.1 Frontend-Entwicklung

Die Entwicklung im Frontend richtet sich nach dem im Kapitel 6.1.1 vorgestellten Prinzip von Backbone.js, den geforderten Views und dem Datenbankmodell (siehe Kapitel 5.3).

Für den Prototyp wurden insgesamt 76 JavaScript-Klassen implementiert. Diese teilen sich in Collections, Models, ListViews, Views, und sonstige auf.

Wie in Kapitel 6.1.1 beschrieben stellen Collections eine Sammlung von Models dar, während Models die Daten der Applikation speichern und diese mit dem Backend synchronisieren. Eine ListView ist die View zu einer Collection. Sie besteht, ähnlich wie die Collection selber, aus einer Sammlung von Views, während eine View die Daten eines Models repräsentiert. Dies soll am Beispiel der Suche verdeutlicht werden. Die JavaScript-Klasse **SearchCollection** ist eine Sammlung von **SearchModels**. Die Klasse **SearchModel** speichert alle relevanten Daten eines Suchergebnisses und synchronisiert sich mit dem Backend. Über das Model werden die Suchbegriffe an das Backend gesendet und die zurückerhaltenen Informationen in eine View eingebaut. Die **SearchListView** stellt die Auflistung aller Suchergebnisse dar. Sie besteht aus mehreren **SearchViews**, die wiederum ein einzelnes Suchergebnis darstellen.

Für den Prototyp wurden folgende, zusammenhängende Klassen programmiert:

- Das **AProfileModel** (Model des Autor-Profiles) mit der **ProfileOverview**.
- Das **ASettingsModel** (Model der Autor-Einstellungen) mit der **ASettingsView**
- Die **CharacterCollection** mit der **CharacterListView**, bestehend aus dem **CharacterModel** mit der **CharacterView**

- Die **CPostCollection** (für Charakter-Beiträge) mit der **CPostListView**, bestehend aus dem **CPostModel** mit der **CPostView**
- Das **CProfileModel** (für das Charakter-Profil) mit der **CProfileView**
- Die **CSLCollection** (für eine Liste aller Geschichten eines Charakters) mit der **CSLListView**, bestehend aus dem **CSLModel** mit der **CSLView**
- Die **ReplyCollection** mit der **ReplyListView**, bestehend aus dem **ReplyModel** mit der **ReplyView**
- Die **SearchCollection** (für die Darstellung von Suchergebnissen) mit der **SearchListView**, bestehend aus dem **SearchModel** mit der **SearchView**
- Das **SignUpUserModel** und das **SignUpAuthorModel** mit der **SignUpView** (die View präsentiert die Daten beider Models)
- Das **LoginModel** mit der **LoginView**
- Die **MentionCollection** mit der **MentionListView**, bestehend aus dem **MentionModel** mit der **MentionView**
- Die **PostCollection** mit der **PostListView** und der **PostListOverview**, bestehend aus dem **PostModel** mit der **PostView** und der **PostOverview** (Anmerkung: Je nach Berechtigungen des Nutzers, werden die Daten entweder in einer Übersichtsseite (Overview) oder einer Detailseite (View) präsentiert. Während die Detailseite Funktionen zur Bearbeitung der Daten (bearbeiten, löschen) liefert, liefert die Übersichtsseite lediglich die Daten selbst.)
- Die **SLPostCollection** mit der **SLPostListView** und der **SLPostListOverview**, bestehend aus dem **SLPostModel** mit der **SLPostView** und der **SLPostOverview** (siehe Anmerkung oben)
- Das **SLProfileModel** mit der **SLProfileOverview** und der **SLProfileView** (siehe Anmerkung oben)
- Die **SLSubscribeCollection** mit der **SLSubscribeListView**, bestehend aus dem **SLSubscribeModel** mit der **SLSubscribeView**
- Die **SubscribeCollection** mit der **SubscribeListView**, bestehend aus dem **SubscribeModel** mit der **SubscribeView**

- Die **SLCharacterCollection** (für eine Liste aller Charaktere einer Geschichte) mit der **SLCharacterListView**, bestehend aus dem **SLCharacterModel** mit der **SLCharacterView**

Zusätzlich zu diesen zusammenhängenden Klassen wurden vier Views zur Darstellung statischer Seiten programmiert. Da diese Views jeweils rein statische Inhalte darstellen, werden hier keine Models benötigt. Diese Views sind:

- Die **AboutView**
- Die **PrivacyPolicyView**
- Die **FirstStepView**
- Die **ImpressumView**

Eine Ausnahme zu dem vorgestellten Modell stellen die folgenden vier Views und zwei Model dar.

- Die **HomeView** (zur Darstellung des Home-Boards des Autors)
- Die **CHomeView** (zur Darstellung des Home-Boards des Charakters)
- Die **CRandomView** (zur Darstellung des Random-Boards)
- Die **NaviView** (zur Darstellung der Navigations-Leiste)
- Das **UserModel** (zum Speichern des eingeloggten Autors)
- Das **CUserModel** (zum Speichern des eingeloggten Charakters)

Diese Views dienen jeweils zur Kapselung anderer Views. Die aufgeführten Models dienen zur Speicherung des eingeloggten Nutzers und werden zur internen Autorisierung verwendet. Daher benötigen sie keine Views.

Die Klasse **Router** dient, wie im Kapitel 6.1.1 beschrieben, der Verwaltung von URLs.

In der HTML-Seite **index.html** sind die Templates für die einzelnen Views enthalten. Sie dient zur Darstellung der Views im Browser im Zuge der Single Page Application. Im CSS-Script **style.css** ist die grundlegende Gestaltung der Web-Applikation beschrieben.

Die **main.js** wird beim Aufruf der Web-Applikation geladen und ausgeführt. Sie initialisiert den Router der Web-Applikation und startet den Web-Client.

In der **run.js** werden externe Bibliotheken, wie jQuery, Underscore.js, und jsPDF, geladen, initialisiert und einer globalen Variable zugewiesen, sodass sie lediglich einmal geladen werden müssen.

In der JavaScript-Datei **PDFExport.js** ist die Funktion zum Exportieren der Geschichte in Form von PDF beschrieben.

Die Groovy-Datei **app.groovy** dient der Ausführung des Programms als Spring Boot Applikation auf dem Server (siehe Kapitel 6.2.3).

6.2.2 Backend-Entwicklung

In diesem Kapitel soll eine kurze Zusammenfassung der Implementierung des Web-Services im Backend mit Hilfe des Java Spring Frameworks gegeben werden.

Für den Web-Service wurden insgesamt 49 Java-Klassen programmiert. Diese lassen sich in 25 DTOs (Data Transfer Objects), 19 Repositories, 3 Konfigurationsdateien, einem RestController, und einer Hauptklasse (der Application.java) unterteilen.

Die DTOs richten sich nach dem Datenbankschema (siehe Kapitel 5.3). Analog wurde für jede Tabelle ein DTO erstellt. Daraus ergeben sich folgende Klassen:

- **Author**
- **AuthorMentions**
- **AuthorPM**
- **AuthorSubscription**
- **Character**
- **CharacterMentions**
- **CharacterPM**
- **CharacterSubscription**
- **Events**
- **EventsOrganizer**
- **EventType**
- **GuestList**
- **Languages**
- **Post**
- **Privacy**
- **Roles**
- **SLCharacter**
- **SLPost**

- **SLSubscription**
- **Status**
- **StoryLine**
- **User**

Die drei zusätzlichen DTOs **MainPost**, **AuthorPMModified** und **CharacterPMModified** dienen der Speicherung von Daten aus Join-Abfragen, die die Performance des Web-Services verbessern.

Passend zu den DTOs wurden folgende Repository-Klassen entwickelt:

- **AuthorRepository**
- **AuthorMentionsRepository**
- **AuthorPMRepository**
- **AuthorSubscriptionRepository**
- **CharactersRepository**
- **CharacterMentionsRepository**
- **CharacterPMRepository**
- **CharacterSubscriptionRepository**
- **EventsRepository**
- **EventsOrganizerRepository**
- **GuestListRepository**
- **MainPostRepository**
- **PostsRepository**
- **RolesRepository**
- **SLCharacterRepository**
- **SLPostRepository**
- **SLSubscriptionRepository**
- **StoryLineRepository**
- **UserRepository**

Der **WriRoRestController** ist der REST-Controller der Applikation. Er empfängt die HTTP-Messages des Web-Clients, bearbeitet diese und sendet entsprechende Antworten an den Client zurück.

Die Klasse **MvcConfig** speichert Konfigurationen der Spring WebMVC-Anwendung. Unter anderem werden hier die Daten für den Datenbankzugriff (Datenbank-URL, Nutzernamen und Passwort, und Datenbanktreiber) gespeichert.

Die **WebSecurityConfig** verwaltet Sicherheits-Konfigurationen des Web-Services mit Hilfe von Spring Security. Unter anderem werden hier Daten bezüglich der Autorisierung und Authentifizierung zum Zugriff auf den Web-Service, und Daten bezüglich des Schutzes vor CSRF gespeichert.

Der **WebAppInitializer** initialisiert die Konfigurations-Klassen.

Die Klasse **Application** ist, wie bereits oben beschrieben, die Hauptklasse des Web-Services. In ihr wird der **SpringApplicationBuilder** initialisiert, der die Applikation startet und ihr einen automatisch konfigurierten Kontext zuweist. Außerdem wurde hier die **main**-Methode des Java-Programms implementiert.

Die XML-Datei **pom.xml** verwaltet die externen Abhängigkeiten der Applikation, z.B. alle Spring Bibliotheken des Spring Frameworks.

Weitere Konfigurationen des Web-Services, wie die Adresse und Portnummer des Services, wurden in der Properties-Datei **application.properties** gespeichert.

6.2.3 *Server-Einbettung*

Die Applikation läuft mit Hilfe von Apache Tomcat 8 und Spring Boot auf einem virtuellen Windows Server 2012 R2.

Der als WAR (Web Archive) verpackte Web-Service wird mit Hilfe von Tomcat auf dem Webserver auf Port 8080 ausgeführt.

Der Web-Client wird hingegen mit Hilfe von Spring Boot, das intern einen Tomcat 8 verwendet, auf Port 8081 ausgeführt. Spring Boot führt eine groovy-Datei aus, die als Start-Skript des Web-Clients dient und diesen lauffähig macht.

6.3 **Ergebnis**

Ergebnis der Implementierung ist ein erster Prototyp der Web-Applikation WriRo, der unter der Web-Adresse <http://wriro.de> zu erreichen ist, und folgende Funktionen unterstützt:

- Autor-Account erstellen und verwalten
- Charakter erstellen und bearbeiten

- Die Möglichkeit jederzeit zwischen Charakteren zu wechseln
- Nutzer (Charakter und Autor) abonnieren bzw. nicht mehr abonnieren
- Geschichtsprofil erstellen und bearbeiten
- Geschichten verfassen (Geschichtsbeiträge erstellen, bearbeiten und löschen)
- Andere Beiträge (Random RP und Statusmeldungen) erstellen, bearbeiten und löschen
- Freitextsuche nach Charakteren und Autoren
- Auflistung aller registrierten Charaktere und Autoren
- Erstellung verschiedener Boards (Main-Board, Leseliste, Profilseite, Random-Board, Story-Board)
- Abonnieren einer Geschichte
- Download einer Geschichte als PDF

Die Web-Applikation WriRo hat einen Login-Bereich (die Startseite), über den sich der Nutzer entweder in einen bestehenden Autor-Account einloggen oder einen neuen Autor-Account registrieren kann.

Welcome to WriRo

The platform for writing roleplay

Please login with your username and password or sign up a new author account.

If you are new to this page, please read the about page before you proceed.

Username

Password

Abbildung 39: Ausschnitt der Startseite von WriRo

Die Hauptseite des Autors zeigt das Home-Board. Dort werden alle eigenen Beiträge und alle Beiträge der abonnierten Autoren angezeigt. Auf der linken Seite befindet sich eine Navigationsleiste. Über diese kann der Autor neue Charaktere anlegen oder einen bereits angelegten Charakter auswählen. Außerdem kann er über die Navigation auf sein Autorprofil zugreifen und alle abonnierten Autoren und Abonnenten in einer jeweiligen Liste einsehen.

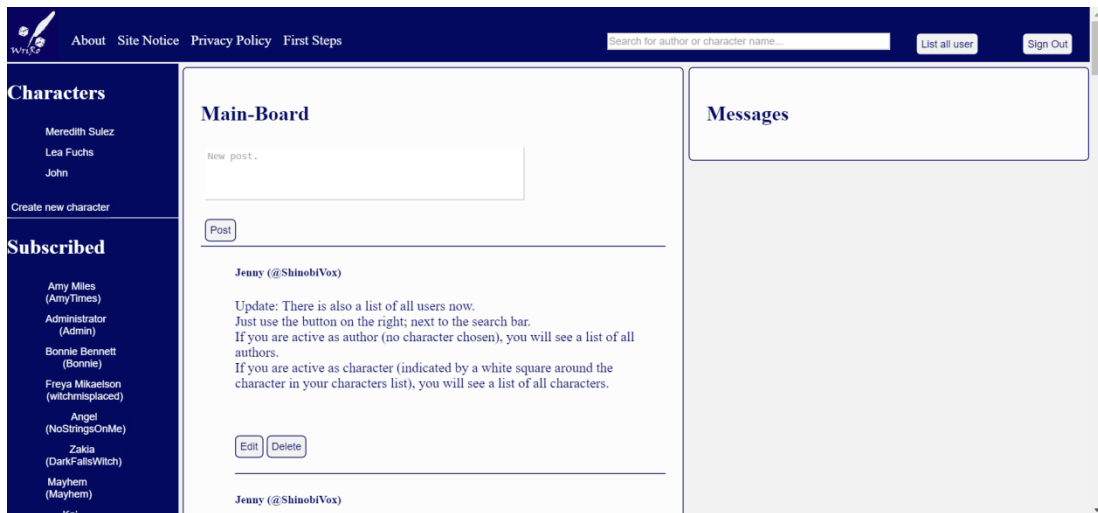


Abbildung 40: Hauptseite des Autors auf WriRo

Im Autor-Profil werden alle eigenen Beiträge des Autors angezeigt, sowie eine Beschreibung des Autors (Name, Biografie, etc.).



Abbildung 41: Eigenes Autor-Profil auf WriRo

Nach Auswahl eines Charakters gelangt der Nutzer auf die Hauptseite des jeweiligen Charakters. Dort werden in dem Home-Board alle Random-Beiträge des Charakters und der abonnierten Charaktere, sowie die Reading-List, die eine Auflistung aller abonnierten Geschichten enthält, dargestellt. Auf der Hauptseite des Charakters hat der Nutzer die Möglichkeit über ein Formular eine neue Geschichte anzulegen.

Die Navigation unterscheidet sich äußerlich lediglich in der Markierung des ausgewählten Charakters. Inhaltlich verweist der Menü-Punkt **Profil** hier auf das Charakterprofil

und die Auflistung der Abonnenten und abonnierten Accounts (hier: Charaktere) bezieht sich hier auf den ausgewählten Charakter.

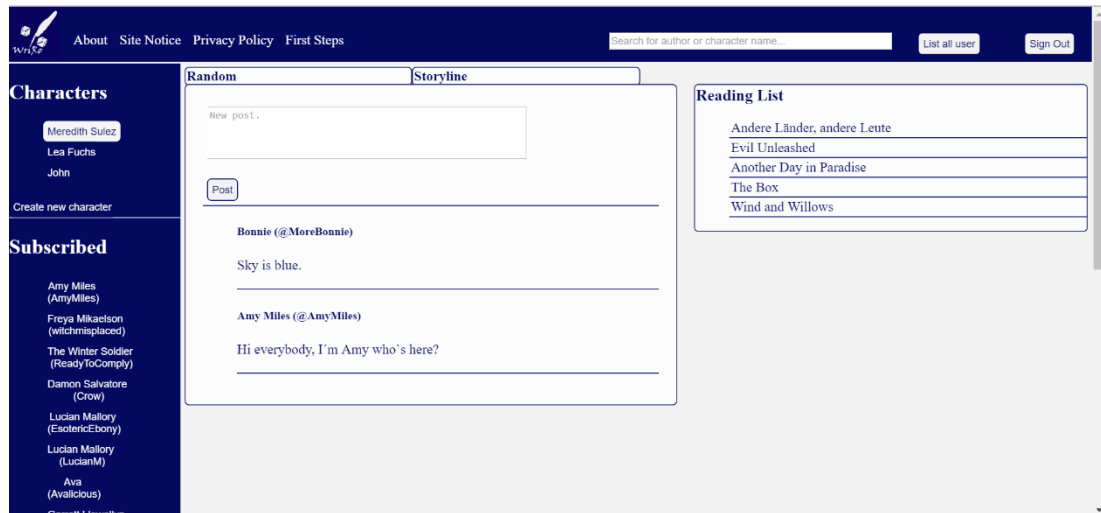


Abbildung 42: Hauptseite eines Charakters in WriRo mit Random-Beiträgen

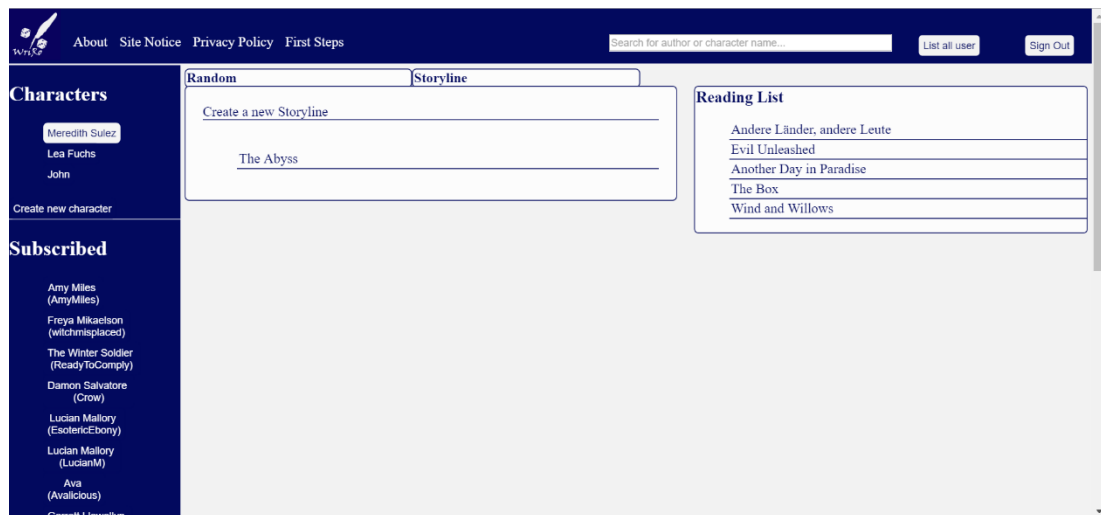


Abbildung 43: Hauptseite eines Charakters in WriRo mit der Geschichtsübersicht

Im Charakterprofil werden ähnlich wie im Autorprofil Informationen über den Charakter, sowie dessen Random-Beiträge, angezeigt. Außerdem werden hier die Geschichten des jeweiligen Charakters aufgelistet.



Abbildung 44: Eigenes Charakter-Profil

Nach dem Anlegen oder Auswählen einer Geschichte gelangt der Nutzer in das Geschichtsprofil. Als Verfasser der Geschichte hat er hier die Möglichkeit die Geschichte zu bearbeiten oder einen Beitrag zu dieser Geschichte zu verfassen.

Besucht der Nutzer die Geschichtsseite einer Geschichte, an der er nicht beteiligt ist, hat er stattdessen die Möglichkeit die Geschichte zu abonnieren.

In beiden Fällen werden auf der Geschichtsseite die Informationen über die Geschichte (Zusammenfassung, Charaktere, Thema, Altersbeschränkung) und alle Beiträge der Geschichte angezeigt.

Außerdem besteht hier die Möglichkeit die Geschichte als PDF herunterzuladen.



Abbildung 45: Geschichtsseite einer eigenen Geschichte auf WriRo



Abbildung 46: Geschichtsseite einer abonnierten Geschichte auf WriRo

Zum Verfassen eines Beitrages befindet sich auf jeder Seite ein Textfeld, mit Hilfe dessen ein neuer Beitrag verfasst werden kann. Auf der Geschichtsseite muss zum Anzeigen des Textfeldes zunächst der Button **Add Post** betätigt werden (siehe Abbildung 45).

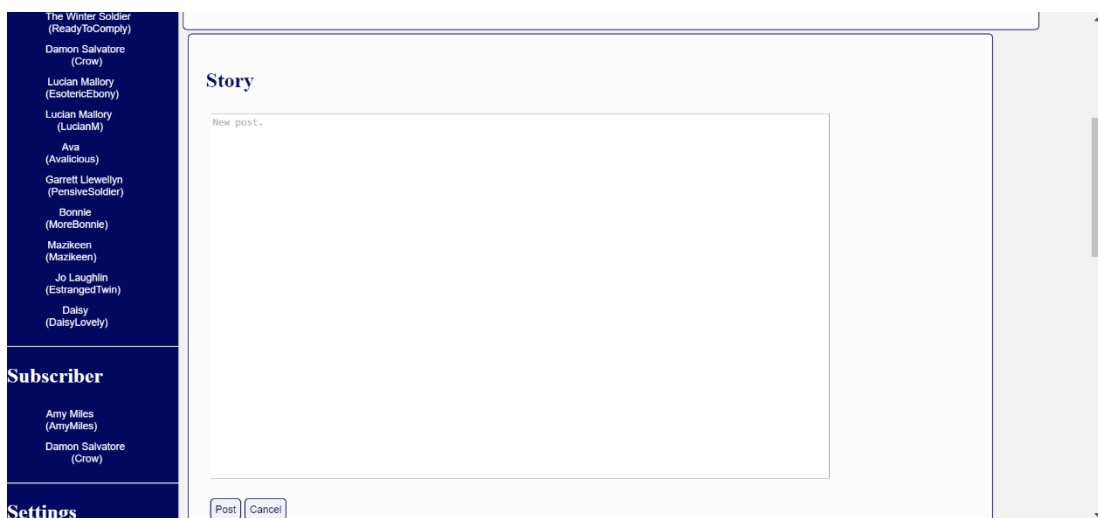


Abbildung 47: Textfeld zum Hinzufügen eines Geschichts-Beitrags auf WriRo

Die eigenen Beiträge können immer editiert oder gelöscht werden, während die Beiträge anderer nur gelesen werden können (siehe Abbildung 48 und Abbildung 49).



Abbildung 48: Random-Beiträge auf WriRo

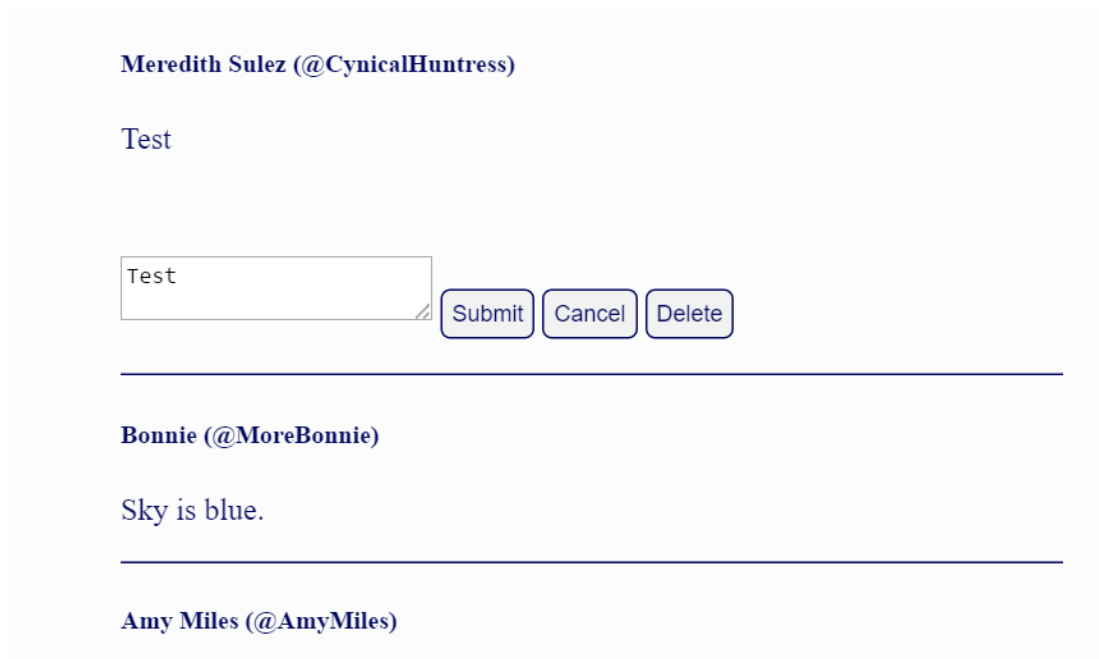


Abbildung 49: Editieren eines Beitrags auf WriRo

Im Header-Bereich der Applikation befindet sich ein Suchfeld. Über dieses können Charaktere oder Autoren über Freitext gesucht werden.

Ist der Nutzer als Autor eingeloggt und kein Charakter ist ausgewählt, kann er mit Hilfe der Freitext-Suche nach einem Autornamen oder Autor-Nutzernamen suchen (siehe Abbildung 50).

Verwendet er stattdessen den **List all user** Button, werden alle registrierten Autoren aufgelistet (siehe Abbildung 51).

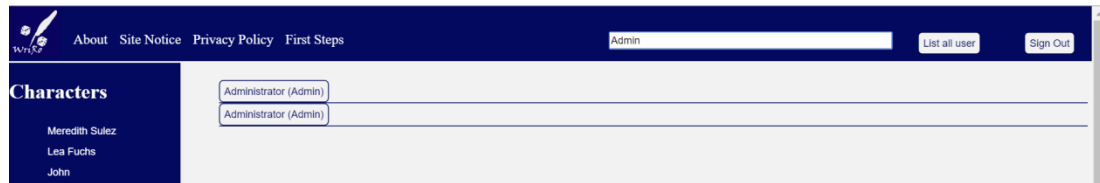


Abbildung 50: Suche nach einem Autor auf WriRo

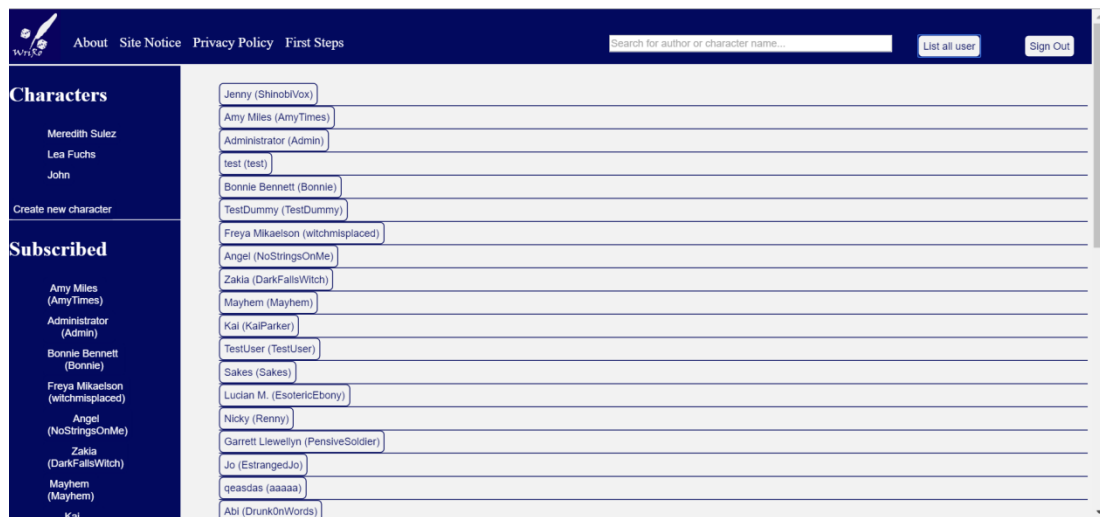


Abbildung 51: Lister aller Autoren auf WriRo

Ist ein Charakter ausgewählt, kann der Nutzer mit Hilfe der Freitext-Suche nach einem Charakternamen oder Charakter-Nutzernamen suchen (siehe Abbildung 52).

Verwendet er stattdessen den **List all user** Button, werden alle registrierten Charaktere aufgelistet (siehe Abbildung 53).

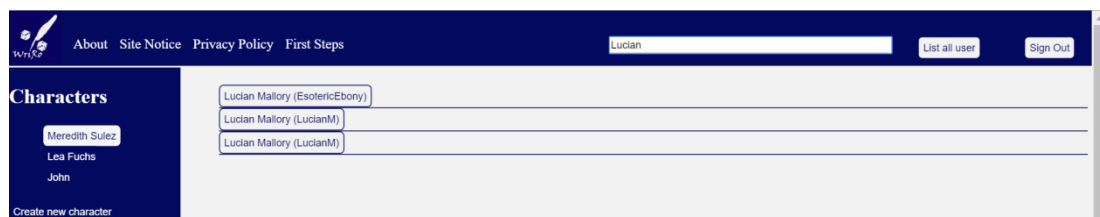


Abbildung 52: Suche eines Charakters auf WriRo

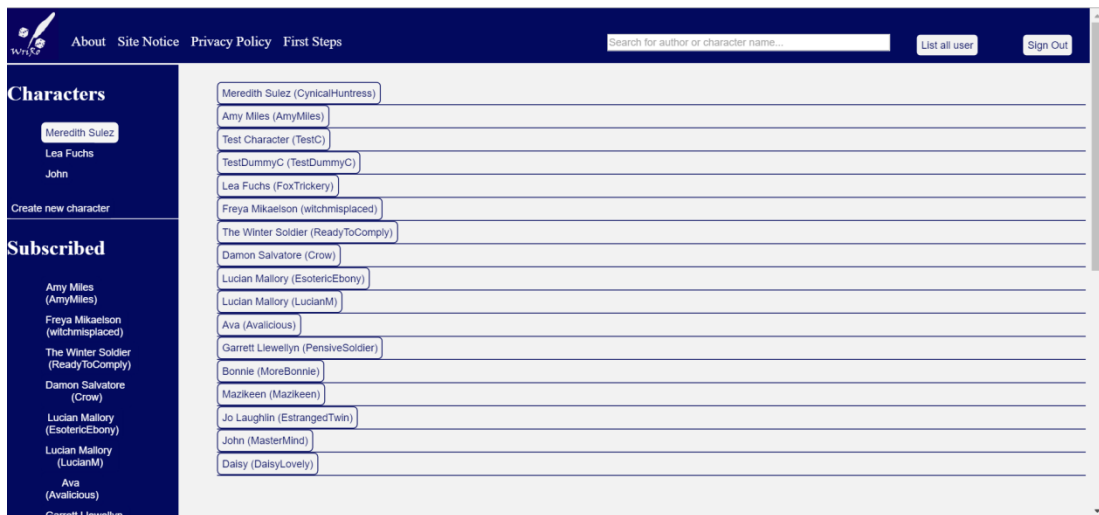


Abbildung 53: Liste aller Charaktere auf WriRo

Nach der Auswahl eines Suchergebnisses gelangt der Nutzer in das Profil des ausgewählten Autors oder Charakters. Dort hat er die Möglichkeit den entsprechenden Autor oder Charakter zu abonnieren.

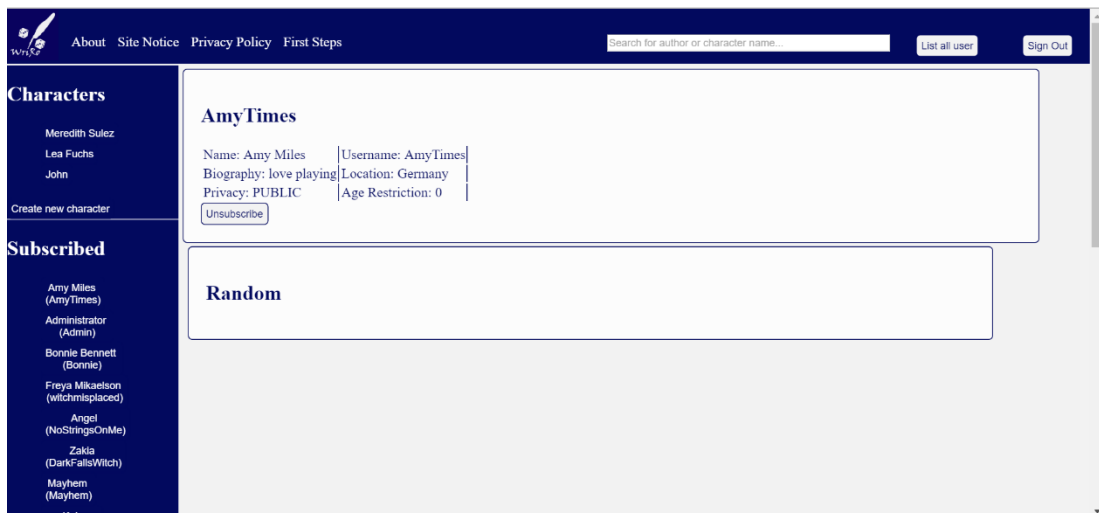


Abbildung 54: Profil eines Autors auf WriRo



Abbildung 55: Profil eines Charakters auf WriRo

Zusätzlich wurden eine Site Notice (Impressum)-, About-, Privacy Notice (Datenschutzrichtlinien)- und First Steps-Seite angelegt.

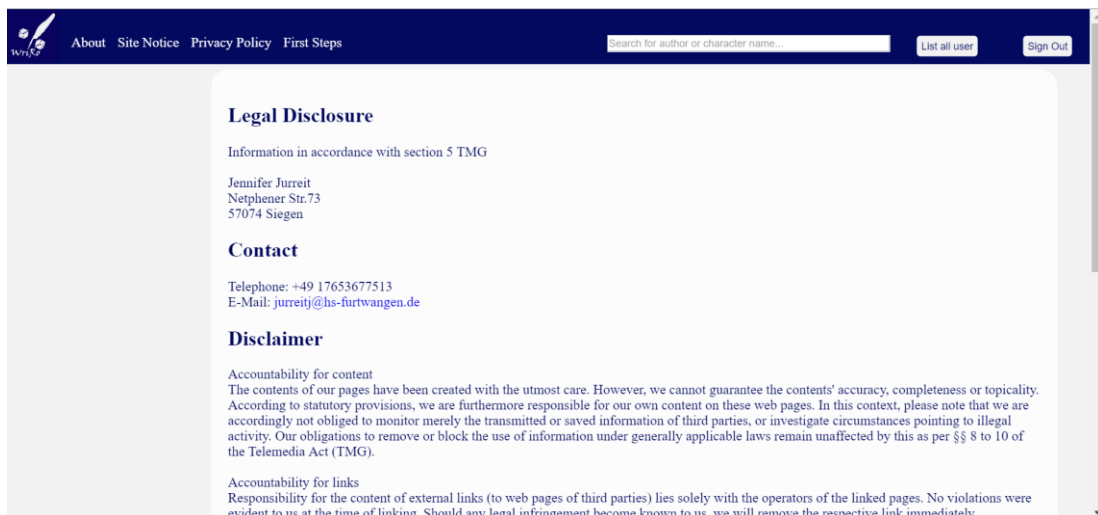


Abbildung 56: Impressum von WriRo

Auf das Design der Web-Applikation wurde aufgrund des Fokus auf die Programmierung keine besondere Rücksicht genommen. Lediglich die Farbauswahl und das WriRo-Logo wurden im Vorfeld zur Implementierung ausgearbeitet.

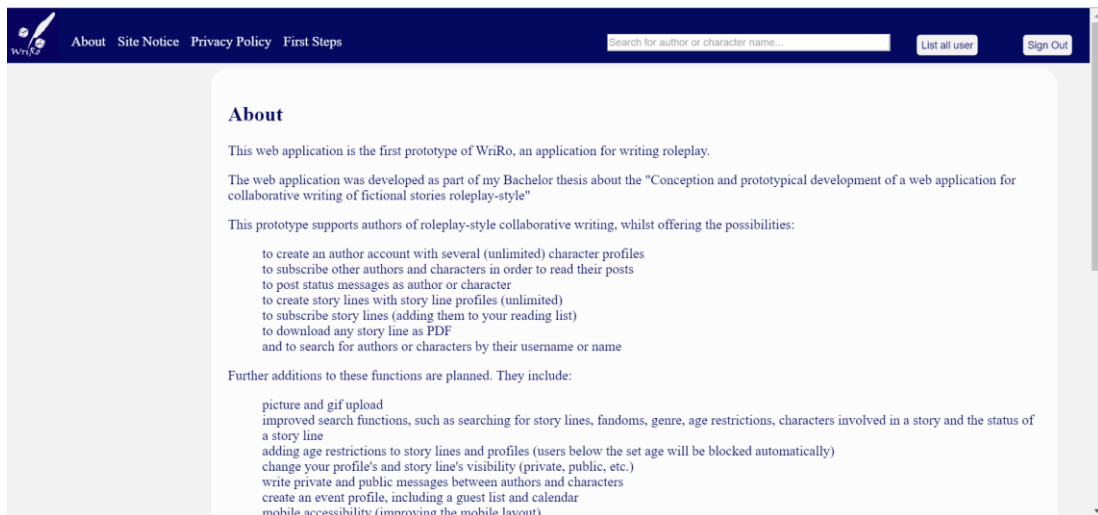


Abbildung 57: About-Seite auf WriRo

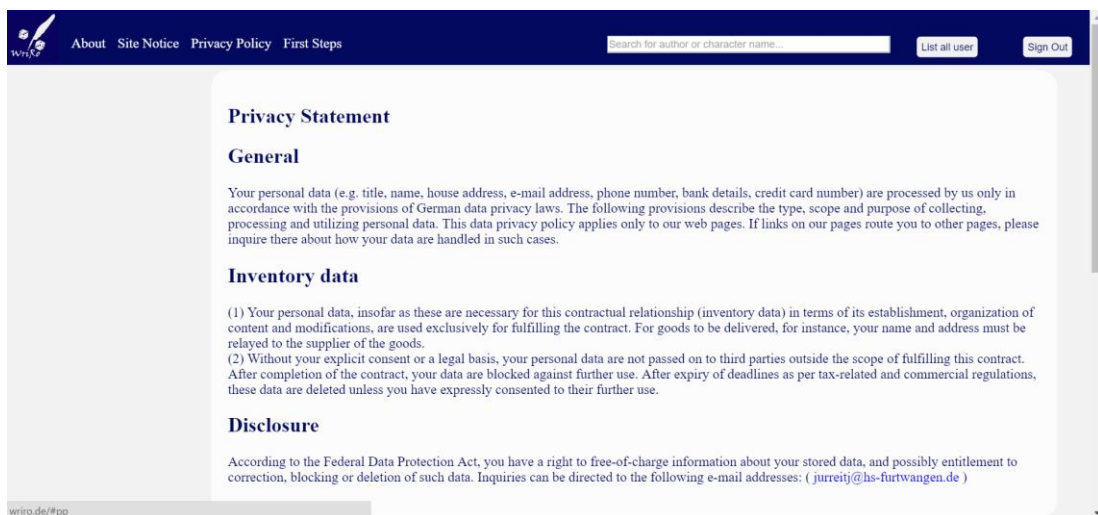


Abbildung 58: Datenschutzrichtlinien auf WriRo

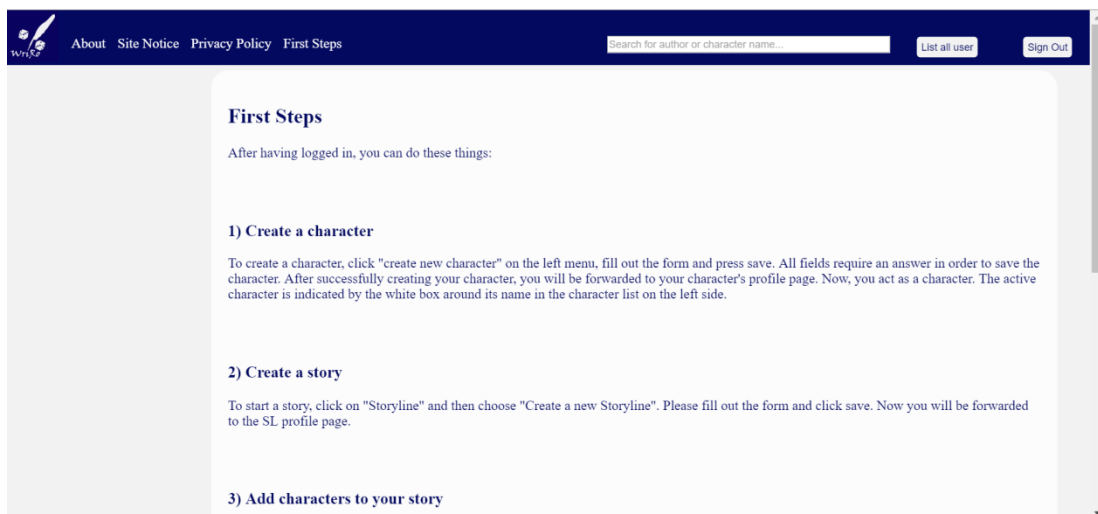


Abbildung 59: First Steps Seite auf WriRo

7 Evaluation

Der Prototyp der Web-Anwendung, WriRo, wurde über einen Zeitraum von zwei Monaten von vierzehn Testpersonen, die bereits Erfahrungen mit anderen Schreibrollenspiel-Anwendungen und dem Ablauf eines Schreibrollenspiels hatten, getestet und über einen Fragebogen ausgewertet.

Die Testpersonen wurden gebeten einen Autor-Account und mindestens einen Charakter zu erstellen, um die Funktionen der Anwendung zu analysieren. Außerdem wurden die Probanden gebeten mindestens eine SL, und möglichst viele Beiträge, zu schreiben. Dies galt dem Testen der Schreiberfahrung und dem Sammeln von Daten über die literarischen Ergebnisse des Schreibrollenspiels.

Anschließend wurden die Probanden gebeten einen Fragebogen auszufüllen, der sie sowohl über ihre Erfahrungen mit dem Prototyp, als auch ihre Erwartungen an die Anwendung, ihre Bewertung des Konzeptes hinter der Anwendung, und mögliche Verbesserungen der Anwendung befragte.

In diesem Kapitel werden die Ergebnisse dieser Umfrage erläutert und die auf diesem Ergebnis basierenden Verbesserungen des ersten Prototyps vorgestellt.

7.1 Ergebnis der Evaluation

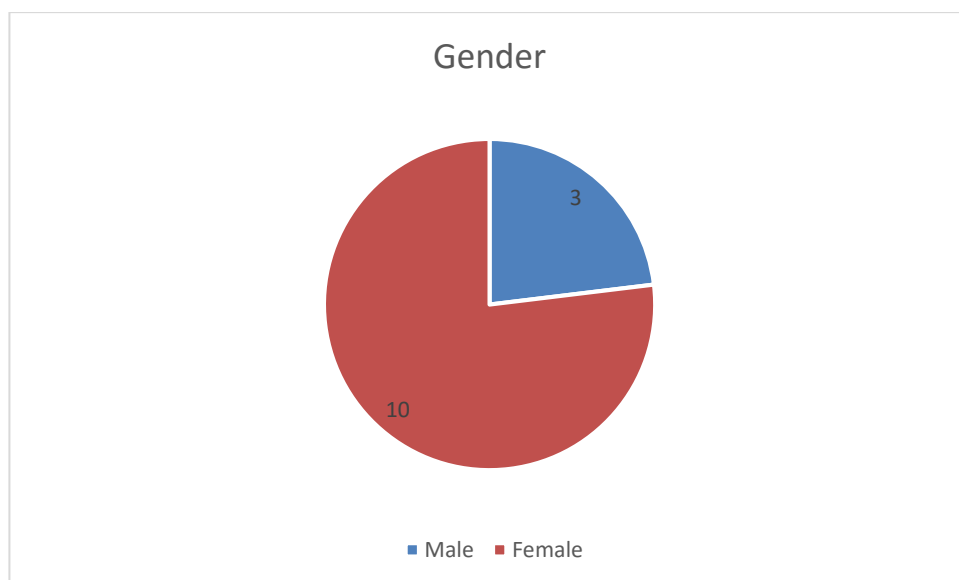


Figure 1: Geschlecht der Testpersonen

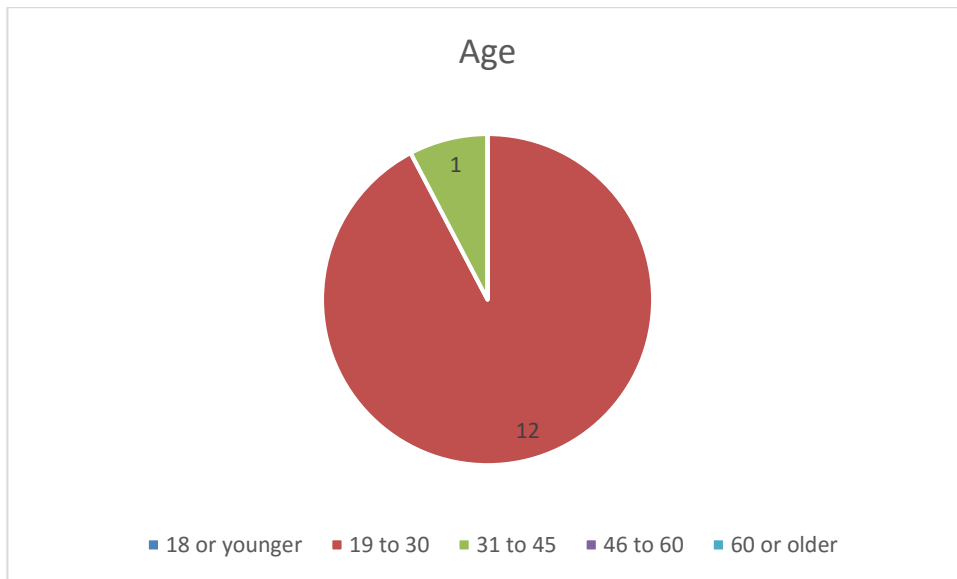


Figure 2: Alter der Testpersonen

Wie aus Figur 1 und 2 hervorgeht waren 10 der 14 Befragten weiblich, während 3 der Befragten männlich waren und 1 Befragter kein Geschlecht angegeben hat.

12 der Befragten waren zwischen 19 und 30 Jahre alt, 1 Testperson war älter (zwischen 31 und 45 Jahren) und 1 Person gab kein Alter an.

Von 14 Befragten gaben 7 an, dass sie die Web-Applikation sehr interessant finden, und 4 Befragte fanden die Applikation interessant. Die restlichen 3 Befragten sind neutral gegenüber der Applikation.

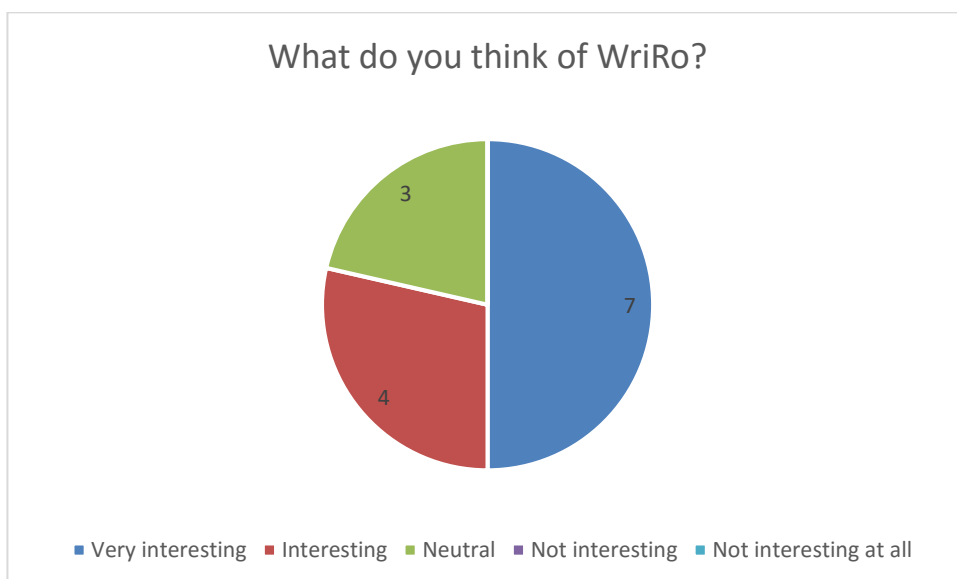


Figure 3: Was halten Sie von WriRo?

Das Konzept der Web-Applikation überzeugt 10 der Befragten, während 4 Personen nicht so zufrieden oder nur einigermaßen zufrieden sind. Unter den 10 Befragten, die das Konzept überzeugt hat, waren 9 sehr zufrieden und 1 extrem zufrieden mit dem Konzept.

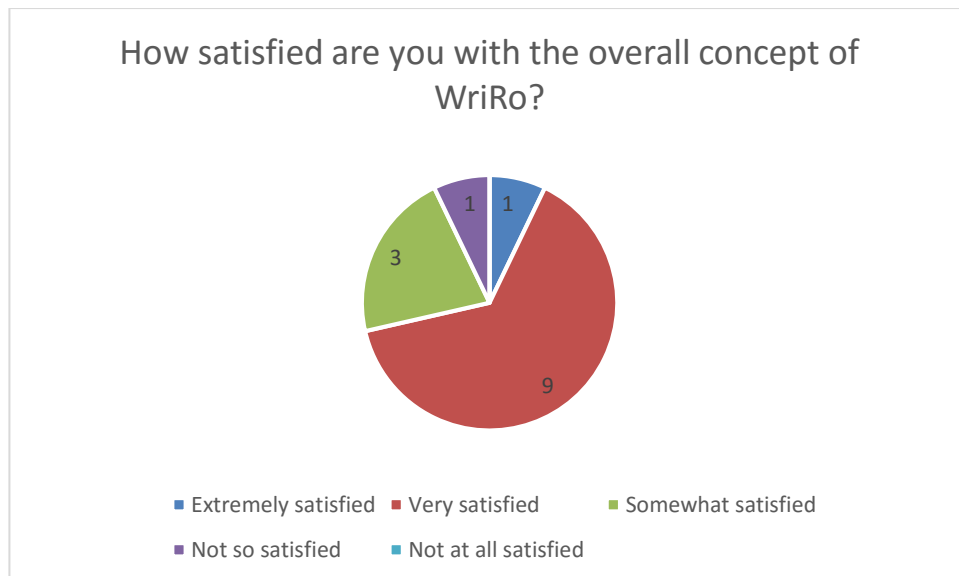


Figure 4: Wie zufrieden sind Sie mit dem Konzept von WriRo?

9 von 14 Befragten sind mit dem Schreiberlebnis sehr oder extrem zufrieden, während 4 der Befragten das Schreiberlebnis als lediglich befriedigend erachten und 1 überhaupt nicht zufrieden war.

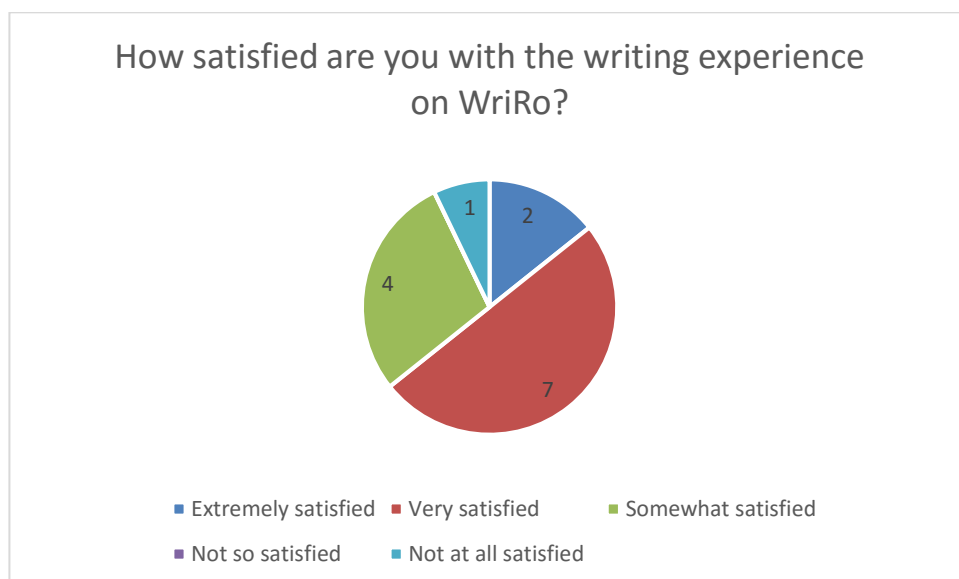


Figure 5: Wie zufrieden sind Sie mit dem Schreiberlebnis auf WriRo?

Die Möglichkeit zur Kollaboration mit anderen Nutzern wurde von den meisten Befragten als befriedigend eingestuft. 8 von 14 Befragten waren extrem oder sehr zufrieden, während 5 nur einigermaßen und 1 überhaupt nicht zufrieden waren.

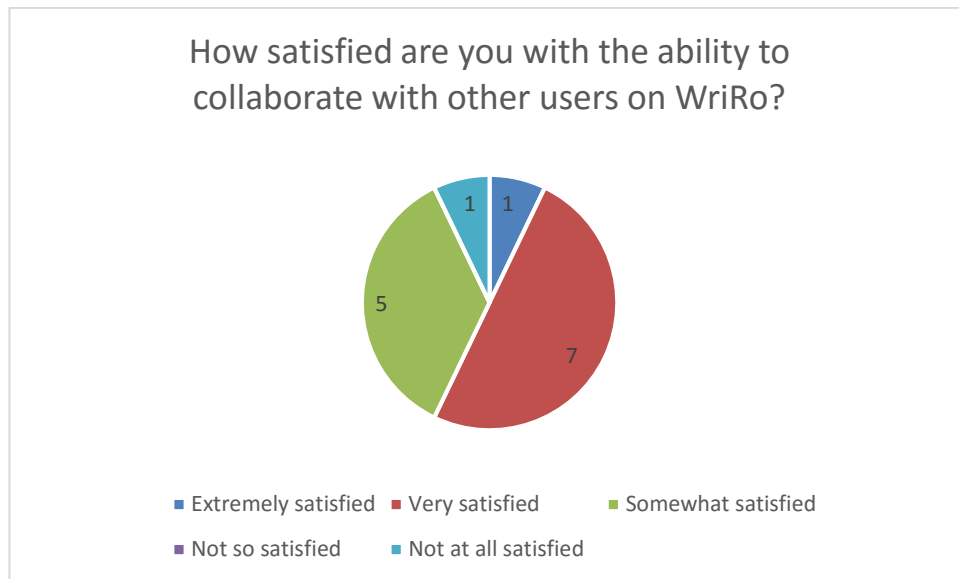


Figure 6: Wie zufrieden sind Sie mit der Möglichkeit zur Kollaboration auf WriRo?

3 Testpersonen gaben an, dass die Applikation kompliziert sei, und 2 gaben an, dass sie die Applikation nicht brauchen würden, als sie gefragt wurden, warum sie die Applikation nicht mögen würden.

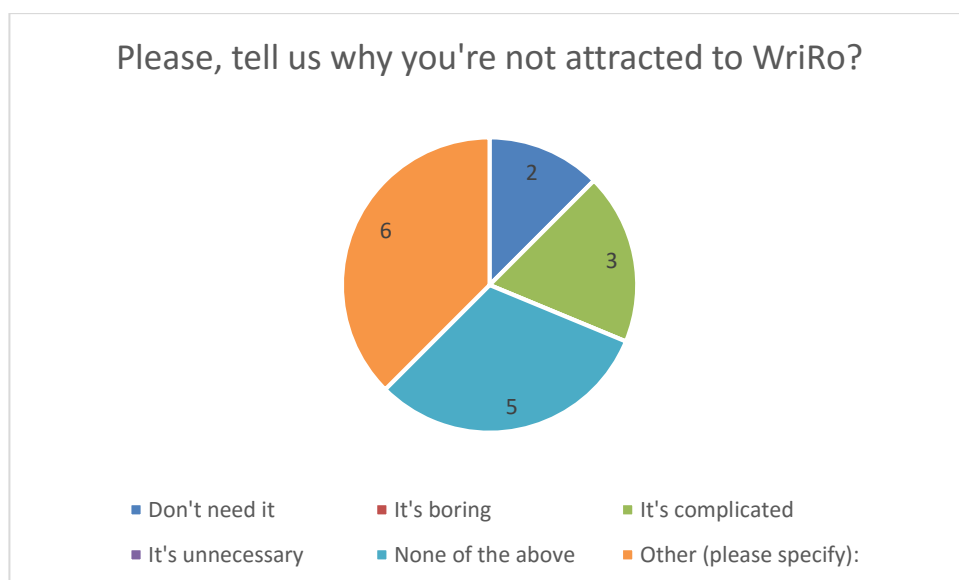


Figure 7: Bitte teilen Sie uns mit, warum Sie WriRo nicht mögen.

Dahingegen gaben 9 Personen an, dass sie die fertige Applikation nutzen würden, während 5 Personen angaben, dass sie die Applikation wahrscheinlich nicht oder nur vielleicht nutzen werden.

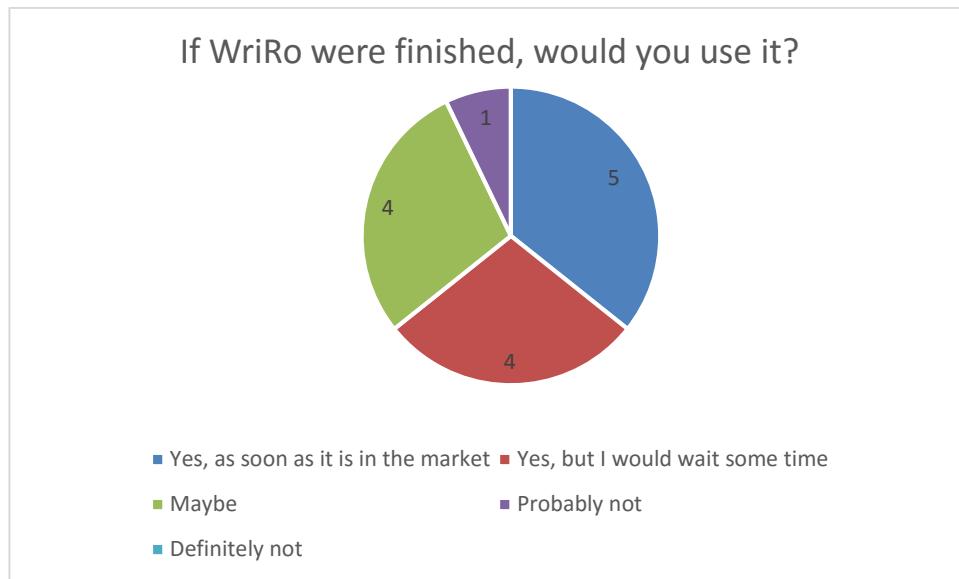


Figure 8: Würden Sie die fertige Applikation nutzen?

Ebenfalls 9 der Befragten würden die Applikation einem Freund bzw. anderen Autoren weiterempfehlen (6 extrem wahrscheinlich, 3 sehr wahrscheinlich). Dahingegen würden 5 der Befragten die Applikation eher nicht weiterempfehlen (3 einigermaßen wahrscheinlich, 2 nicht sehr wahrscheinlich).

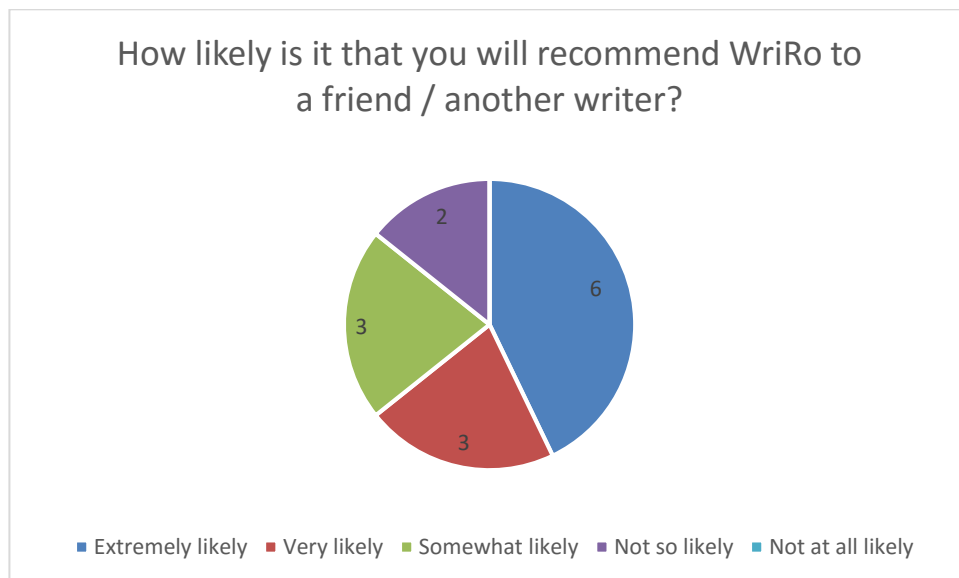


Figure 9: Würden Sie WriRo einem Freund empfehlen?

Besonders begeistert waren die Testpersonen von folgenden Funktionen bzw. Eigenschaften der Web-Applikation (siehe Figure 10):

- Einfachheit (8 Stimmen)
- Einfache Benutzbarkeit (8 Stimmen)
- Unbegrenzte Charakterprofile (11 Stimmen)
- Unbegrenzte Geschichtsprofile bzw. -seiten (11 Stimmen)
- Download der Geschichten als PDF (7 Stimmen)
- Abonnieren von Charakteren (7 Stimmen)
- Abonnieren von Geschichten (10 Stimmen)

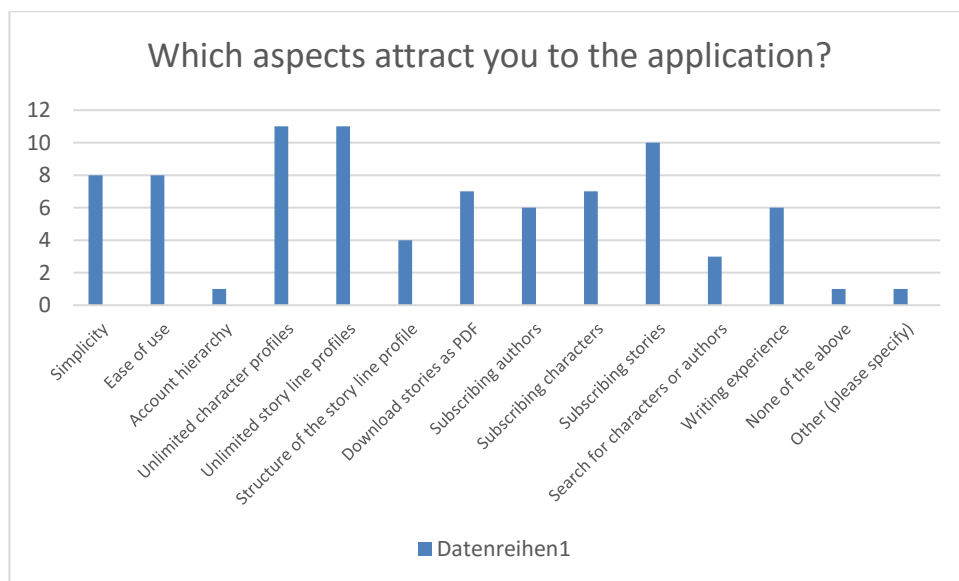


Figure 10: Was gefällt Ihnen besonders an der Applikation?

Dahingegen wurden folgende Funktionen und Eigenschaften von nur wenigen Personen als attraktiv angesehen (siehe Figure 10):

- Account-Hierarchie (1 Stimme)
- Aufbau des Geschichtsprofils (4 Stimmen)
- Freitextsuche nach Autor und Charakter (3 Stimmen)

Außerdem wurden die Testpersonen zu ihrer Meinung bezüglich noch geplanter Funktionen befragt. Sie wurden gefragt, welche der Funktionen zur Web-Applikation hinzugefügt werden sollen. Die meisten Stimmen erhielten dabei folgende Funktionen (vgl. Figure 11):

- Private und Öffentliche Nachrichten versenden (12 Stimmen)

- Hochladen und Einrichten eines Charakterprofil-Bildes (12 Stimmen)
- Favorisieren einer Geschichte (11 Stimmen)
- Auf beliebige Beiträge antworten (10 Stimmen)
- Übersicht aller Charaktere eines Autors auf der Profilseite des Autors (10 Stimmen)
- Notifikationen (10 Stimmen)
- Mobilversion der Applikation als App (iOS, Android, Windows) (10 Stimmen)

Die wenigsten Stimmen erhielten folgende Funktionen (vgl. Figure 11):

- Erstellen eines Event-Profiles (3 Stimmen)
- Gästeliste zum Event (3 Stimmen)
- Event-Kalender (3 Stimmen)

Die Auswertung des Freitextfeldes der Frage, was den Testpersonen nicht an der Applikation gefallen hat, ergab folgende Erkenntnisse.

11/06/2016 12:22:55	There seem to be some issues in the searching features and the creating of new accounts, because I accidentally created one twice just clicking on it, and I don't really know how that happened. Otherwise it was very good.
11/06/2016 12:35:12	Search needs to be modified so others can be found easier. That's my only complaint.
12/06/2016 12:33:29	I imagine that because the app is in it's early stages and not many people are on the app, there are not many people to follow or to read and RP with. otherwise, it is an interesting way to do RP online, in it's own domain.
12/06/2016 22:39:16	I didn't understand the meaning of the application. I didn't understand how to use it properly. I use Twitter and am good with continuing to do so. It is social and you can read everyone's stories as they happen. This seems private and only for you and the person you might write with. I didn't find anyone else when searching, and I didn't know how to find anyone else unless I knew their author name or character name before hand.
14/06/2016 17:00:38	It's still in develop mode. There is nothing I can point out specifically.
06/07/2016 10:32:05	Studies

Figure 11: Freitext-Angaben zu Figure 7

Einige Nutzer (3) gaben Mängel an der Suchfunktion an. Diese bestanden zum Teil aus fehlenden Informationstexten, wenn die Suche keine Ergebnisse lieferte, und zum anderen Teil aus fehlenden, ergänzenden Funktionen, die das Suchen nach Autoren oder Charakteren vereinfachen würden.

Desweiterem wurde einmal von Problemen bei der Account-Erstellung gesprochen.

Ein anderer Nutzer bemängelte den Mangel an Nutzern der Web-Applikation und ein anderer Nutzer gab an, dass die Applikation kompliziert sei.

Letzterer hat die Applikation augenscheinlich nicht verstanden. Er bzw. Sie gab an, „I didn't understand the meaning of the application. I didn't understand how to use it properly.“

Im Freitextfeld zur Frage wie die Applikation verbessert werden könnte wurde wiederholt angegeben, dass die Beiträge in Paragraphen unterteilt werden sollten bzw. die eingegebene Formatierung übernommen werden sollten, und dass es eine Auflistung aller Autoren bzw. aller aktiven Autoren geben sollte, und dass das Layout bzw. Design der Seite verbessert werden sollte (vgl. Figure 13). Außerdem sollten die aufgetretenen Bugs (wie Probleme beim Erstellen von Accounts oder Beiträgen) beseitigt werden.

Desweiterem wurden folgende Vorschläge gemacht:

- Upload von Bildern und GIFs
- Automatische Anzeige und Suche ähnlicher Accounts als Vorschläge
- Detaillierte Anleitung der Funktionen der Web-Applikation
- Klarere bzw. einfachere Struktur der Applikation (verbesserte Navigation)
- Suche von Nutzern verbessern
- Gewinnung von mehr Nutzern

Insgesamt ist das Ergebnis der Umfrage positiv ausgefallen. Die meisten der 14 Testpersonen waren mit der Applikation und dem Konzept zufrieden und würden die Applikation sowohl verwenden als auch weiterempfehlen.

Die geäußerten Verbesserungsvorschläge beziehen sich entweder auf Bugs oder werden von den geplanten Weiterentwicklungen bereits berücksichtigt.

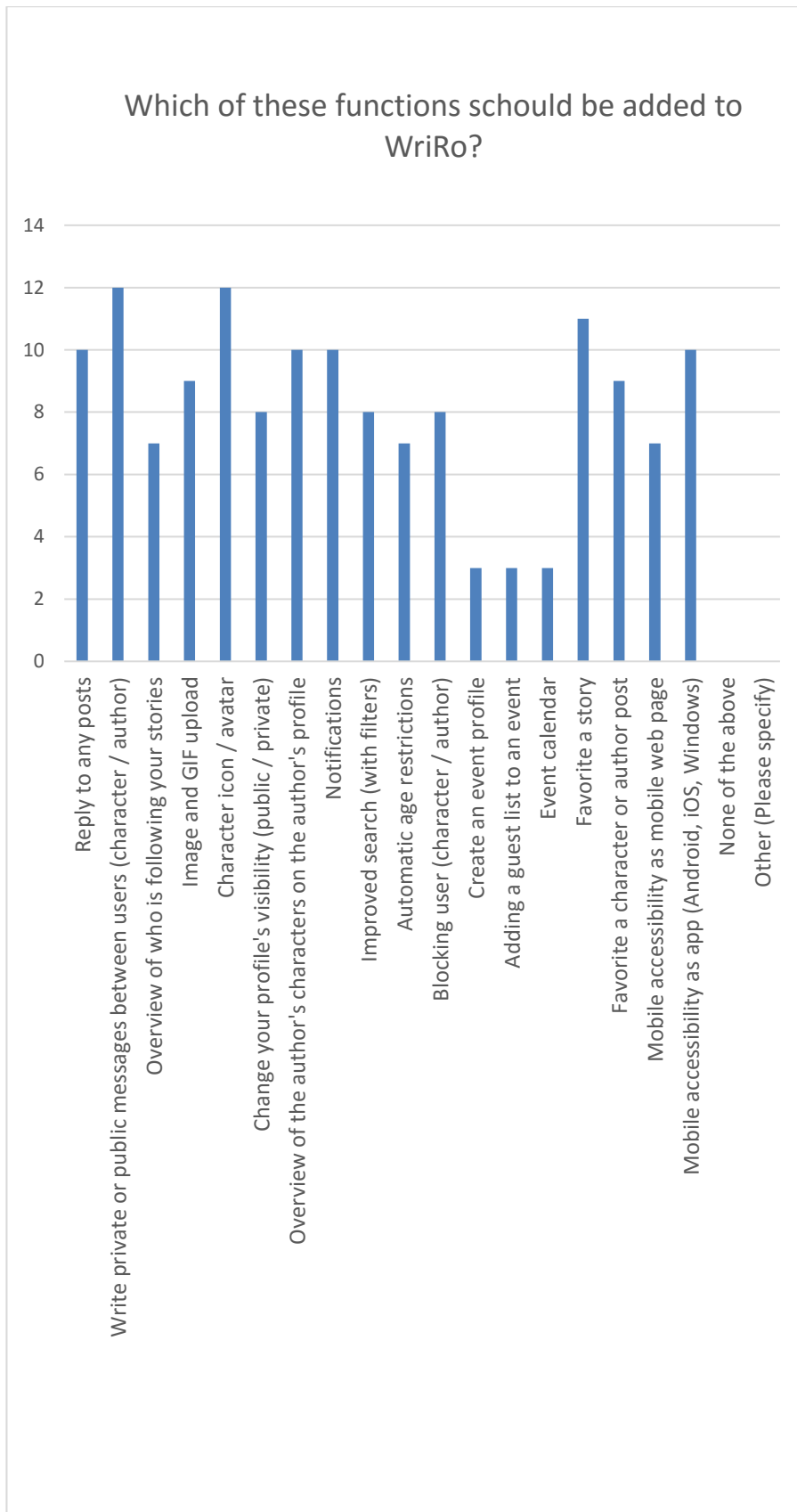


Figure 12: Welche dieser Funktionen sollten zu WriRo hinzugefügt werden?

06/06/2016 0:50:21	The option to use photos/gifs would be great, at the very least profile pictures for each character. I love the idea of being able to download SLs in PDF! It would be nice if I could post a long reply/starter/solo and have it in the format I intended, with paragraphs, and not just displayed as one long, confusing post.
10/06/2016 15:24:04	The concept per sé is good. Well thought out. Perhaps it'd be good if the interaction was easier or we, the users, could have a list of authors instead of looking for them by UN or name.
11/06/2016 12:26:46	There is no paragraph function in Storylines, as far as I am aware. I attempted to create a post with multiple paragraphs and they compiled at the end, which was irritating. I think the search function is a great idea, but maybe there could be a section for recommendations. Say if you subscribe to one writer, three more "similar writers" based on filters, tags, fandoms, age limit, etc appear, because that helps people network and meet new writers with whom they have things in common.
11/06/2016 12:46:23	Keep the instructions. I wasn't sure exactly what to press when I first entered the website, so they made my experience much less frustrating. Also, when I first made my character account, there were two boxes for bio. I would delete one. Lastly, I couldn't find a way to toggle back and forth from author profile to character profile, so if that doesn't exist, it should.
12/06/2016 4:11:00	Not at this moment.
12/06/2016 12:35:21	Perhaps if there were more people on the app to interact with, it would create more activity online.
12/06/2016 22:43:24	Make it easier to find someone else to RP with. It seems to be an app made for a group of people who wants to RP privately but online. How do I find others to RP with..?
13/06/2016 11:45:46	visibility of all authors online
13/06/2016 19:24:57	It just needs a better layout and it will be great.
14/06/2016 14:06:02	-
14/06/2016 17:02:23	None at present. We'll have to see how it goes and what needs tweaking.
06/07/2016 10:33:46	no. but i had to fill this field
06/07/2016 11:40:32	Fix bugs, have a better design, make structure more clear.

Figure 13: Was würden Sie an WriRo verbessern?

06/06/2016 0:50:21	Because it is a prototype and not a finished product there were a few kinks I'd like to be solved if the creator decides to go further with her idea. These kinks made the page a bit disorientating at times; like how searching for other authors only worked sometimes, and how downloading a PDF of a SL only showed you page 1.
10/06/2016 15:24:04	I believe it could be good but needs a lot of work, obviously.
11/06/2016 12:26:46	I think for a prototype it definitely exceeded my expectations. I could really see this as something that has immense potential in a community of people who are there to write stories, not deal with the extra things that are inevitable with twitter and tumblr rp. I was reminded of a traditional forum rp, but very much catered toward the writing community which I loved. Seeing the age limit on storylines was an excellent idea, and if it means that people who's "ages" (that they put on the site, so of course subjective to their truthfulness) are below the age limit cannot see it, I love it. A+.
11/06/2016 12:46:23	Ingenious idea. It makes creating and posting storylines so simple when compared to other means (Twitter, Tumblr), and puts all posts for the story in one place, making them easy to follow and read.
12/06/2016 12:35:21	The prototype worked, however it frequently asked me to login despite having already done so.
12/06/2016 22:43:24	I would want clear instructions how to use it, and why I should use it. Why would it be better than any other app for role play today? I didn't quite understand what I was supposed to do, except try and add characters and look for people. I didn't understand how I should find those I don't already know, if I asked them to join me on WriRo, so new authors to find are a bit of a hassle.
14/06/2016 17:02:23	Very well thought out. One of the better RP catered concepts I've come across in years.
06/07/2016 10:33:46	no. but i had to fill this field
06/07/2016 11:40:32	There is no landing page. clicking on the icon in the upper left corner always tells you to log in. The form is not https protected, so it is pretty unsecure. After clicking on add charater, the page seems to be broken. Posting a new message doesn't work properly. A page reload logs you out.

Figure 14: Nutzerkommentare zu WriRo

Obwohl das Ergebnis der Befragung sehr positiv ausgefallen ist, war und blieb die Beteiligung an dem Test ernüchternd. Von den ursprünglich zwanzig Testpersonen haben nur vierzehn an dem Test und der Befragung teilgenommen. Von diesen vierzehn Probanden haben lediglich fünf Beiträge zu einer Geschichte geschrieben. Dabei handelte es sich überwiegend um Solos, die ohne die Kollaboration mit anderen Nutzern ausgeführt wurden. In der gesamten Testphase von zwei Monaten wurde nur eine SL verfasst, die in der kurzen Zeit nicht abgeschlossen werden konnte.

In den Verbesserungsvorschlägen, die während der Befragung in den Freitextfeldern gemacht wurden, lassen sich keine Hinweise auf den Grund der mangelnden Beteiligung finden. Daher muss darauf geschlossen werden, dass die insgesamt schlechte Beteiligung persönliche Gründe hatte, die nicht im Zuge der Evaluation des Prototyps betrachtet werden können.

7.2 Verbesserungen als Resultat der Evaluation

Im Anschluss an die Auswertung der Evaluation wurden einige Verbesserungen an dem ersten Prototyp vorgenommen.

Zum einen wurde eine Auflistung aller registrierten Charakter- und Autor-Accounts, die über einen Button im Header der Applikation erreicht werden kann, hinzugefügt.

Außerdem wurden Paragraphen zu den Geschichts- sowie Autor- und Charakter-Beiträgen hinzugefügt, sodass das gewählte Format des Verfassers beibehalten wird.

Zudem wurden einige kleine Bugs, unter anderem bezüglich des PDF-Downloads beseitigt. Geschichtsbeiträge werden nicht länger abgeschnitten, wenn eine Seite überstiegen wird. Stattdessen wird der Beitrag auf der nächsten Seite fortgeführt.

Andere, aufwendigere Änderungen konnten aufgrund der gegebenen Zeit nicht durchgeführt werden. Sie werden im Kapitel 8.2 ausführlich erläutert.

8 Fazit

In diesem letzten Kapitel der Thesis soll ein kurzes Fazit, in Form einer Zusammenfassung der Ergebnisse, gegeben werden. Diesem folgt eine Erläuterung der geplanten Weiterentwicklung der Web-Anwendung, die über den Status eines Prototyps hinausgeht.

8.1 Fazit der Thesis

Mit Hilfe von JavaScript, genauer Backbone.js, im Frontend, Java, insbesondere Spring, im Backend und einer relationalen Datenbank konnte innerhalb von fünf Monaten ein Prototyp für eine Web-Applikation für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil, der alle Grundfunktionen unterstützt, entwickelt werden.

Die Client-Server-Struktur der Anwendung und das vorliegende MVC-ähnliche Modell erlauben eine hohe Flexibilität bei der Erweiterung der Anwendung oder dem Austauschen von Anwendungsteilen. So können im weiteren Verlauf der Entwicklung mit geringem Aufwand weitere Plattformen in diese Entwicklung mit einbezogen werden. Zum Beispiel kann eine mobile Anwendung in Form einer App (Android, iOS, oder Windows) entwickelt werden, die lediglich aus dem Client-Programm besteht und den bestehenden Web-Service nutzt.

Die Funktionen des Prototyps umfassen die Grundfunktionen und die wichtigsten Nebenfunktionen der Web-Anwendung. Mit Hilfe dieser Funktionen ist ein kollaboratives Verfassen von fiktionalen Texten im Rollenspiel-Stil bereits möglich.

Allerdings läuft der Prototyp bisher nur weitestgehend stabil, kleinere Fehler (meistens ausgelöst durch längere Ladezeiten) können noch auftauchen.

Zudem ist der Prototyp aufgrund einer mangelnden Sicherheit noch nicht marktfähig. Die Kommunikation zwischen Client und Server findet unverschlüsselt statt, auch die Passwörter der Nutzer werden unverschlüsselt in der Datenbank gespeichert, und der Web-Service hat keine Zugriffsbeschränkung. Lediglich ein CORS (Cross Origin Requests) – Filter wurde dem Web-Service hinzugefügt, sodass er nur Anfragen derselben IP-Adresse oder von der IP-Adresse des Clients zulässt.

Trotz des Funktionsumfangs und weitestgehend fehlerfreier Ausführung des Prototyps fiel die Teilnahme an der Evaluation ernüchternd aus, während das Ergebnis der Evalu-

ation sehr positiv ausgefallen ist (alle Teilnehmer der Evaluation waren vom Prototyp und dem zugrundeliegenden Konzept begeistert). Hier gilt es abzuwarten, ob eine erste marktreife Version der Anwendung größeren Anklang findet.

Insgesamt lässt sich feststellen, dass der Prototyp auf einem guten Stand ist, da er alle Funktionen für das kollaborative Verfassen einer Geschichte im Rollenspiel-Stil unterstützt, und weitestgehend stabil und fehlerfrei läuft.

Obwohl die Testphase nur eine geringe Beteiligung aufzuweisen hat, animieren die Aussagen der Teilnehmer zu einer Weiterentwicklung der Anwendung. Die nächsten Schritte dieser Weiterentwicklung und die geplanten zusätzlichen Funktionen der Anwendung werden im nächsten Kapitel erläutert.

8.2 Weiterentwicklung der Software

Bei der Applikation WriRo, die im Laufe dieser Bachelorarbeit entwickelt wurde, handelt es sich um einen ersten Prototyp der Web-Applikation. Eine Weiterentwicklung dieser Anwendung bietet sich an.

Zunächst sollten einige Verbesserungen an dem Prototyp vorgenommen werden. Diese beinhalten eine Verkürzung der Ladezeiten, welche in einigen Fällen die angegebenen 3 Sekunden übersteigen. Dies kann durch eine Verbesserung der Datenbankverbindung und der Verarbeitung der Daten durch den Web-Client realisiert werden. Aufgrund eines noch nicht behobenen Fehlers, wird für jede Datenbankanfrage eine Datenbankverbindung geöffnet und nicht wieder geschlossen. Aufgrund dessen wird die Applikation langsamer, umso länger sie läuft. Eine Verbesserung der Verarbeitung der Daten im Client-Programm, sowie das parallele Laden mehrerer Daten sollten ebenfalls zu einer Verbesserung der Ladezeiten beitragen.

Als nächstes sollte ein Session-Management eingeführt werden, dass essentielle Session-Parameter in Cookies speichert. Dabei sollten sowohl Benutzer-Daten als auch csrf_tokens gespeichert werden, sodass der Benutzer auch nach erneutem laden der Applikation eingeloggt bleibt und die Möglichkeit hat dauerhaft eingeloggt zu bleiben, und eine Authentifizierung mit dem Web-Server (zum Schutz vor CSRF) möglich ist.

Zudem sollten die Fehlermeldungen auf der Seite verbessert werden. Durch die Fehlermeldung muss beschrieben werden, was für ein Fehler aufgetreten ist und wie der Nutzer auf diesen Fehler reagieren sollte, sofern diese Aussagen über den Fehler getrof-

fen werden können. Wenn ein Formular nicht abgeschickt werden konnte, weil ein essentielles Feld nicht ausgefüllt wurde, muss das entsprechende Feld markiert werden und darauf hingewiesen werden, dass dieses Feld ausgefüllt werden muss. Sollte ein Feld fehlerhaft ausgefüllt worden sein, muss dieser Fehler explizit genannt werden (zum Beispiel das Fehlen eines „@“ in der Email-Adresse).

In einem zweiten Schritt sollten Verbesserungen bezüglich der Sicherheit der Applikation vorgenommen werden.

Zunächst sollte die Übertragung von Daten über HTTP SSL-verschlüsselt ablaufen. Dazu muss sowohl der RESTful Web-Client als auch der Web-Service auf HTTPS umgestellt werden, und ein SSL-Zertifikat für die Web-Applikation beantragt werden.

Außerdem sollten die Passwörter der Nutzer verschlüsselt in der Datenbank gespeichert werden, um zu verhindern, dass diese durch Hacking-Angriffe offenliegen. Dies kann zum Beispiel mit dem spring-security-crypto Modul realisiert werden. Springs Passwortverschlüsselung unterstützt verschiedene Verschlüsselungsmethoden, wie zum Beispiel SCrypt, PBKDF2, und ByteEncryption (vgl. Spring Security Reference).

Nach der Einführung von Cookies, sollte der RESTful Web-Service durch eine Authentifizierung und die Verwendung von csrf_token abgesichert werden. Die Authentifizierung kann ebenfalls durch Spring Security realisiert werden, und sichert den Web-Service gegenüber unautorisierten Zugriffen (vgl. Spring Security Reference). Dies wiederum sichert die Daten der Nutzer. Die Verwendung des csrf_tokens ist eine sichere und einfache Methode des Schutzes vor CSRF (Cross Side Request Forgery) (vgl. Spring Security Reference).

Zudem sollte die Sicherheit der verwendeten Formulare innerhalb der Web-Applikation verbessert werden, sodass die Applikation besser vor SQL Injection geschützt ist. Außerdem schützt dieser Schritt vor falschen Benutzerdaten, zum Beispiel ungültige Geburtsdaten.

In einem dritten Schritt sollte das Layout der Web-Applikation überarbeitet werden. Die Seite sollte responsiv sein, sodass sie auch mobil auf Tablets oder Mobiltelefonen verwendet werden kann. Außerdem sollten einige der Textelemente in der Navigation

durch passende Icons ersetzt werden. Die Navigationsleiste sollte außerdem einklappbar sein, sodass sie bei Nichtgebrauch ausgeblendet werden kann.

Nach diesen drei, essentiellen, Schritten kann die Applikation als erste fertige Version (nicht nur Prototyp) veröffentlicht werden.

Anschließend sollten, in einem vierten Schritt, weitere Funktionen zu der Applikation hinzugefügt werden. Als Grundlage der Wichtigkeit der Funktionen kann das Ergebnis der im Kapitel 7.1 beschriebenen Umfrage herangezogen werden.

Demnach sollten zunächst Messages, in Form von privaten und öffentlichen Nachrichten, und Charakter-Icons (bzw. Avatare) eingeführt werden.

Anschließend sollte eine Funktion zum Favorisieren von Geschichten zum Geschichtsprofil hinzugefügt werden.

Als nächstes sollten Funktionen zum Antworten auf beliebige Beiträge, eine Liste aller Charaktere eines Autors im Autorprofil, und Notifikationen (Email und Browser) erstellt werden. Zur Erstellung dieser Liste aller Charaktere, müsste die entsprechende Information lediglich mit Hilfe eines Joins aus der Datenbank herausgeholt (die Tabellen **characters** und **author** sind über Schlüssel-Fremdschlüssel-Beziehungen miteinander verknüpft) und in einer passenden Stelle des Autorprofils eingefügt werden.

Desweiterem sollte ein Bild- und GIF-Upload innerhalb von Beiträgen, und eine Funktion zum Favorisieren von Beiträgen implementiert werden.

Als fünftes sollte die Suchfunktion durch Filter erweitert werden, Berechtigungseinstellungen (öffentlich und privat) für Accounts und Profilseiten, und eine Funktion zum Blocken von Nutzern implementiert werden.

Für Berechtigungseinstellungen können im Backend Funktionen der Autorisierung über Rollen verwendet werden, während Informationen zu geblockten Nutzern in der Datenbank in einer eigenen Tabelle, die eine Verknüpfung zum geblockten und blockenden Nutzer (analog zur Tabelle **character_subscription**) enthält, gespeichert werden können. Im Frontend können ebenfalls Mechanismen der Autorisierung angewendet werden, um die angezeigten Inhalte der Seite auf Basis der gespeicherten Daten in der Datenbank (Rollen, geblockte, und abonnierte Nutzer) zu ändern.

Als nächstes sollten Funktionen zur automatisierten Altersbeschränkung und zur Vergabe sonstiger Berechtigungen (lesen, schreiben), und eine Liste aller Abonnenten einer Geschichte auf der Seite der entsprechenden Geschichte eingeführt werden. Zur automatisierten Altersbeschränkung kann das angegebene Alter des Autors mit der angegebenen Altersbeschränkung auf Profildaten (Autor, Charakter, Geschichte) verglichen werden. Auf Basis dieses Vergleichs (ist das Alter des Nutzers gleich oder höher) kann das System entsprechend handeln; Anzeigen der Seite oder Anzeigen einer Meldung, die auf die Altersbeschränkung hinweist. Die Abonnenten einer Geschichte sind, ebenso wie die Charaktere eines Autors, bereits in der Datenbank gespeichert und müssen lediglich abgefragt und an entsprechender Stelle angezeigt werden. Die eingestellten sonstigen Berechtigungen können in der Datenbank (**roles**) hinterlegt werden.

Als letztes sollten im Zuge der Weiterentwicklung der Applikation Funktionen für Events entwickelt werden und verschiedene Sprachen implementiert werden. Diese Events enthalten ein Event-Profil und eine Event-Seite, eine Gäste- bzw. Teilnehmerliste des Events, und einen Event-Kalender, der den Event-Termin automatisch in mehreren Zeitzonen anzeigt. Die englische Sprache der Webseite sollte mindestens durch die deutsche ergänzt werden.

In einem fünften Schritt, nachdem alle erweiterten Funktionen zur Web-Applikation hinzugefügt wurden, sollten mobile Apps für die Web-Applikation entwickelt werden. Diese App sollte zunächst für Android, anschließend für iOS, und schließlich für Windows Mobile (als Windows 10 Universal App) verfügbar sein und alle oben aufgeführten Funktionen unterstützen.

Nach der Implementierung der oben genannten Funktionen sollte jeweils, nach jedem Schritt, eine ausführliche Testphase mit geeigneten Softwaretests und Benutzertests folgen.

Nach diesem fünften und letzten bisher geplanten Schritt erfüllt die Web-Applikation alle Grund- und erweiterten Funktionen des Funktionsumfangs, welcher mit Hilfe der Anforderungsanalyse erstellt wurde (siehe Kapitel 4.3).

9 Literaturverzeichnis

Allgemeine Geschäftsbedingungen. Online verfügbar unter

<https://www.facebook.com/terms.php?ref=pf>, zuletzt geprüft am 23-Aug-16.

Arjoranta, Jonne (2011): Defining Role-Playing Games as Language-Games, S. 3–17.

Online verfügbar unter

https://www.researchgate.net/profile/Jonne_Arjoranta/publication/262388323_Defining_Role-Playing_Games_as_Language-Games/links/00b495379c1dc4522b000000.pdf.

Backbone.js (06-May-16). Online verfügbar unter <http://backbonejs.org/>, zuletzt aktualisiert am 06-May-16, zuletzt geprüft am 23-Aug-16.

Berger, Florian; Marbach, Alexander: Workshop: Pen-and-Paper Role-Playing, Bd. 5334, S. 330, zuletzt geprüft am 23-Aug-16.

Bruckman, Amy (2013): Proceedings of the 2013 conference on Computer supported cooperative work companion. New York, NY: ACM, zuletzt geprüft am 25.08.2016.

Carpe Noctem. Online verfügbar unter <http://tdvfantasy.siteboard.eu/>, zuletzt geprüft am 25.08.2016.

Conallen, Jim (1999): Modeling Web application architectures with UML. In: *Commun. ACM* 42 (10), S. 63–70. DOI: 10.1145/317665.317677.

Das Twitter Glossar. Online verfügbar unter

<https://support.twitter.com/articles/473379?lang=de>, zuletzt geprüft am 23-Aug-16.

Duus Henriksen, Thomas (Hg.) (2011): Think larp. Academic writings from KP2011. Rollespilsakademiet (firma); Knudepunkt (konferenz). 1. edition, 1. oplag. Copenhagen: Rollespilsakademiet, zuletzt geprüft am 23-Aug-16.

Facebook hates roleplayers | OngoingWorlds roleplay blog. Online verfügbar unter <http://www.ongoingworlds.com/blog/2010/12/facebook-hates-roleplayers/>, zuletzt geprüft am 23.08.2016.

FOCUS (2012): Facebook-Geschichte als Chronik. Online verfügbar unter http://www.focus.de/digital/internet/facebook/tid-24930/die-geschichte-des-sozialen-netzwerks-facebooks-eroberung-der-welt-facebook-geschichte-als-chronik_aid_709860.html, zuletzt aktualisiert am 02.02.2012, zuletzt geprüft am 25.08.2016.

Freigeben eines Dokuments in SharePoint oder auf OneDrive - Word. Online verfügbar unter <https://support.office.com/de-de/article/Freigeben-eines-Dokuments-in-SharePoint-oder-auf-OneDrive-807de6cf-1ece-41b9-a2b3-250d9a48f1e8>, zuletzt geprüft am 23-Aug-16.

Getting Started: Creating First Yii Application | The Definitive Guide to Yii | Yii PHP Framework. Online verfügbar unter <http://www.yiiframework.com/doc/guide/1.1/en/quickstart.first-app>, zuletzt geprüft am 23-Aug-16.

Gordon, Andrew S.; Swanson, Reid (2008): Say Anything: A Massively Collaborative Open Domain Story Writing Companion. In: Ulrike Spierling und Nicolas Szilas (Hg.): Interactive Storytelling: First Joint International Conference on Interactive Digital Storytelling, ICIDS 2008 Erfurt, Germany, November 26-29, 2008 Proceedings. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 32–40. Online verfügbar unter http://dx.doi.org/10.1007/978-3-540-89454-4_5.

Hadley, Marc J. (2006): Web Application Description Language (WADL), zuletzt geprüft am 23-Aug-16.

How to Roleplay on Tumblr. Online verfügbar unter <http://www.wikihow.com/Roleplay-on-Tumblr>, zuletzt geprüft am 23-Aug-16.

How-To: Create a REST API | Wiki | Yii PHP Framework. Online verfügbar unter <http://www.yiiframework.com/wiki/175/how-to-create-a-rest-api/>, zuletzt geprüft am 23-Aug-16.

Internetauftritt der Bundesbeauftragten für den Datenschutz und die Informationsfreiheit - Datenschutz - Einwilligung. Online verfügbar unter http://www.bfdi.bund.de/DE/Datenschutz/Ueberblick/MeineRechte/Artikel/Einwilligung.html;jsessionid=B904EE743ABA8166CD7E93C845009F81.1_cid354?nn=5217008, zuletzt geprüft am 23-Aug-16.

Jenderek, Bastian: Echtzeitabenteuer ohne Grafik und Sound, S. 313–332, zuletzt geprüft am 23-Aug-16.

Kaufmann, Michael; Meier, Andreas (2016): SQL- & NoSQL-Datenbanken. 8., überarb. u. erw. Aufl. 2016. Berlin, Heidelberg, s.l.: Springer Berlin Heidelberg (eXamen.press). Online verfügbar unter <http://dx.doi.org/10.1007/978-3-662-47664-2>.

- Kirpal, Alfred; Vogel, Andreas (2006): Neue Medien in einer vernetzten Gesellschaft. Zur Geschichte des Internets und des World Wide Web. In: *N.T.M.* 14 (3), S. 137–147. DOI: 10.1007/s00048-006-0239-5.
- Klute, Rainer: Internet-Informationendienste mit Mosaic - Zusammengewebt. In: *iX - Magazin für professionelle Informationstechnik* 1994 (2), S. 150.
- Lourenço, João Ricardo; Abramova, Veronika; Vieira, Marco; Cabral, Bruno; Bernardino, Jorge: NoSQL Databases: A Software Engineering Perspective, Bd. 353, S. 741–750, zuletzt geprüft am 23-Aug-16.
- Lowry, Paul Benjamin; Curtis, Aaron; Lowry, Michelle René (2004): BUILDING A TAXONOMY AND NOMENCLATURE OF COLLABORATIVE WRITING TO IMPROVE INTERDISCIPLINARY RESEARCH AND PRACTICE. In: *Journal of Business Communication* 41 (1), S. 66–99.
- Meier, Andreas (2016): Zur Nutzung von SQL- und NoSQL-Technologien. In: *HMD* 53 (4), S. 415–427. DOI: 10.1365/s40702-016-0225-x.
- Montfort, Nick: *Curveship*, S. 55–62, zuletzt geprüft am 23-Aug-16.
- Neumann, Alexander; Developer, heise (24-Apr-14): Studie zur Softwarequalität: Open Source schlägt proprietär. Heise Medien. Online verfügbar unter <http://www.heise.de/developer/meldung/Studie-zur-Softwarequalitaet-Open-Source-schlaegt-proprietaeer-2175954.html>, zuletzt aktualisiert am 24-Apr-14, zuletzt geprüft am 23-Aug-16.
- Parker, Zachary; Poe, Scott; Vrbsky, Susan V.: Comparing NoSQL MongoDB to an SQL DB. In: Ashraf Saad (Hg.): *the 51st ACM Southeast Conference*. Savannah, Georgia, S. 1.
- Piepmeyer, Lothar (2011): *Grundkurs Datenbanksysteme. Von den Konzepten bis zur Anwendungsentwicklung*. München, Wien: Hanser.
- Richtlinien für Parodie-, Kommentar- oder Fan-Accounts. Online verfügbar unter <https://support.twitter.com/articles/20170144>, zuletzt geprüft am 23-Aug-16.
- RolePlay.me | RolePlay Online. Online verfügbar unter <http://www.roleplay.me/faq>, zuletzt geprüft am 23-Aug-16.

Slator, Brian M.; Borchert, Otto; Brandt, Lisa; Chaput, Harold; Erickson, Kellie; Groesbeck, Gabriel et al.: From Dungeons to Classrooms: The Evolution of MUDs as Learning Environments, Bd. 62, S. 119–159, zuletzt geprüft am 23-Aug-16.

Spell, Brett (2015): Going Inside Java: Apress. Online verfügbar unter http://link.springer.com/chapter/10.1007/978-1-4842-0641-6_1/fulltext.html.

Spring Projects. Online verfügbar unter <https://spring.io/projects>, zuletzt geprüft am 23-Aug-16.

Spring Security Reference. Online verfügbar unter <http://docs.spring.io/spring-security/site/docs/4.1.2.RELEASE/reference/htmlsingle/>, zuletzt geprüft am 23.08.2016.

The Strange World of Internet Role-Play Has Gone Mainstream. Online verfügbar unter <http://motherboard.vice.com/read/the-strange-world-of-internet-roleplay-is-getting-less-strange>, zuletzt geprüft am 23-Aug-16.

Tour: Gruppen - Splitstory. Online verfügbar unter <http://www.splitstory.com/de/tour/groups>, zuletzt geprüft am 23-Aug-16.

Tour: Ideen - Splitstory. Online verfügbar unter <http://www.splitstory.com/de/tour/ideas>, zuletzt geprüft am 23-Aug-16.

Tour: Texte und Bilder - Splitstory. Online verfügbar unter <http://www.splitstory.com/de/tour/texts-and-images>, zuletzt geprüft am 23-Aug-16.

Tour: Überblick - Splitstory. Online verfügbar unter <http://www.splitstory.com/de/tour>, zuletzt geprüft am 23-Aug-16.

Tutorial: Create a Roleplaying Blog on Tumblr | Personal Reflections on Word-Press.com. Online verfügbar unter <https://personalreflections2014.wordpress.com/2014/01/22/tutorial-create-a-roleplaying-blog-on-tumblr/>, zuletzt geprüft am 23-Aug-16.

Über | Tumblr. Online verfügbar unter <https://www.tumblr.com/about>, zuletzt geprüft am 23.08.2016.

Walter, Thomas: Entwicklung der Web-Programmierung, S. 3–24, zuletzt geprüft am 23-Aug-16.

Witte, Frank (2016): Testmanagement und Softwaretest. Wiesbaden: Springer Fachmedien Wiesbaden, zuletzt geprüft am 23-Aug-16.

10 Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbständig und ohne unzulässige fremde Hilfe angefertigt habe. Alle verwendeten Quellen und Hilfsmittel sind angegeben.

Furtwangen, den 31.08.2016

X *J. Jurrut*
Jennifer Jurrut

Jennifer Jurreit	Walter	Eisenbiegler
Konzeption und prototypische Entwicklung einer Web-Applikation für kollaboratives Schreiben fiktiver Texte im Rollenspiel-Stil		SS 2016