



HTL - Perg
Höhere Abteilung für Informatik

Diplomarbeit

P@BS Connect

Webapplikation für die Verwaltung des Absolventenverbandes der HTL Perg

Projektteam: Lukas Huber
 Christoph Penner
 Daniel Tüchler

Projektbetreuer: Direktor Prof. Dipl.-Ing. Christian Reisinger

In Zusammenarbeit mit dem Absolventenverband der HTL Perg
Obmann Herr Prof. Dipl.-Ing. Gerhard Zweimüller

Bearbeitungszeitraum: 03.10.2016 – 05.04.2017

„At its heart, engineering is about using science to find creative, practical solutions. It is a noble profession.”

(Queen Elizabeth II)

„Being able to talk to people, to transmit images, flying, accessing vast amounts of data like an oracle. These are all things that would have been considered magic a few hundred years ago.

So, engineering is, for all intents and purposes, magic, and who wouldn't want to be a magician?”

(Elon Musk)

“If you think math is hard, try web design.”


(Trish Parr)


„The purpose of Software Engineering is to control complexity, not to create it.”


(Pamela Zave)

1 Eidesstattliche Erklärung

Hiermit versichern wir, die vorliegende Arbeit selbständig, ohne fremde Hilfe und ohne Benutzung anderer als der von uns angegebenen Quellen angefertigt zu haben. Alle Stellen, die wörtlich oder sinngemäß aus fremden Quellen direkt oder indirekt übernommen wurden, sind als solche gekennzeichnet.

Perg, 20.04.2017 Unterschrift 
(Lukas Huber)

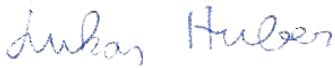
Perg, 20.04.2017 Unterschrift 
(Christoph Penner)


Perg, 20.04.2017 Unterschrift 
(Daniel Tüchler)


2 Gender-Erklärung

Für die bessere Lesbarkeit werden in dieser Diplomarbeit personenbezogene Bezeichnungen, die sich zugleich auf Frauen und Männer beziehen, generell nur in der im Deutschen üblichen maskulinen Form angeführt.

Dies soll jedoch keinesfalls eine Geschlechterdiskriminierung oder eine Verletzung des Gleichheitsgrundsatzes zum Ausdruck bringen.

Perg, 20.04.2017 Unterschrift 
(Lukas Huber)

Perg, 20.04.2017 Unterschrift 
(Christoph Penner)

Perg, 20.04.2017 Unterschrift 
(Daniel Tüchler)

3 Danksagung

Wir möchten uns bei sämtlichen Personen und Organisationen bedanken, die diese Diplomarbeit möglich gemacht haben.

Besonders erwähnen möchten wir unseren Betreuungslehrer Herrn Direktor Prof. Dipl.-Ing. Christian Reisinger, der uns stets für Fragen zur Seite stand und uns unterstützte, sowie den Obmann des Absolventenverbandes der HTL Perg, Herrn Prof. Dipl.-Ing. Gerhard Zweimüller, der als Auftraggeber und Ansprechperson das Projekt ermöglicht hat.

Weiters bedanken wir uns bei Herrn Prof. Ing. Werner Schöllner sowie Herrn Clemens Denzinger, die uns als IT-Administratoren der HTL Perg die technische Basis durch die schuleigene Serverumgebung zur Verfügung gestellt haben. Zusätzlich möchten wir unseren Dank Herrn Prof. Dipl.-Ing. Christian Aberger aussprechen, der uns bei technischen Fragen mit Rat und Tat zur Seite stand.

4 Impressum

Schule

HTBLA Perg für Informatik
Machlandstraße 48
4320 Perg

Schuljahr

2016/17

Klasse

5AHIF

Projektname

P@BS Connect

Projektteam

Lukas Huber
Christoph Penner
Daniel Tüchler

Betreuungslehrer

Direktor Prof. Dipl.-Ing. Christian Reisinger

Auftraggeber

P@BS – EDV Absolventen und Förderer der HTL–Perg
Machlandstraße 48
4320 Perg

Obmann und Ansprechperson: Prof. Dipl.-Ing. Gerhard Zweimüller

5 Inhaltsverzeichnis

1	EIDESSTATTLICHE ERKLÄRUNG	3
2	GENDER-ERKLÄRUNG	4
3	DANKSAGUNG	5
4	IMPRESSUM	6
5	INHALTSVERZEICHNIS	7
6	KURZBESCHREIBUNG	12
7	ABSTRACT	14
8	STAKEHOLDER	16
8.1	TEAM	16
8.1.1	<i>Lukas Huber</i>	16
8.1.2	<i>Christoph Penner</i>	17
8.1.3	<i>Daniel Tüchler</i>	18
8.2	BETREUUNG	19
8.2.1	<i>Direktor Prof. Dipl.-Ing. Christian Reisinger</i>	19
8.3	LEHRANSTALT	19
8.3.1	<i>HTL Perg, Höhere Abteilung für Informatik</i>	19
8.4	AUFTRAGGEBER.....	20
8.4.1	<i>P@BS – EDV Absolventen und Förderer der HTL – Perg</i>	20
8.4.2	<i>Obmann Prof. Dipl.-Ing. Gerhard Zweimüller</i>	21
9	EINLEITUNG	22
9.1	IST-ZUSTAND.....	22
9.2	MOTIVATION	23
9.3	ZIEL DER ARBEIT	23
9.3.1	<i>Geschäftsziele</i>	23
9.4	FUNKTIONSBEREICHE DES WEBPORTALS.....	24
9.4.1	<i>Rollensystem</i>	24
9.4.2	<i>Informelle Homepage</i>	24
9.4.3	<i>Stammdatenverwaltung</i>	25
9.4.4	<i>Newsverwaltung</i>	25
9.4.5	<i>Veranstaltungsverwaltung</i>	25
9.4.6	<i>Stellenangebotsverwaltung</i>	25
9.4.7	<i>Mitgliedsbeitragsverwaltung</i>	25
9.4.8	<i>HTL-Erfolgsgeschichten</i>	26

9.4.9	Benachrichtigungen.....	26
10	GRUNDLAGEN	27
10.1	VERWENDETE TECHNOLOGIEN	27
10.1.1	Ubuntu.....	27
10.1.2	SQL (Structured Query Language).....	27
10.1.3	MySQL.....	28
10.1.4	Java-Technologie.....	28
10.1.5	Java (Programmiersprache).....	29
10.1.6	WildFly.....	29
10.1.7	Java Persistence API.....	30
10.1.8	HTML.....	31
10.1.9	CSS.....	31
10.1.10	PHP.....	31
10.1.11	JavaScript (JS).....	32
10.1.12	Angular 2	32
10.1.13	TypeScript	32
10.1.14	Git.....	33
10.1.15	Joomla.....	33
10.2	VERWENDETE ENTWICKLUNGSSYSTEME.....	34
10.2.1	MySQL Workbench.....	34
10.2.2	Eclipse.....	35
10.2.3	JetBrains WebStorm.....	36
10.3	VERWENDETE BIBLIOTHEKEN UND PLUG-INS.....	37
10.3.1	iText.....	37
10.3.2	JA Builder.....	37
10.3.3	Akeeba Backup und Kickstart.....	37
10.4	VERWENDETE ARCHITEKTUREN.....	38
10.4.1	DAO (Data Access Object).....	38
10.4.2	REST (Representational State Transfer).....	38
10.4.3	JWT (JSON Web Token).....	39
10.5	VERWENDETE TEMPLATES.....	41
10.5.1	Angle Bootstrap Admin Template.....	41
10.6	SONSTIGE VERWENDETE SOFTWARE	42
10.6.1	PuTTY.....	42
10.6.2	Notepad++.....	43
10.6.3	VMware vSphere Client.....	44
10.6.4	SourceTree.....	45

10.6.5	<i>GitKraken</i>	45
10.6.6	<i>Microsoft Office 2016</i>	46
10.6.7	<i>Dropbox</i>	46
10.6.8	<i>GIMP</i>	47
10.6.9	<i>Adobe Photoshop</i>	47
10.6.10	<i>FileZilla Client</i>	48
10.6.11	<i>Telegram</i>	48
10.6.12	<i>Postman</i>	49
11	PLANUNG UND REALISIERUNG	50
11.1	CHANGE-REQUEST.....	50
11.2	MEILENSTEINE VOR DEM CHANGE-REQUEST	51
11.3	MEILENSTEINE NACH DEM CHANGE-REQUEST	52
11.4	PROJEKTSTRUKTURPLAN	53
11.5	VERANTWORTLICHKEITEN (IMV-MATRIX).....	54
11.6	FUNKTIONALE IDEE.....	55
11.6.1	<i>Absolventen & Firmen</i>	55
11.6.2	<i>Administrator</i>	56
11.7	USE-CASE-DIAGRAMME.....	57
11.7.1	<i>Events</i>	57
11.7.2	<i>Stellenangebote</i>	57
11.7.3	<i>Mitgliedsbeitrag</i>	58
11.7.4	<i>Erfolgsgories</i>	58
11.8	ARCHITEKTUR.....	59
11.8.1	<i>Backend</i>	60
11.8.2	<i>Frontend</i>	60
11.9	DESIGN DES WEBPORTALS - SEITENAUFBAU	61
11.9.1	<i>Login</i>	61
11.9.2	<i>Side-Menu</i>	62
11.9.3	<i>Startseite</i>	63
11.9.4	<i>Profil</i>	64
11.9.5	<i>Jobs – Funktionsübersicht</i>	66
11.9.6	<i>Jobs – Offene Stelleninseerte</i>	67
11.9.7	<i>Jobs – Stellenangebote verwalten</i>	68
11.9.8	<i>Jobs – Stelleninserat hochladen</i>	69
11.9.9	<i>Jobs – Administratorfunktionen</i>	70
11.9.10	<i>Events bzw. News – Übersicht</i>	71
11.9.11	<i>Events bzw. News – Einzelansicht</i>	72

11.9.12	<i>Events bzw. News erstellen und bearbeiten</i>	73
11.9.13	<i>Erfolgsstories</i>	74
11.9.14	<i>Erfolgsstories – Administratorfunktionen</i>	75
11.9.15	<i>Mitgliedsbeitrag</i>	76
12	IMPLEMENTIERUNG	77
12.1	BACKEND	78
12.1.1	<i>Datenmodell</i>	78
12.1.2	<i>Allgemeine Informationen</i>	79
12.1.3	<i>Tabellenbeschreibungen</i>	80
12.1.4	<i>Datenbankoperationen</i>	89
12.1.5	<i>REST-API</i>	90
12.1.6	<i>Statuscodes</i>	91
12.1.7	<i>Zugriff auf die Datenbank</i>	92
12.1.8	<i>Kommunikation per REST-API</i>	94
12.1.9	<i>Authentifizierung</i>	96
12.1.10	<i>Passwort-Hashing</i>	99
12.1.11	<i>Generierung der HTL-Erfolgsstories</i>	101
12.1.12	<i>Dateiupload</i>	105
12.1.13	<i>Mail-System</i>	107
12.2	FRONTEND	109
12.2.1	<i>Login</i>	109
12.2.2	<i>Dateiupload</i>	110
12.2.3	<i>Mitgliedsbeitrag</i>	110
12.2.4	<i>CSV-Import von neuen Benutzern</i>	110
13	ERGEBNIS	112
13.1	EINSTIEG IN WEBPORTAL	112
13.2	FUNKTIONEN AUS ABSOLVENTENSICHT	114
13.2.1	<i>Bearbeiten der Stammdaten</i>	114
13.2.2	<i>Einsehen und Kommentieren von News</i>	115
13.2.3	<i>Einsehen von aktuellen Jobinseraten</i>	116
13.2.4	<i>Einsehen, Kommentieren von und Teilnehmen an Events</i>	117
13.2.5	<i>Hochladen einer Erfolgsstory</i>	118
13.2.6	<i>Bezahlung des Mitgliedbeitrages</i>	118
13.3	FUNKTIONEN AUS FIRMENSICHT	119
13.3.1	<i>Hochladen von Jobinseraten</i>	119
13.3.2	<i>Verwalten der persönlichen Jobinserte</i>	120
13.4	FUNKTIONEN AUS ADMINISTRATOR-SICHT	121

13.4.1	Übersicht über alle Benutzer	122
13.4.2	Bearbeiten einzelner Benutzer	122
13.4.3	Löschen von Benutzern	122
13.4.4	Anlegen von neuen Benutzern	123
13.4.5	Importieren von neuen Benutzern	123
13.4.6	Anzeigen aller Benutzer-E-Mails in einem Textfeld	123
13.4.7	Erstellen von neuen News	124
13.4.8	Bearbeiten und Löschen von News	124
13.4.9	Erstellen von neuen Events	125
13.4.10	Bearbeiten und Löschen von Events	125
13.4.11	Verwalten der Erfolgsgeschichten	126
13.5	BENACHRICHTIGUNGEN	126
14	RESÜMEE	128
15	AUFGABENVERTEILUNG	130
15.1	LUKAS HUBER	130
15.2	CHRISTOPH PENNER	131
15.3	DANIEL TÜCHLER	132
16	GLOSSAR	133
17	ANHANG	138
17.1	DOKUMENTATION DER REST-API	138
17.1.1	Auth	139
17.1.2	PCUser	140
17.1.3	PCNews	141
17.1.4	PCEvent	142
17.1.5	PCFileUpload	143
17.1.6	PCJobAd	144
17.1.7	PCNotificator	145
17.1.8	PCStory	146
17.2	AUSZUG AUS DER PROJEKTDOKUMENTATION	147
17.2.1	Risikomanagement	147
17.2.2	Auszug Change-Request	148
17.2.3	Teilnahmeformular	149
17.3	DIPLOMARBEITSPAKAT	151
17.4	LITERATURVERZEICHNIS	152
17.5	ABBILDUNGSVERZEICHNIS	158
17.6	TABELLENVERZEICHNIS	161

6 Kurzbeschreibung

Die Diplomarbeit P@BS Connect wurde im Rahmen der standardisierten Reife- und Diplomprüfung 2017 an der HTL Perg für Informatik erstellt.

Problemstellung

Die Absolventen der HTL Perg können nach ihrem Abschluss sehr leicht den Kontakt zu ihrer ehemaligen Schule verlieren. Die Diplomarbeit soll diesem Phänomen entgegenwirken und eine stärkere Bindung zwischen Absolventen, lokalen Firmen und der Schule herstellen.

Viele Absolventen möchten sich über aktuelle Stellenangebote von Firmen des Absolventenverbands informieren. In der Vergangenheit wurden dazu Stellenausschreibungen per E-Mail vom Sekretariat der HTL Perg an alle Absolventen versandt. Das ist sehr zeitaufwändig und beansprucht Ressourcen, die an anderer Stelle dringender benötigt würden.

Ziel der Diplomarbeit ist es, für diese Problematik eine Lösung zu finden und zu implementieren.

Realisierung

Die Basis für die Implementierung stellt ein Linux-Server im Netzwerk der HTL Perg, auf dem eine MySQL-Datenbank und eine in Java implementierte REST-API gehostet werden. Diese Schnittstelle stellt die Daten für eine Webanwendung bereit. Die Webapplikation ist auf allen gängigen Browsern lauffähig und mittels Angular 2 umgesetzt. Durch das responsive Layout ist das System auch auf mobilen Geräten verwendbar.

Nebenbei wurde eine Webseite mit allgemeinen Informationen in Joomla realisiert, die ebenfalls auf dem vorhin genannten Linux-Server betrieben wird.

Ergebnis

Das Ergebnis ist aus Benutzersicht eine informelle Webseite mit allgemeinen Informationen zum Absolventenverband und eine Webapplikation, die als interner Bereich für Absolventen und Firmen mit einem Benutzerkonto erreichbar ist.

In dieser Applikation können die Nutzer ihre Stammdaten verwalten, News-Beiträge einsehen und kommentieren bzw. Events einsehen, kommentieren und diesen beitreten. Weiters können Firmen ihre Stelleninserate hochladen, wodurch diese für alle Absolventen sichtbar sind. Auf Wunsch erhalten die Benutzer eine Benachrichtigung per E-Mail bei neuen News-Beiträgen, Events oder Stellenangeboten. Außerdem ist es möglich, seinen Werdegang zu veröffentlichen, aus welchem dann ein PDF generiert wird, das als HTL-Perg-Erfolgsstory im Schulgebäude ausgestellt wird. Noch dazu können die Nutzer die Mitgliedsbeiträge für den Absolventenverband direkt online über den Zahlungspartner PayPal begleichen.

Als Administrator hat man die Möglichkeit, umfassende Verwaltungsaufgaben zu sämtlichen Funktionen durchzuführen.

7 Abstract

The diploma thesis P@BS Connect was created within the scope of the A-levels 2017 at the College of Engineering for informatics in Perg.

Problem

Graduates of the HTL Perg can lose contact to their former school quite easily. This thesis aims at counteracting this phenomenon and creating a stronger bond between alumni, local businesses and the school.

Many graduates would like to be informed about current job vacancies from firms of the alumni association. In the past, the school's office sent out emails to the graduates, which costs a lot of time and consumes resources that could be used better otherwise.

The diploma thesis aims at solving these problems.

Implementation

The basis for the implementation is a Linux-server in the network of the HTL Perg, which hosts a MySQL-database and a REST-API written in Java. This interface provides the data for a web application.

The web app is accessible with all popular browsers and is implemented using Angular 2. Due to the responsive layout, it is also usable on mobile devices.

Besides, a website containing general information was created using Joomla, which is also hosted on the previously mentioned Linux-server.

Result

The result from a user's perspective is a website with general information about the alumni association and a web application, which is accessible to graduates and companies with a user account as an internal area.

In this application, the users can manage their personal data, read and comment on news articles or see, comment on and participate in events. Furthermore, companies can upload job advertisements that are visible to the graduates. If they wish to, alumni receive notifications via email if a new news-entry, a new event or a new job vacancy is put online. Moreover, graduates can publish their career history, which is

put into a PDF file and gets displayed in the school building as a success story. Also, users can pay their membership fees to the alumni association directly online, using our payment partner PayPal.

8 Stakeholder

8.1 Team

8.1.1 Lukas Huber

Persönliche Daten

<i>Name</i>	Lukas Huber
<i>Geburtsdatum</i>	25. September 1997
<i>Adresse</i>	Henndorf 41
<i>Wohnort</i>	4372 St. Georgen am Walde
<i>E-Mail</i>	lukash1997@gmail.com



Abbildung 1: Lukas Huber
(Privat, 2017)

Referenzen

<i>Schulbildung</i>	2012 – 2017	HTL für Informatik in Perg
	2008 – 2012	Hauptschule St. Georgen am Walde
	2004 – 2008	Volksschule St. Georgen am Walde
<i>Praxis</i>	07/2015	Wimmer Medien GmbH & Co.KG., Linz (Ferialpraktikum)
	07/2014	Wacker Neuson Linz GmbH (Ferialpraktikum)

Kompetenzen

<i>Sprachen</i>	Deutsch (Muttersprache)
	Englisch (fließend in Wort und Schrift)
<i>IT</i>	Java, C#, iOS (Swift), Android, JavaScript, Angular 2, PHP, HTML, Oracle und MS SQL Server, MySQL, SAP, Grundkenntnisse ABAP, MS Office, Adobe Creative Suite

8.1.2 Christoph Penner

Persönliche Daten

<i>Name</i>	Christoph Penner
<i>Geburtsdatum</i>	30. Oktober 1997
<i>Adresse</i>	Poneggenstraße 26
<i>Wohnort</i>	4311 Schwertberg
<i>E-Mail</i>	penner.christoph@gmail.com



Abbildung 2: Christoph Penner (Privat, 2017)

Referenzen

<i>Schulbildung</i>	2012 – 2017	HTL für Informatik in Perg
	2008 – 2012	Europagymnasium Baumgartenberg
	2004 – 2008	Volksschule Schwerberg
<i>Praxis</i>	seit 10/2016	Storyclash GmbH, Linz (geringfügig beschäftigt)
	08/2016	Raiffeisen Software GmbH, Linz (Ferialpraktikum)
	seit 07/2015	Turtle Entertainment GmbH, Köln (ehrenamtlich)
	07/2015	RACON Software GmbH, Linz (Ferialpraktikum)
	07/2014	Raiffeisenlandesbank Oberösterreich AG, Linz (Ferialpraktikum)
	08/2013	Poschacher Baustoffe GmbH, Mauthausen (Ferialpraktikum)

Kompetenzen

<i>Sprachen</i>	Deutsch (Muttersprache)
	Englisch (fließend in Wort und Schrift)
	Französisch (Grundkenntnisse)
<i>IT</i>	Java, C#, iOS (Swift), Android, JavaScript, Angular 2, PHP, HTML, Oracle, MS SQL Server, MySQL, SAP, MS Office

8.1.3 Daniel Tüchler

Persönliche Daten

<i>Name</i>	Daniel Tüchler
<i>Geburtsdatum</i>	11. Juni 1997
<i>Adresse</i>	Mühlenweg 17
<i>Wohnort</i>	4311 Schwertberg
<i>E-Mail</i>	daniel_tuechler@hotmail.com



Abbildung 3: Daniel Tüchler (Privat, 2017)

Referenzen

<i>Schulbildung</i>	2012 – 2017	HTL für Informatik in Perg
	2008 – 2012	Europagymnasium Baumgartenberg
	2004 – 2008	Volksschule Schwerberg
	2003 – 2004	Vorschule Schwertberg
<i>Praxis</i>	06/2016	Unisoftwareplus GmbH, Perg (Ferialpraktikum)
	07/2015	SYSco GmbH, Schwertberg (Ferialpraktikum)
	07/2014	Herbert Handlos GmbH, Tragwein (Ferialpraktikum)

Kompetenzen

<i>Sprachen</i>	Deutsch (Muttersprache)
	Englisch (fließend in Wort und Schrift)
	Französisch (Grundkenntnisse)
<i>IT</i>	Java, C#, iOS (Swift), Android, JavaScript, Angular 2, PHP, HTML, SQL, Oracle, MS SQL Server, MySQL, SAP, MS Office

8.2 Betreuung

8.2.1 Direktor Prof. Dipl.-Ing. Christian Reisinger

Kontakt

c.reisinger@htl-perg.ac.at

Unsere Diplomarbeit wird von unserem Direktor Prof. Reisinger betreut, der seit dem dritten Jahrgang unseren Unterricht in Systemplanung und Projektentwicklung leitet.

Als Obmann-Stellvertreter des Absolventenverbandes und durch die Hintergrundinformationen aus der Position des Schulleiters konnten wir auf Tipps und Unterstützung bei der Anforderungsanalyse bauen.

Durch die Funktion als Lehrkraft für Projektentwicklung und die langjährige einschlägige Berufspraxis in diesem Bereich durften wir auf die Erfahrung und das Knowhow in Sachen Projektmanagement von Herrn. Prof. Reisinger zurückgreifen.



Abbildung 4: Dipl.-Ing. Christian Reisinger (HTL Perg, 2017)

8.3 Lehranstalt

8.3.1 HTL Perg, Höhere Abteilung für Informatik

Kontakt

HTBLA Perg

Machlandstraße 48

4320 Perg

Tel. 0 72 62 / 539 26

Fax. 0 72 62 / 539 26 – 6

office@htl-perg.ac.at

www.htl-perg.ac.at

Schulkennzahl: 411457



Abbildung 5: HTL Perg (FCP, 2017)

„Der Schwerpunkt der Ausbildung liegt im Bereich der Softwareentwicklung, wobei betriebswirtschaftliche Aspekte besondere Berücksichtigung finden.

Gleichzeitig wird auch eine fundierte Allgemeinbildung geboten.

Diese Kombination bedeutet eine umfangreiche, praxisnahe Ausbildung, die später im Beruf direkt umgesetzt werden kann, aber auch die Möglichkeit zu Universitäts- bzw. Fachhochschulstudium bietet.

Das rege Interesse der Wirtschaft an dieser Fachrichtung wird durch ständige Kooperationen mit Unternehmen in Form von Projekten unterstrichen, die von Schülern im Rahmen des Gegenstandes Projektentwicklung oder als Diplomarbeit realisiert werden.“ (HTL Perg, 2017)

8.4 Auftraggeber

8.4.1 P@BS – EDV Absolventen und Förderer der HTL – Perg

Kontakt

Machlandstraße 48

4320 Perg

pabs@htl-perg.ac.at

„Der Verein P@BS – EDV Absolventen und Förderer der HTL – Perg hat den Zweck der Bildung und Aufrechterhaltung einer Informationsplattform zwischen Schule, Firmen, Absolventen und sonstigen Interessenten. Der Praxisbezug der Ausbildung soll so auf dem neuesten Stand erhalten bzw. erhöht werden.

Zudem ist die Förderung der Ausbildung an der HTL Perg durch Geld- und Sachwerte an die Schule bzw. die Schüler Inhalt der Tätigkeit.

Die Mittel ergeben sich auf der einen Seite durch Mitgliedsbeiträge, Erträge aus Veranstaltungen und Spenden und auf der anderen Seite durch Vorträge, Diskussionsveranstaltungen, Firmenpräsentationen, Präsentationen von Schulprojekten, Firmenbesuche und Informationsveranstaltungen zur Präsentation des Vereins.

Mit der Mitgliedschaft sollte Unternehmen die Möglichkeit gegeben werden, Projekte mit der HTL Perg in Form von Jahres- oder Maturathemen zu realisieren. Weiters steht der Verein unterstützend bei der Vermittlung von Absolventen bzw. Adressen von Absolventen zur Seite. In der Folge sollten sich gemeinsame Präsentationen von abgeschlossenen Themen, die in Kooperation umgesetzt wurden, ergeben.“
(P@BS, 2017)

8.4.2 Obmann Prof. Dipl.-Ing. Gerhard Zweimüller

Kontakt

g.zweimueller@htl-perg.ac.at

Unser Ansprechpartner im Absolventenverband ist dessen Obmann Prof. Zweimüller, der uns seit dem zweiten Jahrgang im Fach Netzwerksysteme und verteilte Systeme begleitet.

Wir durften die Anforderung für das Projekt anhand seiner Vorgaben definieren und sein Feedback in das Produkt einfließen lassen.



*Abbildung 6: Dipl.-Ing. Gerhard Zweimüller
(HTL Perg, 2017)*

9 Einleitung

9.1 Ist-Zustand

Die Verwaltung des Absolventenverbandes der HTL Perg ist momentan sehr zeitaufwändig und nimmt Ressourcen in Anspruch, die an anderer Stelle dringender benötigt würden. Beispielsweise werden Stelleninserate, welche der HTL Perg per Mail von Mitgliedsfirmen des Absolventenverbandes zugesendet werden, über einen E-Mail-Verteiler an die Absolventen weitergeleitet. Dadurch lastet auf dem Sekretariat der HTL Perg eine unnötige Zusatzaufgabe.

Weiters werden die sogenannten HTL-Perg-Erfolgsstories manuell erstellt. Absolventen, die ihren Erfolg mit ihrer ehemaligen Schule teilen wollen, schicken die benötigten Daten an das Sekretariat. Diese Daten werden dann per Hand in eine Microsoft Publisher Vorlage eingefügt, als PDF-Datei exportiert, auf DIN A3 ausgedruckt und in der HTL Perg ausgestellt.

Zusätzlich gestaltet sich die Verwaltung der Finanzen des Absolventenverbandes schwierig. Momentan können Absolventen ihren Mitgliedsbeitrag nur per Überweisung begleichen.

Ein weiterer wichtiger Punkt betrifft die Verbindung der Absolventen zu ihrer ehemaligen Schule. Der Kontakt kann bereits nach kurzer Zeit leicht abbrechen und man wird nicht ausreichend über Neuigkeiten oder anstehende Veranstaltungen informiert.

Darüber hinaus ist die bestehende Homepage des Absolventenverbandes veraltet. Hierbei spielen nicht nur die veralteten Informationen eine Rolle, sondern durch die Verwendung veralteter Technologien besteht ein erhöhtes Sicherheitsrisiko.



Abbildung 7: Startseite der alten Homepage

9.2 Motivation

Diese Diplomarbeit fällt in den Bereich des Web-Engineering. Unter Web-Engineering fällt die Entwicklung von Webanwendungen, Webservices oder anderer komplexer Websites (vgl. wikipedia.org, 2017). Dieser Bereich der Softwareentwicklung wird immer wichtiger. In Form dieser Arbeit haben wir die Möglichkeit, uns intensiver damit auseinanderzusetzen. Ebenso können wir bereits Erlerntes aus dem Unterricht in der Praxis anwenden und dadurch Erfahrungen sammeln, die im späteren Berufsleben mit hoher Wahrscheinlichkeit sehr relevant werden.

Die Zusammenarbeit mit einem Auftraggeber und die daraus folgenden Herausforderungen bringen zusätzliche Erfahrung für das Projektteam.

9.3 Ziel der Arbeit

Das Ziel der Arbeit ist die Erstellung eines Webportals auf Basis aktueller Technologien. Durch den neuen Webauftritt soll die Präsenz des Absolventenverbandes und die daraus resultierende Werbewirkung erhöht werden. Der Verwaltungsaufwand des Sekretariats der HTL Perg soll auf ein Minimum reduziert werden. Ebenso soll eine engere Bindung zwischen Absolventen, Firmen und der Schule selbst erzielt werden.

9.3.1 Geschäftsziele

- Erhöhung der Präsenz des Absolventenverbandes
- Werbewirkung für die Schule und Firmen
- Informationsbereitstellung für Absolventen
- Stellenangebote der Firmen direkt an die Absolventen tragen, um beiden Parteien wirtschaftliche Vorteile zu bieten
- Einfachere Verwaltung der Finanzen durch den Verband (Online-Zahlungsmöglichkeit der Mitgliedsbeiträge)
- Reduktion der Aufwände für P@BS-Tätigkeiten im HTL-Sekretariat

9.4 Funktionsbereiche des Webportals

Im Folgenden werden die geforderten Funktionsbereiche des Systems näher beschrieben.

9.4.1 Rollensystem

Das System soll je nach der Art des Benutzers verschiedene Funktionen bieten, daher werden unterschiedliche Rollen benötigt.

- *Absolvent*
 - Dieser Rolle werden alle Absolventen der HTL Perg zugeordnet.
- *Firma*
 - Unter der Rolle Firma sind alle Mitgliedsfirmen des P@BS zu finden.
- *Administrator*
 - Der Administrator soll das System verwalten können und verfügt über spezielle Rechte zur Administration der Anwendung.

9.4.2 Informelle Homepage

Die alte Homepage des Absolventenverbandes soll neu und ansprechender gestaltet werden. Die Informationen dafür werden von der bestehenden Website übernommen.

Auf dem informellen Teil des Webportals sollen folgende Dinge anspruchsvoll dargestellt werden:

- Beschreibung des P@BS
- Das Team des P@BS
- Kontaktdaten
- Öffentliche Neuigkeiten

9.4.3 Stammdatenverwaltung

Alle Benutzer des Systems sollen die Möglichkeit haben, ihre persönlichen Stammdaten zu bearbeiten. Diese umfassen Adresse und E-Mail, sowie bei Absolventen den Vor- und Nachnamen beziehungsweise bei Unternehmen die Firmenbezeichnung und den Ansprechpartner.

9.4.4 Newsverwaltung

Der Administrator soll die Möglichkeit haben, News-Beiträge, welche nur den Absolventen und Firmen vorbehalten sind, erstellen und löschen zu können. Die Benutzer können dann die Neuigkeiten lesen und kommentieren.

9.4.5 Veranstaltungsverwaltung

Dem Administrator soll es möglich sein, Veranstaltungen zu erstellen. Absolventen beziehungsweise Firmen können diese Veranstaltungen betrachten und ihre eventuelle Teilnahme bestätigen. Auch hier können Kommentare verfasst werden.

9.4.6 Stellenangebotsverwaltung

Mitgliedsfirmen sollen selbstständig die Möglichkeit haben, ihre Stelleninserate in Form einer PDF-Datei hochladen zu können. Die Absolventen haben die Option, diese Stellenangebote anzusehen.

9.4.7 Mitgliedsbeitragsverwaltung

Alle Mitglieder sollen die Option haben, ihren Mitgliedsbeitrag oder generelle Forderungen des Absolventenverbandes online mittels gängigen Zahlungsmitteln bezahlen zu können.

9.4.8 HTL-Erfolgsgeschichten

Auf Absolventenwunsch soll es möglich sein, seinen beruflichen Erfolg mit Besuchern der HTL Perg zu teilen. Man gibt hierzu seinen Werdegang und einige zusätzliche Daten ein und stimmt der Veröffentlichung zu. Danach werden diese Erfolgsgeschichten vom System auf eine A3-Seite zusammengefasst und vom Administrator ausgedruckt und in der HTL Perg ausgestellt.

9.4.9 Benachrichtigungen

Auf Wunsch der Absolventen soll es die Möglichkeit geben, automatisch bei neuen Stelleninseraten und / oder bei neuen News-Beiträgen bzw. Veranstaltungen per Mail darüber informiert zu werden.

10 Grundlagen

Das nachfolgende Kapitel beschäftigt sich mit der Beschreibung einiger Grundlagen wie der verwendeten Technologien und Entwicklungssysteme.

10.1 Verwendete Technologien

10.1.1 Ubuntu



Abbildung 8: Ubuntu-Logo ([wikipedia.org](https://www.wikipedia.org), 2017)

Ubuntu ist eine kostenfreie, auf Debian basierende Linux-Distribution. Das beliebte Betriebssystem wird von Canonical gesponsert und auch von Community-Mitgliedern weiterentwickelt. (vgl. [wikipedia.org](https://www.wikipedia.org), 2017)

Wir verwenden im Projekt die Ubuntu Server-Version 16.04, auf der unsere Joomla-Website, die MySQL-Datenbank und die REST-API laufen.

10.1.2 SQL (Structured Query Language)

SQL ist eine Datenbankabfragesprache und dient zur Definition, zum Bearbeiten und zum Abfragen von Datenstrukturen in relationalen Datenbanken. Die SQL-Syntax ist sehr trivial aufgebaut und die Sprache ist semantisch an die englische Sprache angelehnt, woraus die Einfachheit von SQL resultiert. Die Structured Query Language kann in Verbindung mit beinahe allen gängigen Datenbankmanagementsystemen verwendet werden. Auf SQL aufbauend gibt es Erweiterungen wie beispielsweise Transact-SQL beim Microsoft SQL Server. (vgl. [wikipedia.org](https://www.wikipedia.org), 2017)

Mithilfe von SQL wird in unserer Diplomarbeit mit der MySQL-Datenbank kommuniziert.

10.1.3 MySQL

MySQL repräsentiert eines der am weitesten verbreiteten Datenbankmanagementsysteme. MySQL ist als Open-Source-Software und als kommerzielle Version erhältlich. Bei dieser Diplomarbeit wird die kostenlose Version verwendet. Das DBMS ist vor allem im Bereich der Content-Management-Systeme weit verbreitet und wird auch bei uns als Basis für Joomla verwendet. Es gehört zur Kategorie der relationalen Datenbanken und arbeitet als Client-Server-System. Dies bedeutet, die Datenbank spiegelt den Server wider und nimmt Anfragen des Clients, in diesem Fall der Webanwendung bzw. der REST-API, entgegen und übersetzt diese in ausführbaren Code. Nach der Ausführung dieses Codes leitet die Datenbank die jeweiligen Informationen an den Client zurück. (vgl. wikipedia.org, 2017)



Abbildung 9: MySQL-Logo (wikipedia.org, 2017)

10.1.4 Java-Technologie

Java ist eine ursprünglich von Sun Microsystems entwickelte Spezifikationen-Sammlung, die einerseits die Programmiersprache Java, andererseits verschiedene Laufzeitumgebungen für Computerprogramme beschreibt. Mittlerweile wurde Sun aufgekauft und Java wird von Oracle weiterentwickelt.

Zur Technologie gehören die Programmiersprache Java, das Java Development Kit (ein Entwicklungswerkzeug mit einem Compiler und einigen Bibliotheken) und die Java-Laufzeitumgebung (eine standardisierte Software-Plattform zur Ausführung der entwickelten Programme mithilfe der Java Virtual Machine). Ein Vorteil von Java ist die Ausführbarkeit desselben Codes auf sehr vielen Plattformen. Man ist nicht an ein spezifisches Betriebssystem gebunden. (vgl. wikipedia.org, 2017)



Abbildung 10: Java-Logo (wikipedia.org, 2017)

10.1.5 Java (Programmiersprache)

Die Programmiersprache Java zählt zu den objektorientierten Programmiersprachen. Sie ist plattformunabhängig, daher können alle in Java programmierten Anwendungen auf sämtlichen Plattformen ausgeführt werden, falls diese die Technologie Java installiert haben. Java unterstützt beinahe alle Techniken der Objektorientierung wie die Vererbung, Überladung aber auch Polymorphie und eignet sich besonders gut für die Entwicklung von Webanwendungen. (vgl. wikipedia.org, 2017)

Die REST-API in dieser Diplomarbeit ist in Java geschrieben.

10.1.6 WildFly

WildFly ist ein plattformunabhängiger, in Java geschriebener Open-Source-Applikationsserver für Java Enterprise Anwendungen nach dem Java-EE-Standard.

Die Vorzüge liegen in der einfachen Verwendbarkeit durch das Konfigurationsinterface als Webanwendung, die schonende Speicherverwendung und die schnelle Startzeit.

Früher war WildFly unter dem Namen JBoss bekannt. Als ORM-Mapper wird Hibernate eingesetzt.

(vgl. Wildfly, 2017)



Abbildung 11: WildFly-Logo (Visual Studio Marketplace, 2017)

In der Diplomarbeit wird der WildFly Applikationsserver in der Version 10 verwendet, um die Java-REST-API auszuführen und online zur Verfügung zu stellen.

10.1.7 Java Persistence API

Die Java Persistence API, kurz JPA, ist eine Ansammlung von Methoden und Klassen, welche verwendet werden, um Objekte für die persistente Speicherung in eine Datenbank zu übertragen und zuzuordnen.

Die Technologie wird zum Verbinden von Objekten und Datenbanken in Applikationen verwendet. Somit können ganz normale Java-Objekte (POJOs – Plain Old Java Objects) sehr einfach und effizient persistiert werden.

(vgl. tutorialspoint, 2017)

Bestandteil	Beschreibung
Entity	Objekt, welches persistiert, im Programm verwaltet und in der Datenbank gespeichert wird; Ein Objekt stellt einen Datenbankeintrag dar
EntityManager	Interface, welches als Aufgabe die Verwaltung der Operationen auf die Datenbankobjekte hat
EntityManagerFactory	Erstellt und verwaltet mehrere EntityManager-Instanzen
EntityTransaction	Interface, welches die Verwaltung der Transaktionen des EntityManagers übernimmt
Persistence	Klasse, welche zum Erzeugen von EntityManagerFactory-Instanzen verwendet wird
Query	Interface, welches Abfragen auf die Datenbank ermöglicht

Tabelle 1: JPA-Bestandteile (vgl. tutorialspoint, 2017)

JPA wird am Applikationsserver für die Kommunikation mit der Datenbank in dieser Diplomarbeit verwendet.

10.1.8 HTML

HTML steht für Hypertext Markup Language und beschreibt den strukturellen Aufbau einer Website im Internet. Es stellt damit die Grundlage für das World Wide Web dar.

HTML beschreibt hierbei nur die Struktur und trifft keine Aussagen über das Aussehen der späteren Seite. Die textbasierte Sprache wird vom World Wide Web Consortium (W3C) weiterentwickelt.

In der aktuellen Version HTML 5 wurden zahlreiche neue Funktionalitäten wie die direkte Einbettung von Audio- und Videoelementen eingeführt. (vgl. selfhtml, 2017)

Mittels HTML wird in der Diplomarbeit der Seitenaufbau der Webanwendung definiert.



Abbildung 12: HTML-Logo
(wikipedia.org, 2017)

10.1.9 CSS

Die Formatierungssprache CSS steht für Cascading Style Sheets und beschreibt das Aussehen einer Website. Es wird somit in der Regel in Verbindung mit HTML eingesetzt. Im HTML-Code werden CSS-Klassen angegeben, welche das Design der Elemente beschreiben.

Das Erscheinungsbild der Webanwendung in der Diplomarbeit wird über ein Template mithilfe von CSS definiert.



Abbildung 13: CSS-Logo
(toptal.com, 2017)

10.1.10 PHP

PHP ist eine Skriptsprache, die zur Erstellung dynamischer Webseiten oder Webanwendungen verwendet wird. PHP bildet die Grundlage für Joomla und ist daher die Basis für den öffentlichen, informellen Teil des Webportals.



Abbildung 14: PHP-Logo
(qph.ec.quoracdn.net, 2017)

10.1.11 JavaScript (JS)

JavaScript (beziehungsweise hier eigentlich ECMAScript) ist eine dynamisch typisierte, objektorientierte, aber klassenlose Skriptsprache für die Webentwicklung. Mit JS können dynamische Webinhalte realisiert werden. (vgl. wikipedia.org, 2017)

JavaScript bildet die Basis für die Verwendung des von uns eingesetzten Frameworks Angular 2 sowie der von uns eingesetzten Programmiersprache TypeScript.



Abbildung 15: JavaScript-Logo (javacodegeeks.com, 2017)

10.1.12 Angular 2

Angular 2 ist ein umfangreiches, clientseitiges JavaScript-Framework und wird in der Regel für Anwendungen (oftmals Single-Page-Applikationen) nach einem Model-View-ViewModel-Muster verwendet.

Die Open-Source-Lösung wird von Google entwickelt, wobei Angular 2 nicht rückwärtskompatibel zum ziemlich unterschiedlichen AngularJS (Version 1) ist. (vgl. wikipedia.org, 2017)



Abbildung 16: Angular 2-Logo (mbtcheck.com, 2017)

Die Webanwendung für diese Diplomarbeit wird mithilfe von Angular 2 entwickelt.

10.1.13 TypeScript

TypeScript ist eine Programmiersprache von Microsoft, die auf den Vorschlägen zum ECMAScript-6-Standard aufbaut. Da jeder JavaScript-Code auch gültiger TypeScript-Code ist, können JS-Bibliotheken bzw. Frameworks wie jQuery oder Angular 2 verwendet werden.



Abbildung 17: TypeScript-Logo (wikipedia.org, 2017)

Die Webanwendung der Diplomarbeit wird mit TypeScript in Kombination mit Angular 2 realisiert.

10.1.14 Git

Git ist eine Open-Source-Lösung zur verteilten Versionsverwaltung, welche sehr gerne für Software-Projekte eingesetzt wird. In einem Git-Repository werden alle Teile der Software zentral abgespeichert. Ein Teammitglied kann die gespeicherten Dateien klonen und so lokal damit weiterarbeiten. Die Änderungen können später zurück an den Server gesandt werden, damit sie für alle Mitglieder des Entwicklungsteams bereitstehen. Konflikte bei der Zusammenführung können relativ einfach beseitigt werden und man hat immer die Möglichkeit, auf einen früheren Zeitpunkt und dem damit verbundenen Versionsstand des Codes zurückzukehren.



Abbildung 18: Git-Logo
(git-for-windows.github.io,
2017)

In der Diplomarbeit wird mit zwei Git-Repositories gearbeitet, die bei Atlassian Bitbucket (bitbucket.org) gehostet werden.

10.1.15 Joomla

Joomla ist ein Content-Management-System, kurz CMS, das zur Erstellung von Webseiten dient. Es ist in PHP 5 geschrieben und verwendet als Datenbank MySQL. (vgl. wikipedia.org, 2017)



Abbildung 19: Joomla-Logo (wikipedia.org, 2017)

Unter dem Begriff Content-Management System versteht man eine Software zur gemeinschaftlichen Erstellung, Bearbeitung und Organisation von Website-Inhalten. (vgl. wikipedia.org, 2017)

Die Startseite des Webportals mit allgemeinen Informationen zum Absolventenverband ist in Joomla realisiert.

10.2 Verwendete Entwicklungssysteme

10.2.1 MySQL Workbench

MySQL Workbench ist ein Tool zur visuellen Erstellung und Verwaltung von Datenbanken von Oracle. Es können Datenbankmodelle graphisch erstellt und daraus durch das sogenannte „Forward-Engineering“ direkt einsatzbereite MySQL-Datenbanken generiert werden. Ebenfalls können direkt Testdaten ohne das manuelle Schreiben von SQL-Befehlen eingefügt werden.

Über eine Datenbankverbindung können die bestehenden Daten in einer Datenbank angesehen, manipuliert und gelöscht werden. Außerdem können die aktuelle Performance sowie einige Leistungskennzahlen zur Datenbank direkt in der Anwendung analysiert werden.

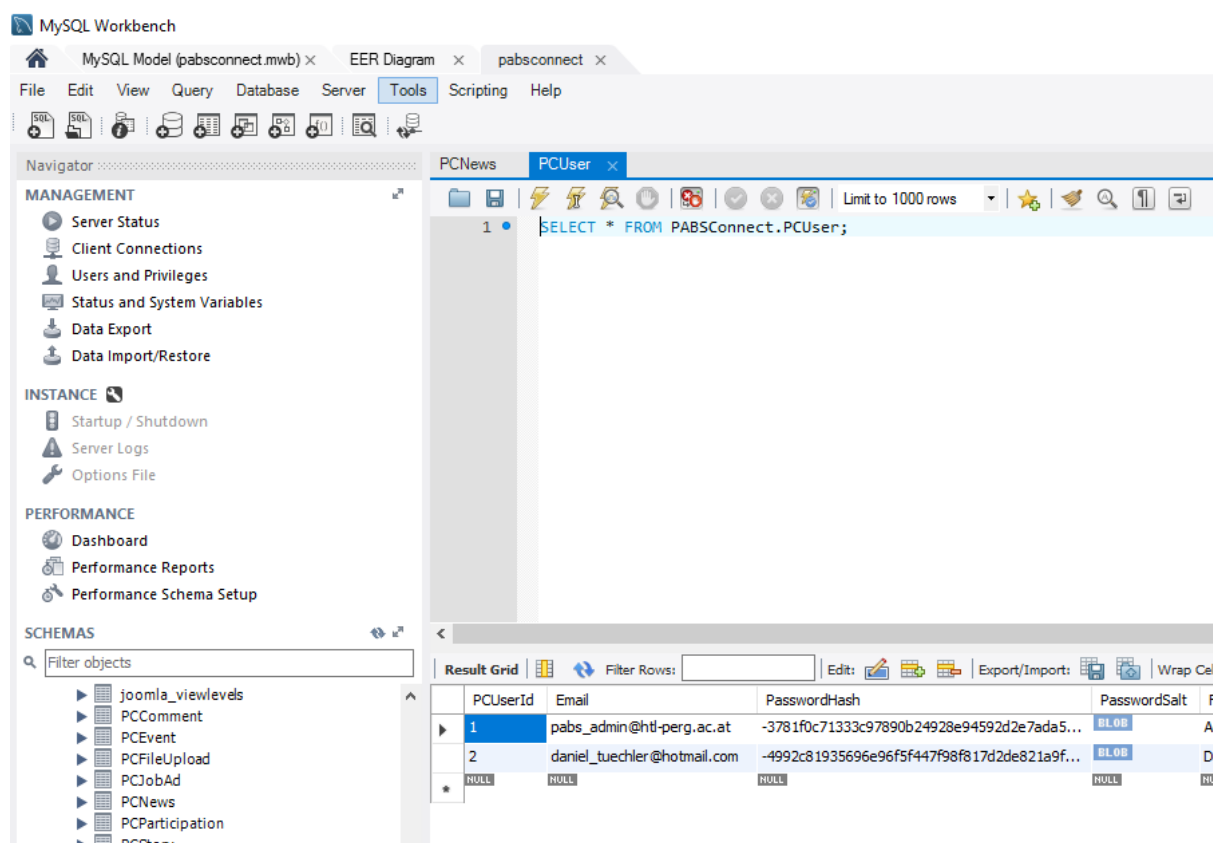


Abbildung 20: Ausschnitt von MySQL Workbench

In der Diplomarbeit wird MySQL Workbench in der Version 6.2 zur Erstellung des Datenbankmodells und der Generierung der daraus resultierenden Datenbank sowie zur Kontrolle von Nutzdaten im Testbetrieb eingesetzt.

10.2.2 Eclipse

Eclipse ist eine Open-Source-Entwicklungsumgebung (IDE – Integrated Development Environment) für diverse Sprachen, hauptsächlich wird sie für Java und Java EE verwendet. Sie wird von der Eclipse Foundation weiterentwickelt und gilt als sehr beliebte Entwicklungsumgebung für Java-Software-Projekte. (vgl. wikipedia.org, 2017)

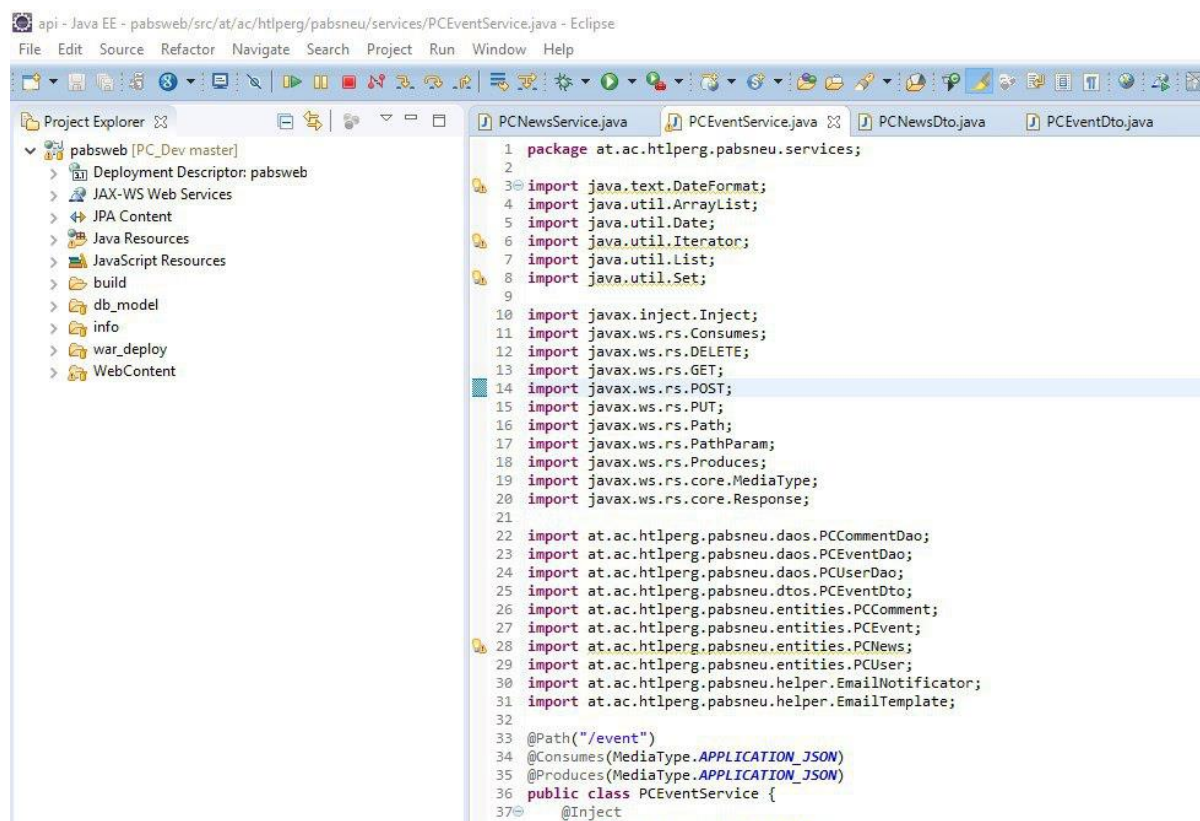


Abbildung 21: Ausschnitt der Eclipse-IDE

In der Diplomarbeit wird Eclipse in der Version 4.6.0 zur Entwicklung der REST-API in Java EE verwendet.

10.2.3 JetBrains WebStorm

WebStorm ist eine Entwicklungsumgebung für die Programmiersprache JavaScript und wird von JetBrains entwickelt und vertrieben. Neben reinem JavaScript werden beispielsweise auch HTML5, Node.js, AngularJS, TypeScript oder Bootstrap unterstützt. (vgl. wikipedia.org, 2017)

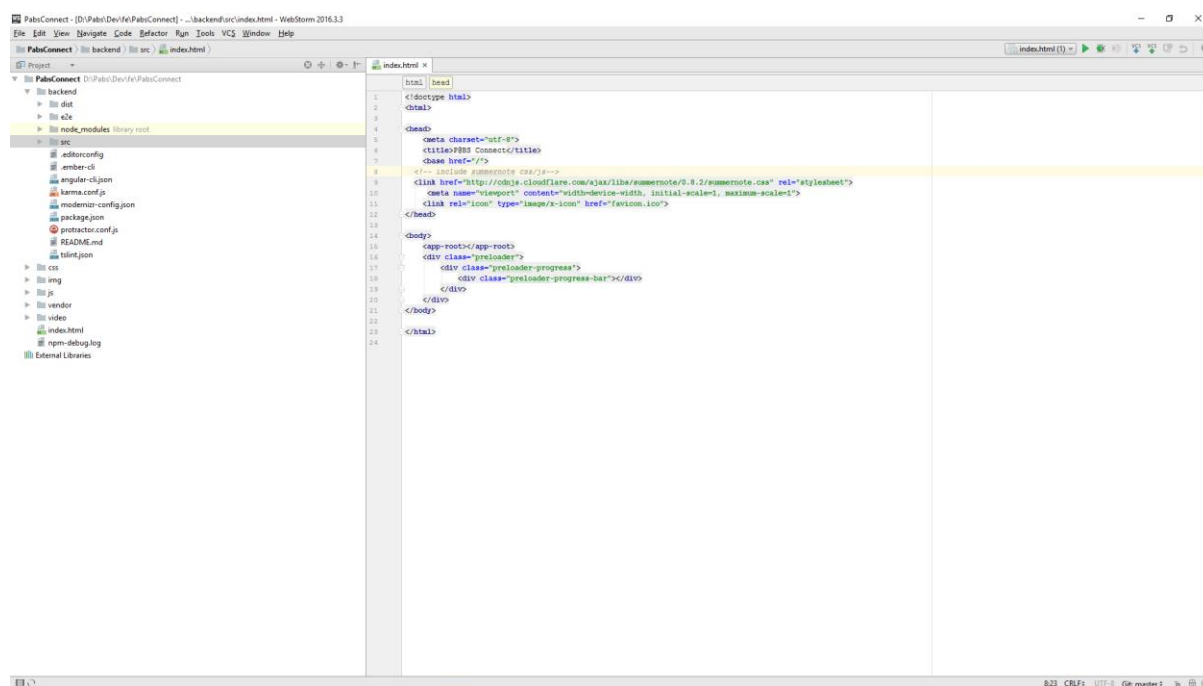


Abbildung 22: JetBrains WebStorm

WebStorm wird in dieser Diplomarbeit in der Version 2016.3.3 für die Entwicklung der Webanwendung in TypeScript mittels Angular 2 verwendet.

10.3 Verwendete Bibliotheken und Plug-Ins

10.3.1 iText

iText ist eine kostenlose Programmbibliothek zur Erzeugung und Bearbeitung von PDF-Dateien mithilfe der Programmiersprache Java. (vgl. wikipedia.org, 2017)



Abbildung 23: iText-Logo (PDF association, 2017)

10.3.2 JA Builder

Der JA Builder von Joomla!Art ist ein Website-Baukasten Plug-In für Joomla. Dadurch ist es möglich, sehr schnell Webseiten zu erstellen, welche im späteren Verlauf einfach bearbeitet und verändert werden können.

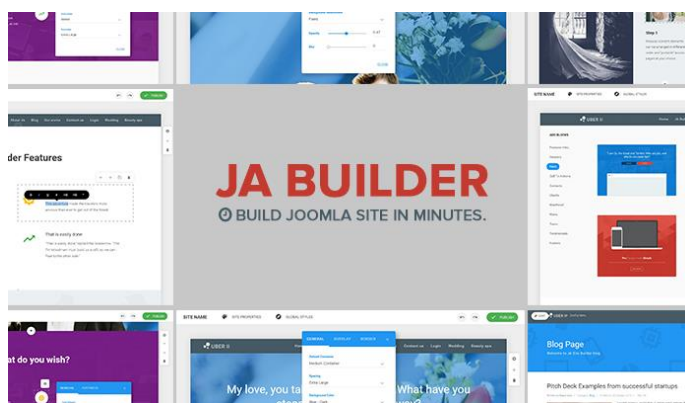


Abbildung 24: JA Builder (Joomla!Art, 2017)

Für den informellen Teil des Webportals wurde in dieser Diplomarbeit auf den JA Builder zurückgegriffen.

10.3.3 Akeeba Backup und Kickstart

Akeeba Backup ist eine kostenlose Joomla-Komponente, die es ermöglicht, seine Joomla-Installation zu sichern. Sollte ein Problem auftreten, so kann ein früherer Stand wiederhergestellt werden.

Akeeba Kickstart dient zum Einspielen von Akeeba Backup-Sicherungen, falls die Administrationsoberfläche von Joomla nicht mehr zugänglich ist.



Akeeba
Backup

Abbildung 25: Logo Akeeba Backup (jomboom.com, 2017)

10.4 Verwendete Architekturen

10.4.1 DAO (Data Access Object)

Data Access Object (Datenzugriffsobjekt) ist ein Entwurfsmuster zum gekapselten Zugriff auf Datenquellen. Somit kann die unterliegende Datenquelle (z.B. Datenbank) ausgetauscht werden, ohne Änderungen in der aufrufenden Business-Schicht einer Anwendung durchführen zu müssen. (vgl. wikipedia.org, 2017)

Mit diesem Entwurfsmuster wird in dieser Diplomarbeit eine logische Trennung der Datenschicht und der Geschäftslogik realisiert.

10.4.2 REST (Representational State Transfer)

REST ist ein Programmierparadigma zur Erstellung von Webservices. REST hat einen Architekturstil zum Ziel, der den Anforderungen eines modernen World Wide Web gerecht wird. REST beschreibt eine Kommunikation über das sehr weit verbreitete http-Protokoll. In REST hat jede Ressource eine eindeutige Identifikation, über die sie angesprochen werden kann. Jede Kommunikation zwischen Client und Server ist zustandslos, das heißt eine Anfrage ist in sich geschlossen und enthält sämtliche notwendigen Informationen. Eine Verbindung zu vorherigen oder künftigen Nachrichten ist nicht gegeben.

Meist werden die Daten im JSON- oder XML-Format übertragen. Eine Alternative wäre das SOAP-Verfahren, welches jedoch einen großen Overhead mit sich bringt.

In der Diplomarbeit wird das REST-Paradigma zur Kommunikation zwischen dem Webbrowser als Client der Webanwendung und der API für den Datenaustausch verwendet.

HEADER: ALGORITHM & TOKEN TYPE

```
{  
  "alg": "HS256"  
}
```

PAYLOAD: DATA

```
{  
  "jti": "1",  
  "iat": 1490988785,  
  "sub": "pabs_admin@htl-perg.ac.at",  
  "iss": "pabs.htl-perg.ac.at",  
  "Role": "Admin",  
  "exp": 1490992385  
}
```

Abbildung 27: dekodierter JWT

Informationen wie der verwendete Algorithmus, das Ausstellungsdatum, der Aussteller des Tokens, der verwendete Benutzername (E-Mail), die zugewiesene Rolle oder das Ablaufdatum des JWT sind auslesbar. (vgl. jwt.io, 2017)

Die möglichen Funktionen und Einsatzgebiete sind weitreichend – in dieser Diplomarbeit werden JWTs dazu verwendet, die Benutzer am Server für jegliche Anfragen nach dem Einloggen zu authentifizieren.

10.5 Verwendete Templates

10.5.1 Angle Bootstrap Admin Template

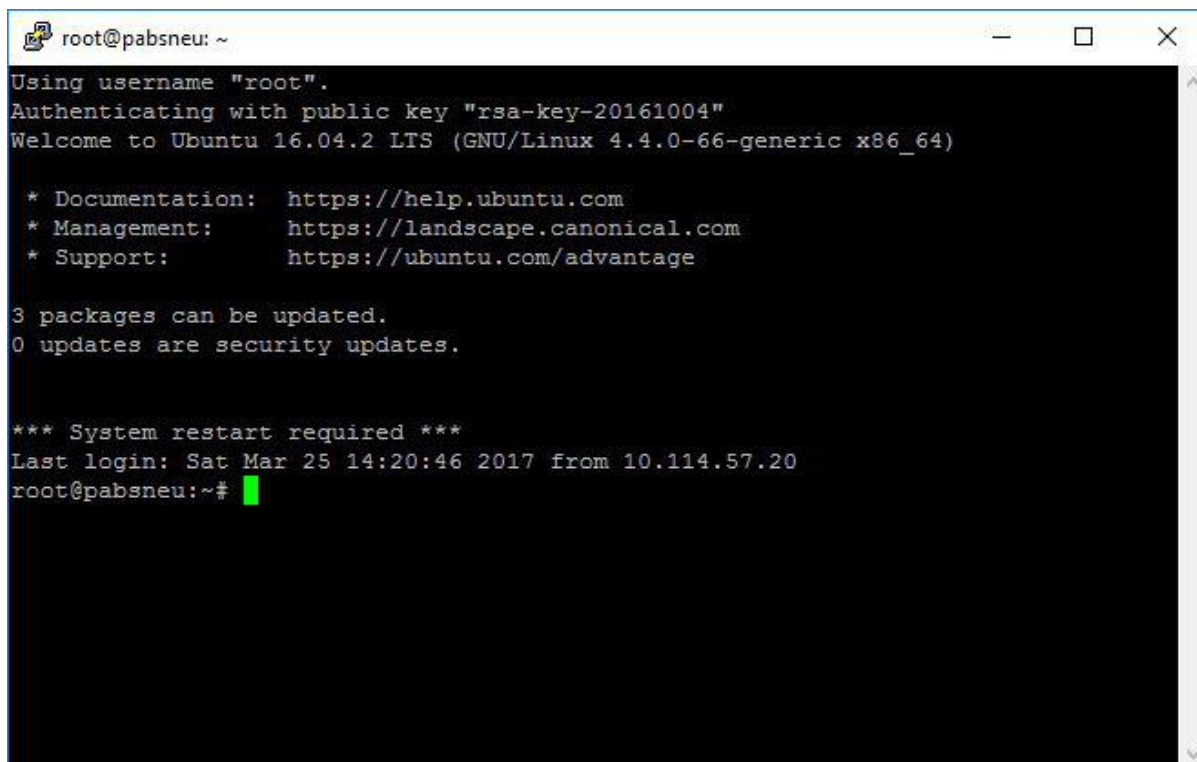
Angle ist ein kostenpflichtiges Bootstrap Admin Template, welches für diverse Programmiersprachen und Frameworks eingesetzt werden kann. Neben einer Landing Page als mögliche Startseite steht eine Vielzahl an Komponenten zur Verwendung in einer Webanwendung zur Verfügung. Das Template ist responsive, kann also auch auf mobilen Geräten ohne markante Qualitätseinbußen verwendet werden.

In der Diplomarbeit wird das Angular 2 Template als Basis für die visuelle Gestaltung der Webanwendung verwendet.

10.6 Sonstige verwendete Software

10.6.1 PuTTY

PuTTY ist ein graphischer, kostenloser SSH und telnet-Client. Meist wird damit von einem Windows-Rechner auf einen Linux-Server zugegriffen. (vgl. ubuntuusers, 2017)



```
root@pabsneu: ~
Using username "root".
Authenticating with public key "rsa-key-20161004"
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-66-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

3 packages can be updated.
0 updates are security updates.

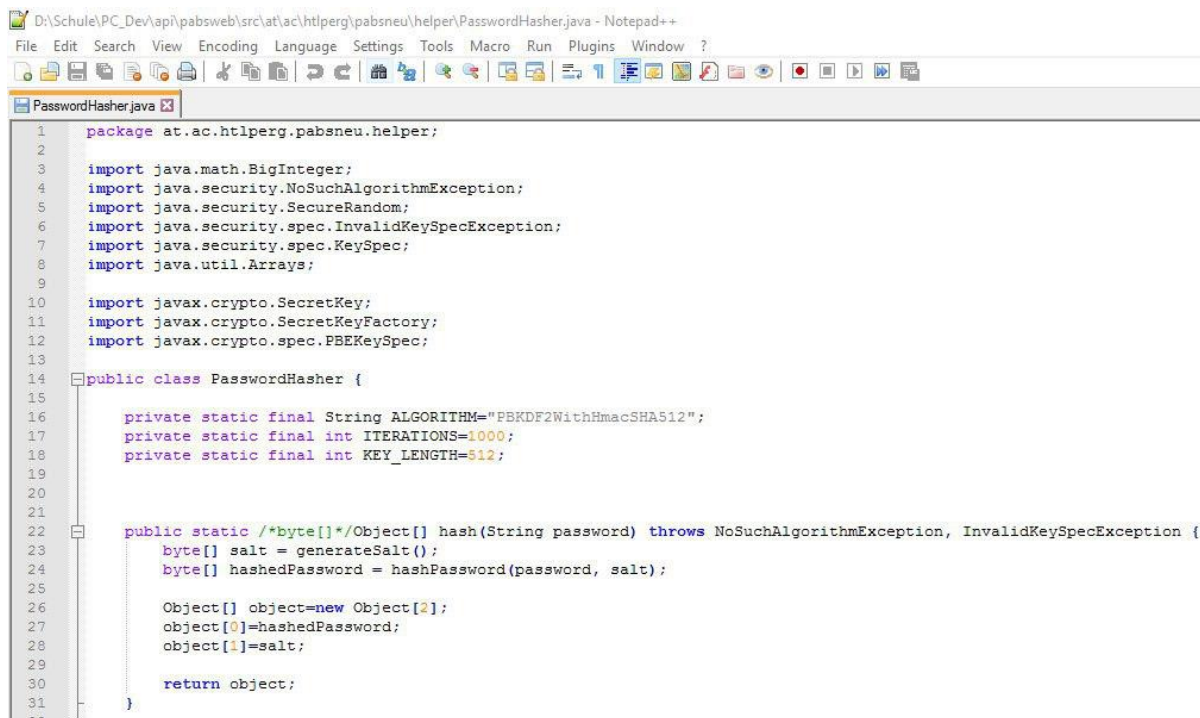
*** System restart required ***
Last login: Sat Mar 25 14:20:46 2017 from 10.114.57.20
root@pabsneu:~#
```

Abbildung 28: PuTTY-Terminal

Mittels PuTTY in der Version 0.65 wird in der Diplomarbeit auf den Linux-Server zugegriffen, auf dem Joomla, die MySQL-Datenbank sowie die REST-API laufen, um Software zu installieren, Konfigurationen durchzuführen sowie Log-Dateien auszulesen.

10.6.2 Notepad++

Notepad++ ist ein in C++ geschriebener, kostenloser und leichtgewichtiger Source-Code-Editor, der zahlreiche Sprachen unterstützt und somit eine Hervorhebung der jeweiligen Syntax unterstützt. Gegenüber dem Standard-Windows-Notepad ist es weiters möglich, weitaus größere Dateien viel schneller zu laden und zu editieren. (vgl. Notepad++, 2017)



```
1 package at.ac.htlperg.pabsneu.helper;
2
3 import java.math.BigInteger;
4 import java.security.NoSuchAlgorithmException;
5 import java.security.SecureRandom;
6 import java.security.spec.InvalidKeySpecException;
7 import java.security.spec.KeySpec;
8 import java.util.Arrays;
9
10 import javax.crypto.SecretKey;
11 import javax.crypto.SecretKeyFactory;
12 import javax.crypto.spec.PBEKeySpec;
13
14 public class PasswordHasher {
15
16     private static final String ALGORITHM="PBKDF2WithHmacSHA512";
17     private static final int ITERATIONS=1000;
18     private static final int KEY_LENGTH=512;
19
20
21
22     public static /*byte[]*/Object[] hash(String password) throws NoSuchAlgorithmException, InvalidKeySpecException {
23         byte[] salt = generateSalt();
24         byte[] hashedPassword = hashPassword(password, salt);
25
26         Object[] object=new Object[2];
27         object[0]=hashedPassword;
28         object[1]=salt;
29
30         return object;
31     }
32 }
```

Abbildung 29: Java-Datei in Notepad++

Der Editor überzeugt durch seine gute Performance und seine Stabilität sowie die Vielzahl an unterstützten Programmiersprachen und wird deshalb in der Version 7.3.3 zur schnellen Einsicht und Bearbeitung diverser Source-Code-Dateien in der Diplomarbeit verwendet.

10.6.3 VMware vSphere Client

Mit dem VMware vSphere Client können virtuelle Maschinen verwaltet werden. In der HTL Perg wird die Software von Schülern verwendet, um ihre jeweiligen Projekt- oder Diplomarbeitsserver zu verwalten.

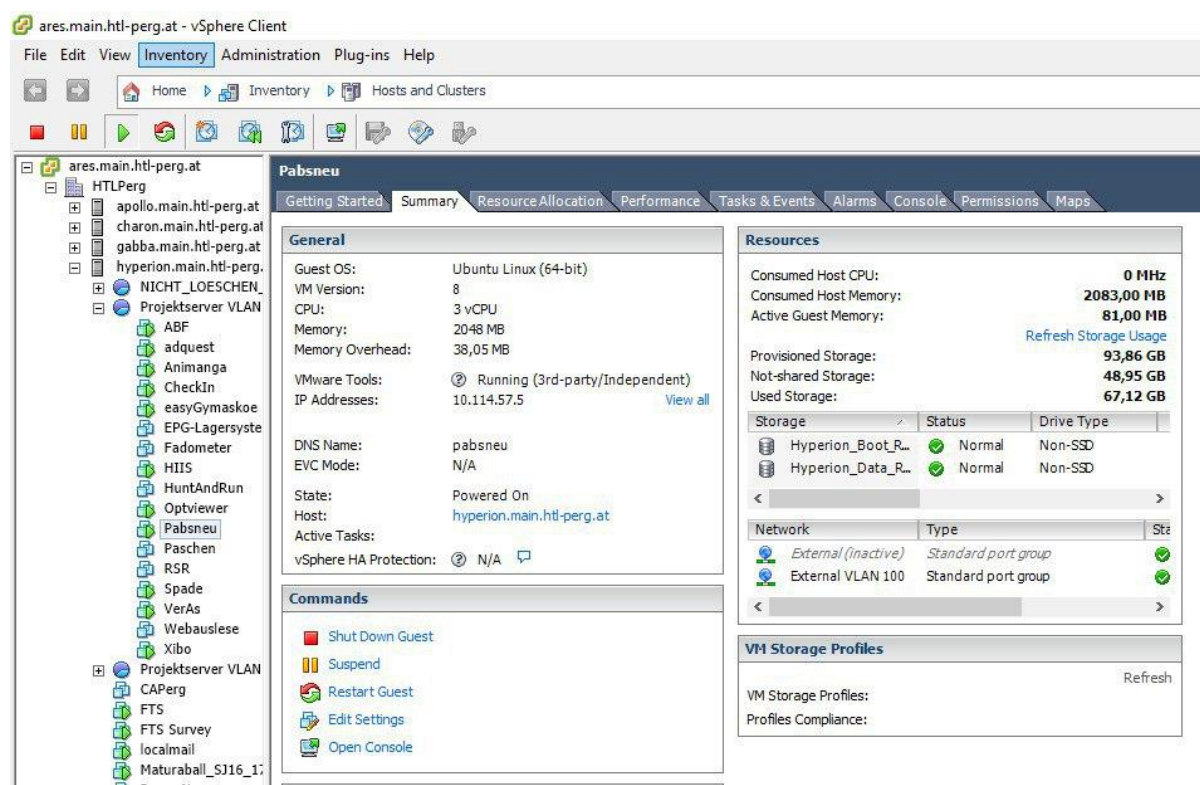


Abbildung 30: Übersicht der Diplomarbeit-VM im VMware vSphere Client

In dieser Diplomarbeit wird der VMware vSphere Client verwendet, um die projekteigene virtuelle Maschine im HTL-Netzwerk neu zu starten und um Snapshots zu erstellen, die es erlauben, den Server auf vergangene Zeitpunkte zurückzustellen.

10.6.4 SourceTree

SourceTree ist ein kostenloser, graphischer Git-Client von Atlassian. Die Software erkennt automatisch Änderungen in Dateien, welche auf den Git-Server geladen werden können. Ebenso werden Änderungen von anderen Teammitgliedern angedeutet und der aktuelle Stand des Source-Codes kann für die lokale Bearbeitung heruntergeladen werden.

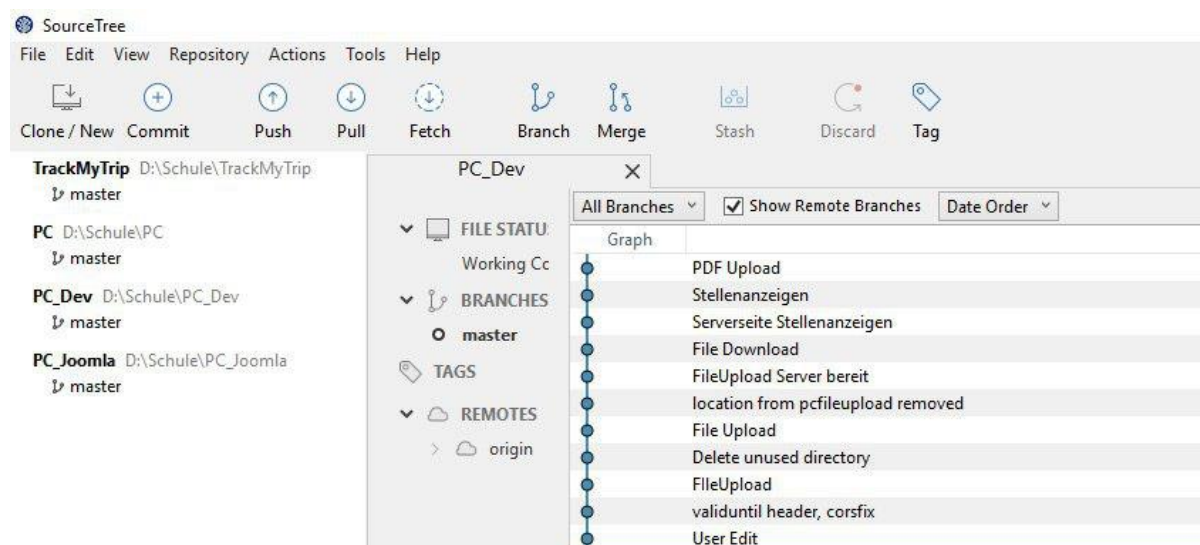


Abbildung 31: SourceTree

In der Diplomarbeit wird SourceTree als Git-Client unter Windows verwendet. Da sich das verwendete Git-Repository auf bitbucket.org befindet, ist die Integration besonders einfach, da beide Services von Atlassian sind.

10.6.5 GitKraken

GitKraken ist ein weiterer graphischer Git-Client. Entwickelt wurde die Software von axosoft für die Betriebssysteme Windows, MAC und Linux. Die Funktionsweise gleicht der von SourceTree.

Die Verwendung beruht auf der persönlichen Präferenz eines Teammitglieds gegenüber dem Konkurrenzprodukt SourceTree.



Abbildung 32: Git-Kraken-Logo (Zazzle, 2017)

10.6.6 Microsoft Office 2016

Microsoft Office bezeichnet eine Software-Suite, die zahlreiche Anwendungen für den Büroalltag beinhaltet.

In der Diplomarbeit werden die Programme Excel 2016 und Word 2016 für die Dokumentation und Planung sowie PowerPoint 2016 für Projektpräsentationen eingesetzt.



Abbildung 33: Microsoft Office-Logo (Microsoft, 2017)

10.6.7 Dropbox

Dropbox bezeichnet sowohl einen Cloud-Dienst zur Ablage von Dateien sowie den zugehörigen Client. Mittels Dropbox können Personen ihre persönlichen Daten in die Cloud laden und weltweit von mehreren Geräten darauf zugreifen. Dateien oder Verzeichnisse können auch für andere Nutzer freigegeben werden, um den Zugriff für mehrere Personen zu ermöglichen. Ebenso kann ein Link generiert werden, der den öffentlichen Zugriff ohne Account für eine Datei ermöglicht. Auch für Firmenkunden gibt es Modelle.



Dropbox

Abbildung 34: Dropbox-Logo (Copernic Forum, 2017)

Über Dropbox können in der Diplomarbeit sehr schnell Dateien ausgetauscht werden. Die gleichzeitige Bearbeitung von Dateien ist allerdings nicht problemlos möglich, wodurch sich der Dienst nicht für die Speicherung der Source-Code-Dateien eignet. Weitere Mankos wie die fehlende Versionierung tragen dazu ebenfalls bei.

10.6.8 GIMP

GIMP steht für GNU Image Manipulation Program und ist ein Open-Source-Bildbearbeitungsprogramm.

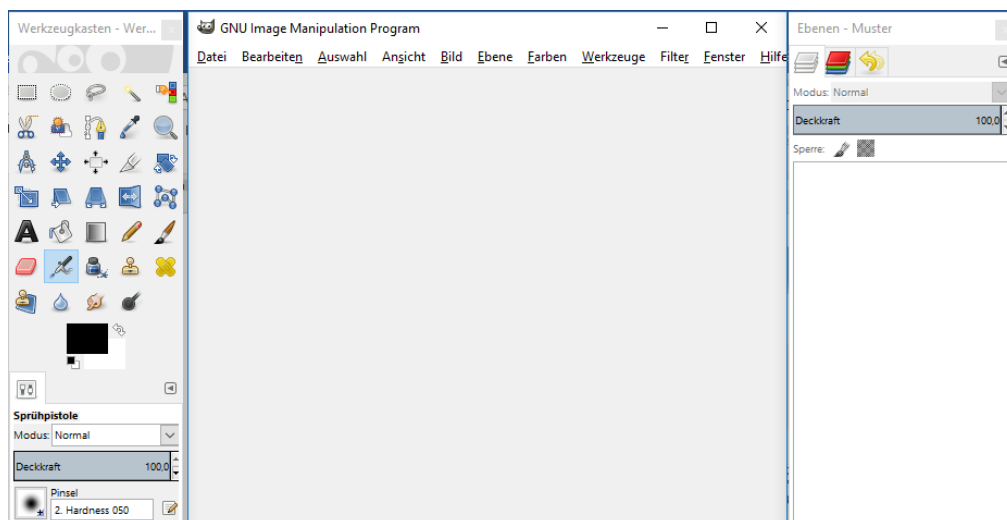


Abbildung 35: Leere GIMP-Anwendung nach dem Start

In der Diplomarbeit wurde Mittels GIMP 2 das Projektlogo entworfen sowie das Projektplakat erstellt.

10.6.9 Adobe Photoshop

Adobe Photoshop ist ein Bildbearbeitungsprogramm für Pixelgrafiken. Es ist ein Bestandteil der Adobe Creative Cloud.

In der Diplomarbeit wurde Adobe Photoshop in der Version CC 2017 verwendet, um die verwendeten Bilder des informellen Teils zu bearbeiten.



Abbildung 36: Logo Adobe Photoshop (wikipedia.org, 2017)

10.6.10 FileZilla Client

FileZilla ist ein kostenloser FTP-Client, mit dem Daten zwischen zwei Rechnern per FTP ausgetauscht werden können.

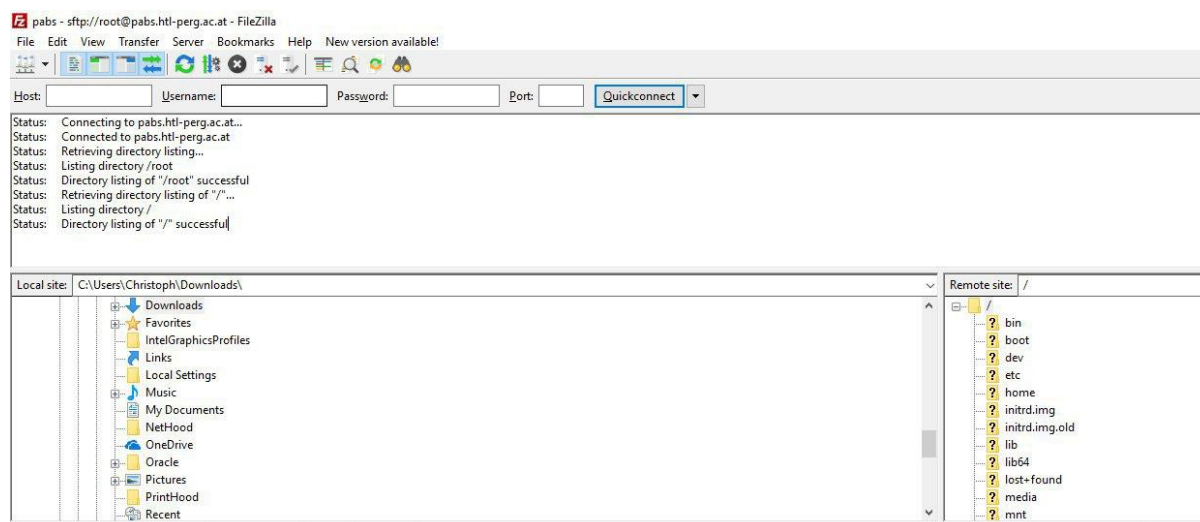


Abbildung 37: FileZilla Client

In der Diplomarbeit wird der FileZilla Client zum Datenaustausch zwischen Windows-Entwicklungsrechner und dem Ubuntu-Projektserver verwendet.

10.6.11 Telegram

Telegram ist ein kostenloser Instant Messenger mit Clients für alle gängigen Plattformen, sowohl auf mobilen Geräten als auch auf Desktop-Betriebssystemen. Telegram ermöglicht neben dem Austausch von Nachrichten auch das Versenden von Dateien, die dauerhaft gespeichert werden. Die Speicherung erfolgt in einer Cloud, wodurch die Daten auf allen persönlichen Endgeräten überall auf der Welt abrufbar sind.



Abbildung 38: Telegram-Logo (Google Play, 2017)

In der Diplomarbeit wird Telegram in eigens dafür angelegten Gruppen dazu verwendet, um Nachrichten sowie Dateien zwischen den Teammitgliedern auszutauschen.

10.6.12 Postman

Postman ist ein kostenloser, graphischer Client zum Testen von APIs. Mit der Software können beispielsweise REST-Requests abgesetzt werden, um die Funktionsfähigkeit einer API zu überprüfen.

Neben einer möglichen Installation unter Mac, Windows oder Linux ist auch eine beliebte Erweiterung für Google Chrome von Postman verfügbar. (vgl. getpostman.com, 2017)



Abbildung 39: Postman-Logo
(<http://blog.getpostman.com>, 2017)

In dieser Diplomarbeit wird Postman zum Testen der REST-API verwendet.

11 Planung und Realisierung

Im Kapitel Planung und Realisierung wird die Planung und Umsetzung der Diplomarbeit genauer erläutert. Die verwendeten Technologien wurden bereits im vorigen Kapitel beschrieben. Allerdings muss berücksichtigt werden, dass während der Entwicklung des Projekts ein Change-Request an den Auftraggeber übermittelt wurde. Dies geschah aufgrund aufgetretener Probleme, welche bei der Entwicklung des internen Bereiches der Webapplikation via der Technologie Joomla ersichtlich wurden.

11.1 Change-Request

Ursprünglich geplant:

- Webapplikation gänzlich mit Joomla entwickelt
- Native Android-App
- Native iOS-App

Nach dem Change-Request geplant:

- Öffentlicher Bereich der Webapplikation mit Joomla entwickelt
- Interner Bereich mit Angular2 entwickelt

Der Change-Request ist im Anhang beigelegt.

11.2 Meilensteine vor dem Change-Request

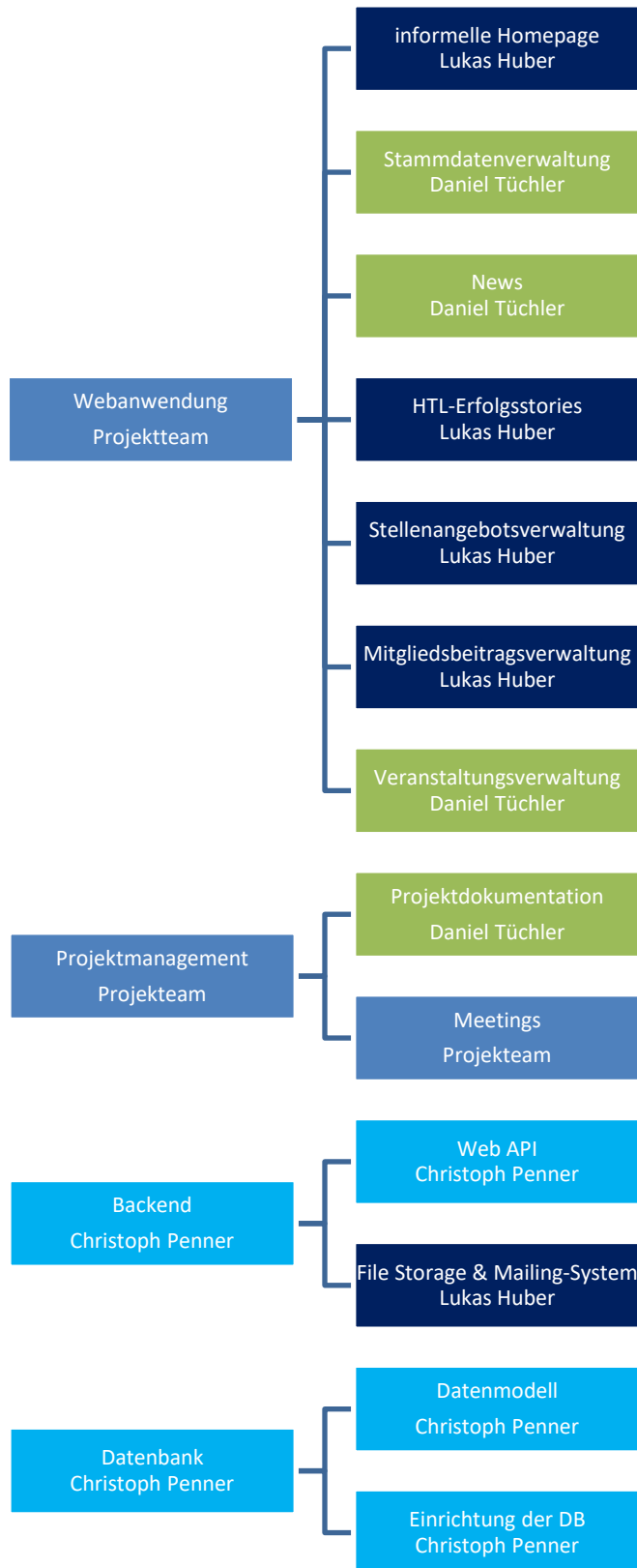
Die Umstände des Change-Requests werden im Kapitel Resümee noch näher erläutert.



11.3 Meilensteine nach dem Change-Request



11.4 Projektstrukturplan



Der Hauptverantwortliche des jeweiligen Bereiches ist im Projektstrukturplan immer in der zweiten Zeile vermerkt.

11.5 Verantwortlichkeiten (IMV-Matrix)

	Huber	Penner	Tüchler
Projektmanagement	V	M	M
Webapplikation	M	I	V
Backend	M	V	I
Datenbank	I	V	I
File Storage und Mailing System	V	M	I
Projektdokumentation	M	M	V

V Verantwortlich

M Mitarbeit

I Information

11.6 Funktionale Idee

11.6.1 Absolventen & Firmen

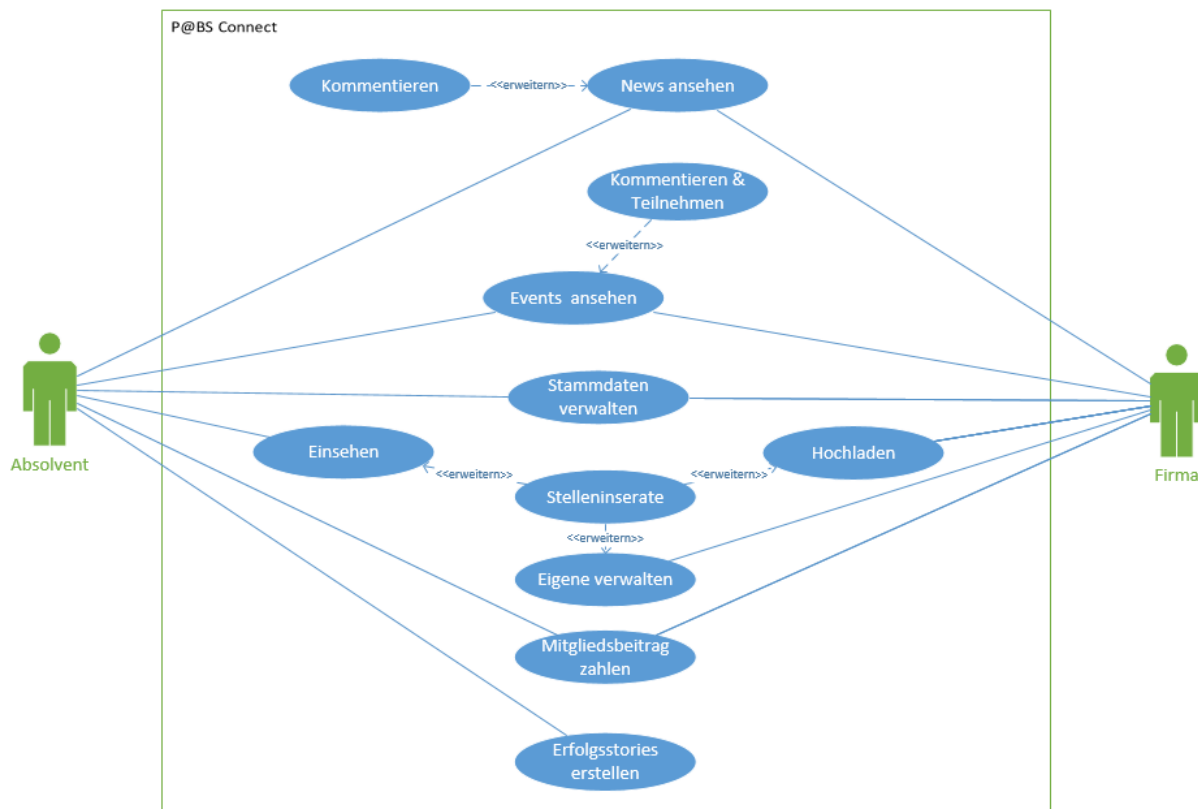


Abbildung 40: Funktionale Idee Absolventen & Firmen

11.6.2 Administrator

Der Administrator verfügt über sämtliche Funktionen der Absolventen und Firmen. Zusätzlich stehen ihm die nachfolgenden Funktionalitäten zur Verfügung.

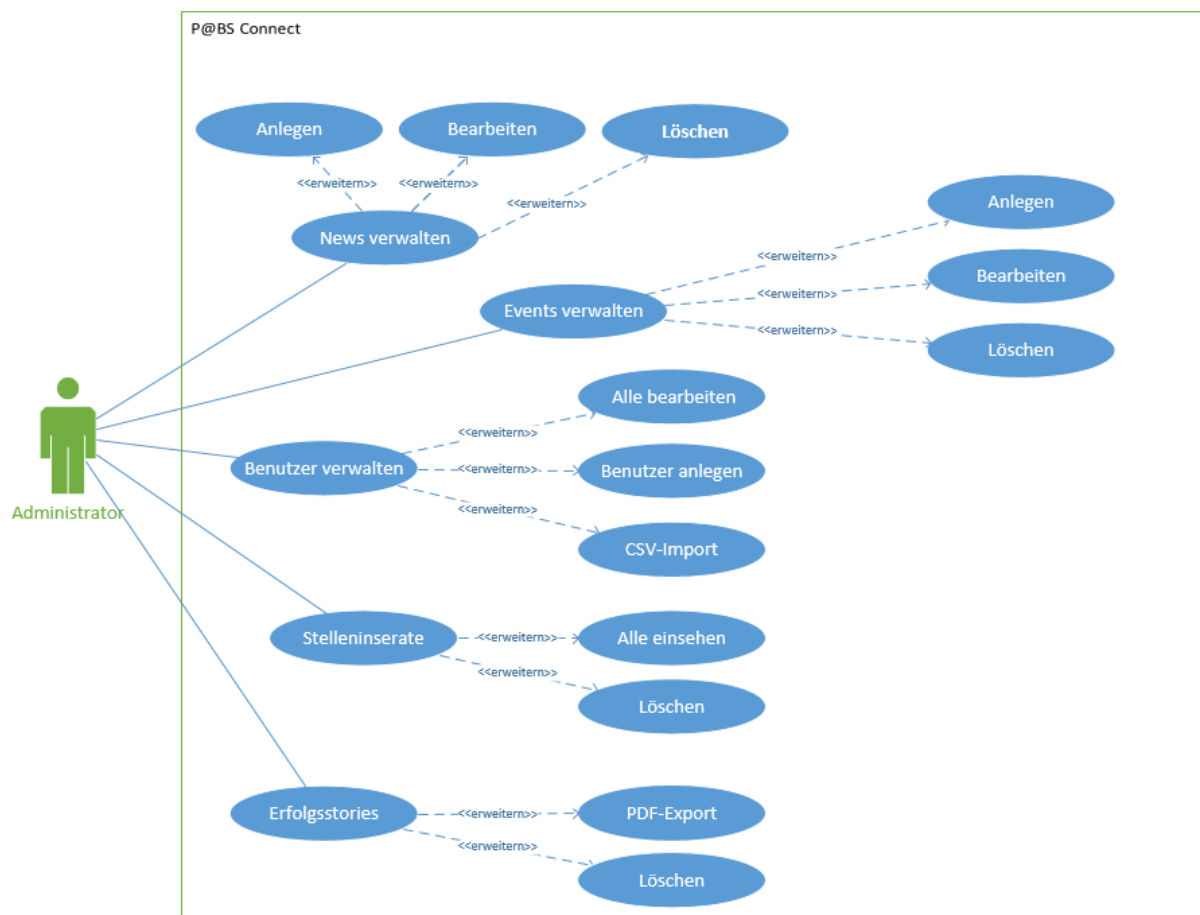


Abbildung 41: Funktionale Idee Administrator

11.7 Use-Case-Diagramme

11.7.1 Events

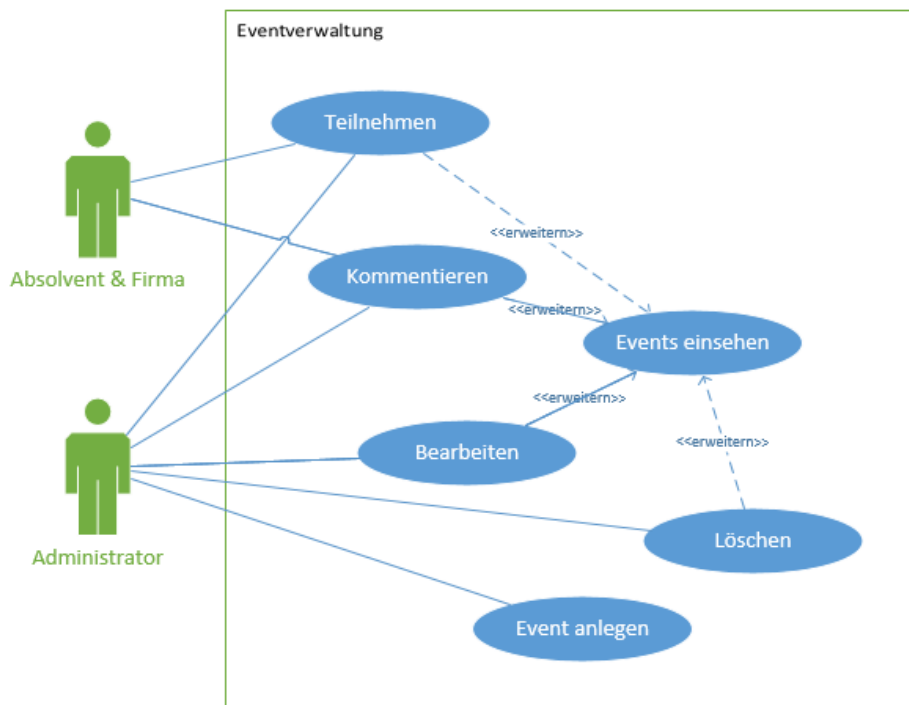


Abbildung 42: Use-Case-Diagramm Events

11.7.2 Stellenangebote

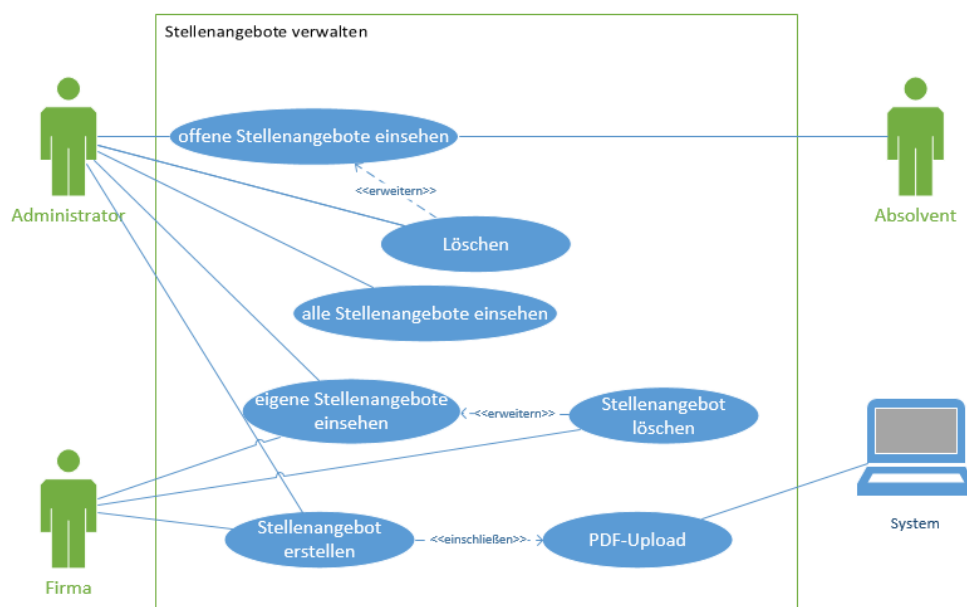


Abbildung 43: Use-Case-Diagramm Stellenangebote

11.7.3 Mitgliedsbeitrag

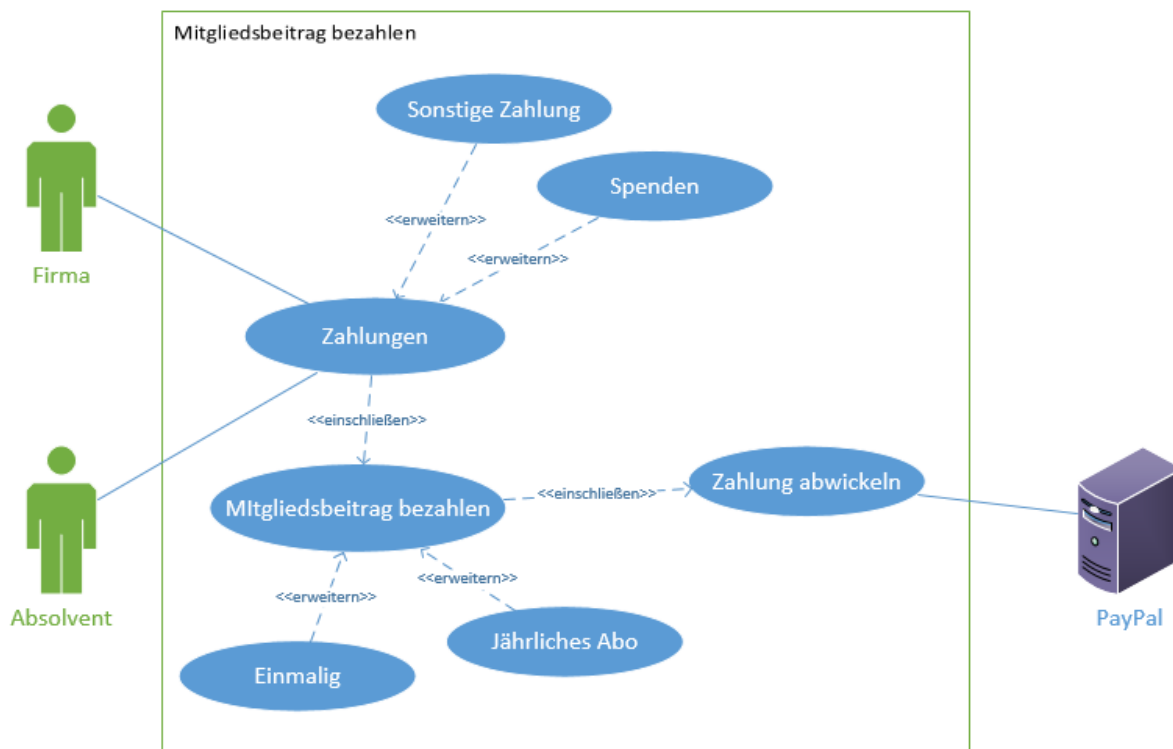


Abbildung 44: Use-Case-Diagramm Mitgliedsbeitrag

11.7.4 Erfolgstories

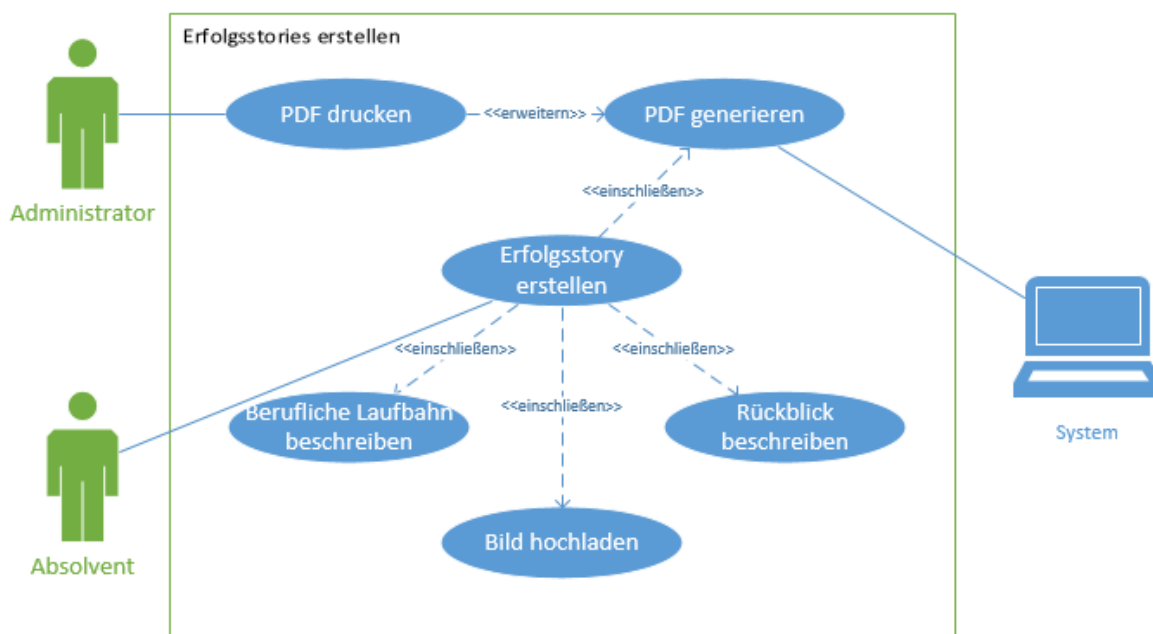


Abbildung 45: Use-Case-Diagramm Erfolgstories

11.8 Architektur

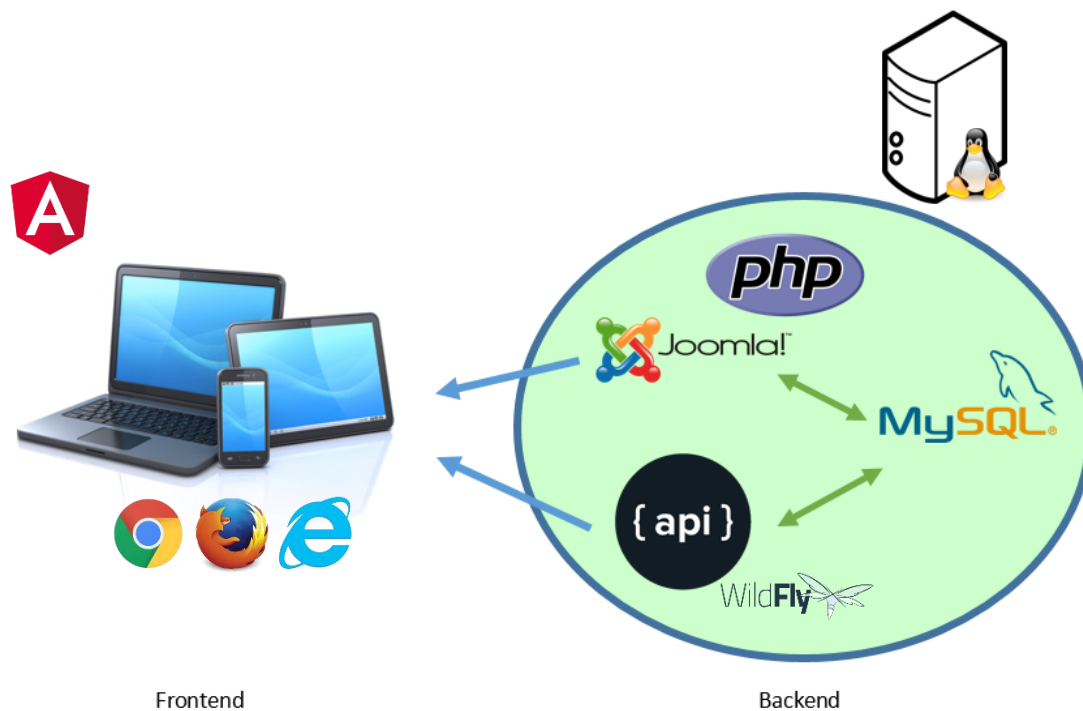


Abbildung 46: P@BS Connect Architektur (clker.com, 2017), (centratrip.com, 2017), (wikipedia.org, 2017), (clker.com, 2017), (wikipedia.org, 2017), (memz.co, 2017), (fedoraproject.org, 2017) (MBT Check, 2017) (outdatedbrowser.com, 2017)

11.8.1 Backend

Das Backend unseres Systems läuft unter Ubuntu 16.04 in einer virtuellen Maschine im Netzwerk der HTL Perg. Darauf installiert sind MySQL, WildFly und Joomla, sowie die dafür benötigten Technologien, wie PHP, Java, etc. in den jeweiligen aktuellsten Versionen. Unser Web-API wird als REST-Service über einen WildFly-Server angeboten. Joomla und das API beziehen die notwendigen Daten aus der MySQL-Datenbank. Genauer zu den einzelnen verwendeten Technologien ist im Punkt Grundlagen bereits angeführt worden.

Das Datenmodell sowie die API werden im Kapitel Implementierung beschrieben.

11.8.2 Frontend

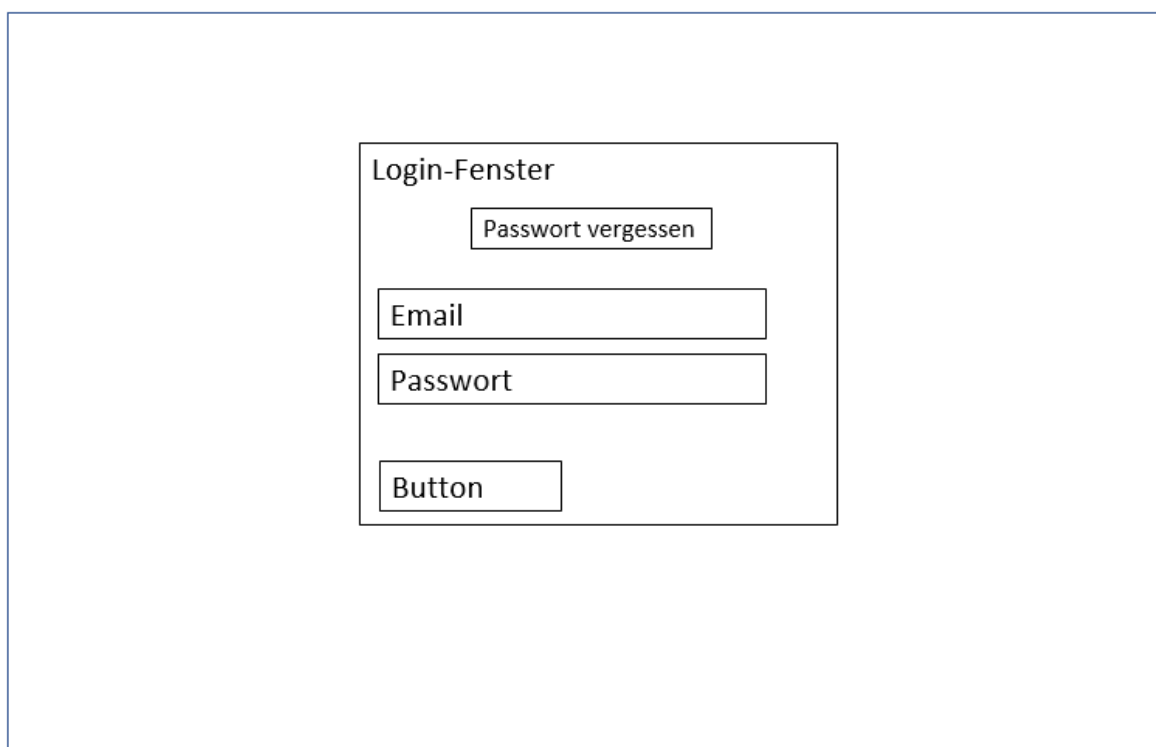
Das Webportal ist mit allen gängigen Browsern, wie Google Chrome, Edge und Internet Explorer von Microsoft, Mozilla Firefox sowie Safari verwendbar. Da wir ein responsives Layout verwendet haben, ist es möglich, das Frontend von P@BS Connect auch auf mobilen Geräten zu verwenden.

11.9 Design des Webportals - Seitenaufbau

Ein besonderes Augenmerk wird in dieser Diplomarbeit auf die Übersichtlichkeit und selbsterklärende Darstellung geworfen. Der Benutzer soll sich bereits beim erstmaligen Aufrufen der Webseite sofort zurechtfinden.

Im Folgenden sind die jeweiligen Abschnitte des Webportals skizziert.

11.9.1 Login



Die Skizze zeigt ein Login-Fenster mit dem Titel 'Login-Fenster'. Darin sind folgende Elemente dargestellt:

- Ein Button 'Passwort vergessen' am oberen Rand.
- Ein Eingabefeld 'Email'.
- Ein Eingabefeld 'Passwort'.
- Ein Button 'Button' am unteren Rand.

Abbildung 47: Webportal Entwurf Login-Seite

Um sich anzumelden und in den internen Bereich zu gelangen, ist die Eingabe von E-Mail und Passwort notwendig. Dafür sind zwei Eingabefelder und ein „Anmelden“-Button vorgesehen. Falls das Passwort vergessen wurde, wird per Knopfdruck auf den „Passwort vergessen“-Button die E-Mail-Adresse des Administrators angezeigt, um sich mit ihm in Verbindung zu setzen und das Passwort zurückzusetzen.

11.9.2 Side-Menu

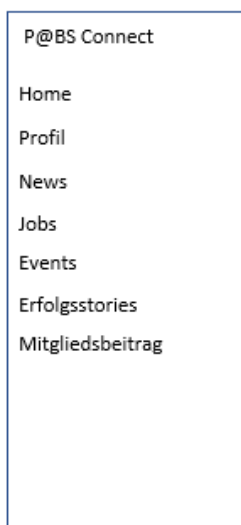


Abbildung 48: Webportal Entwurf Side-Menu

In den folgenden Entwürfen wird das Side-Menu aufgrund von Übersichtlichkeitsgründen nicht mehr in den Planungsskizzen eingezeichnet. Beim Klick auf einen Menüpunkt wird entsprechend weitergeleitet. Das Side-Menu enthält die Punkte:

- Home
 - Zeigt die Startseite an
- Profil
 - Sämtliche Funktionen zur Stammdatenpflege bzw. Benutzerverwaltung
- News
 - Abbildung aller News bzw. Oberflächen zum Erstellen und Editieren von News
- Jobs
 - Einsehen, Erstellen und Löschen von Stellenangeboten
- Events
 - Abbildung aller Events bzw. Oberflächen zum Erstellen und Editieren von Events
- Erfolgsstories
 - Erstellen und Verwalten der HTL-Perg-Erfolgsstories
- Mitgliedsbeitrag
 - Möglichkeiten zur Bezahlung des Mitgliedsbeitrages, zum Begleichen von Forderungen oder zum Spenden

11.9.3 Startseite

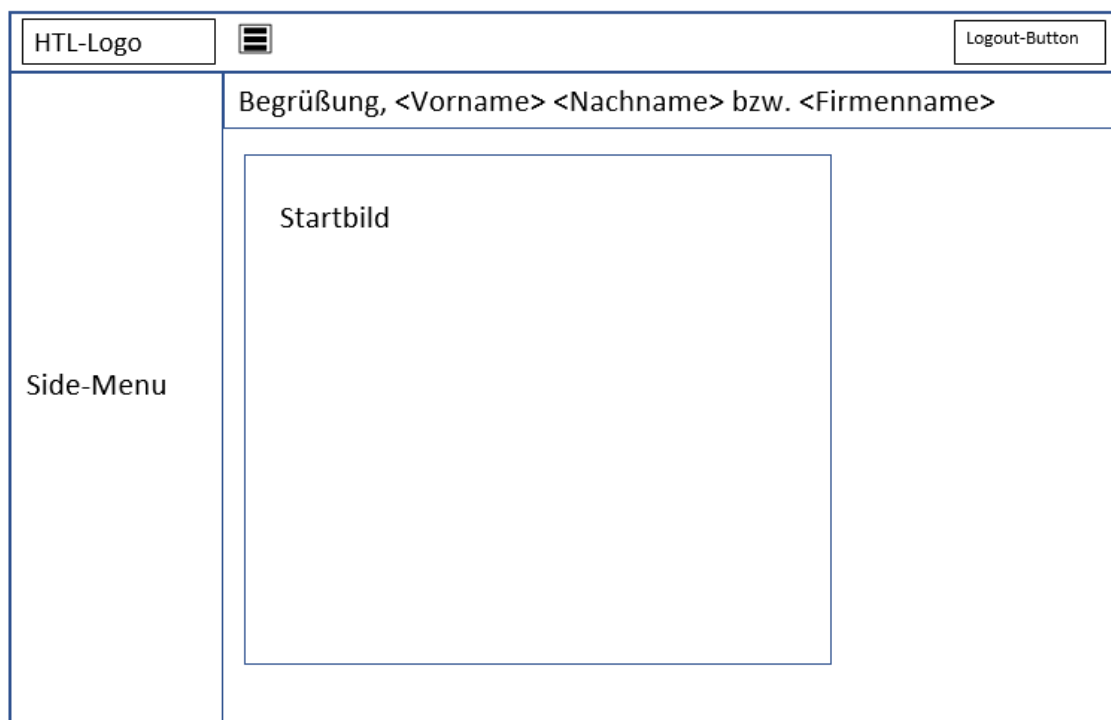


Abbildung 49: Webportal Entwurf Startseite

Die Startseite wird nach der erfolgreichen Anmeldung angezeigt. Ein Bild der HTL-Perg und eine Überschrift mit einer Begrüßung für den aktuell eingeloggten Benutzer sind sichtbar.

Die Schaltfläche mit den drei Querstrichen in der Navigationsleiste neben dem Logo dient zum Ein- und Ausklappen des Side-Menus für die Verwendung auf kleinen Bildschirmen.

11.9.4 Profil



The diagram illustrates the layout of a web portal profile page. At the top left is the HTL-Logo, followed by a hamburger menu icon. At the top right is a Logout-Button. Below the header is a main content area with a title 'Überschrift'. On the left side, there is a vertical 'Side-Menu'. The main content area contains a 'Stammdatenformular' with input fields for Vorname, Nachname, Adresse, PLZ, Stadt, and Email. Below these fields are buttons for 'Passwort ändern', 'Events und News', and 'Jobs'. At the bottom of the form are 'Speichern' and 'Zurück' buttons. To the right of the form is a box titled 'Zu den Administratorfunktionen' containing an 'Administratorfunktionen' button.

Abbildung 50: Webportal Entwurf Profil

Beim Klick auf „Profil“ im Side-Menu kann der Benutzer seine eigenen Stammdaten verwalten. Weiters kann er auswählen, ob er Benachrichtigungen bei neuen Events und News bzw. neuen Jobangeboten erhalten will. Administratoren haben eine zusätzliche Oberfläche „Administratorfunktionen“, über welche sie einen Benutzer erstellen bzw. direkt mehrere Benutzer aus einer CSV-Datei importieren können. Außerdem können sie sich alle Nutzer anzeigen lassen und deren Daten verwalten oder das Konto löschen.



Abbildung 51: Webportal Entwurf Profil (Administratorfunktionen)

11.9.5 Jobs – Funktionsübersicht

Administratoransicht

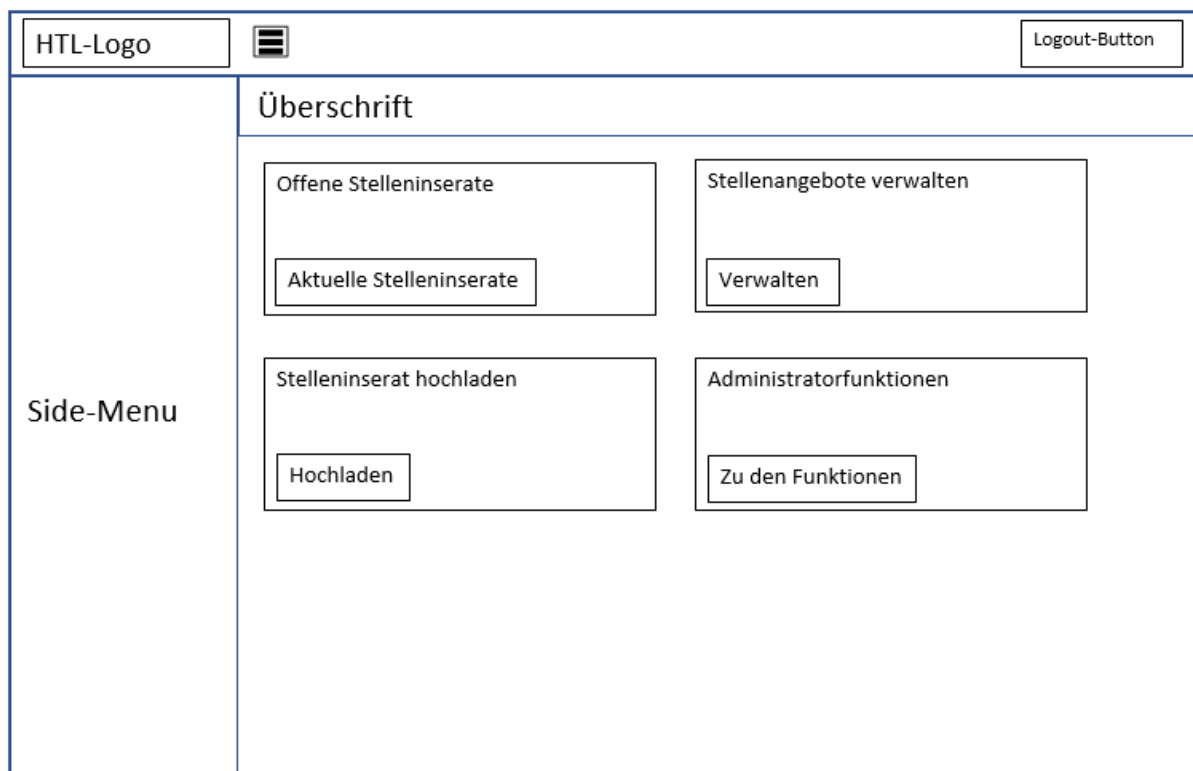


Abbildung 41: Webportal Entwurf Jobs - Funktionsübersicht Administrator

Auf dieser Seite haben Administratoren die Möglichkeit, alle Stellengebote einzusehen bzw. zu verwalten.

Die Firmen sehen jedoch nur die Funktionen „Stellenangebote verwalten“ und „Stellenangebot hochladen“.

Absolventen sehen lediglich die aktuellen Stelleninserate, die sie einsehen können.

11.9.6 Jobs – Offene Stelleninserate

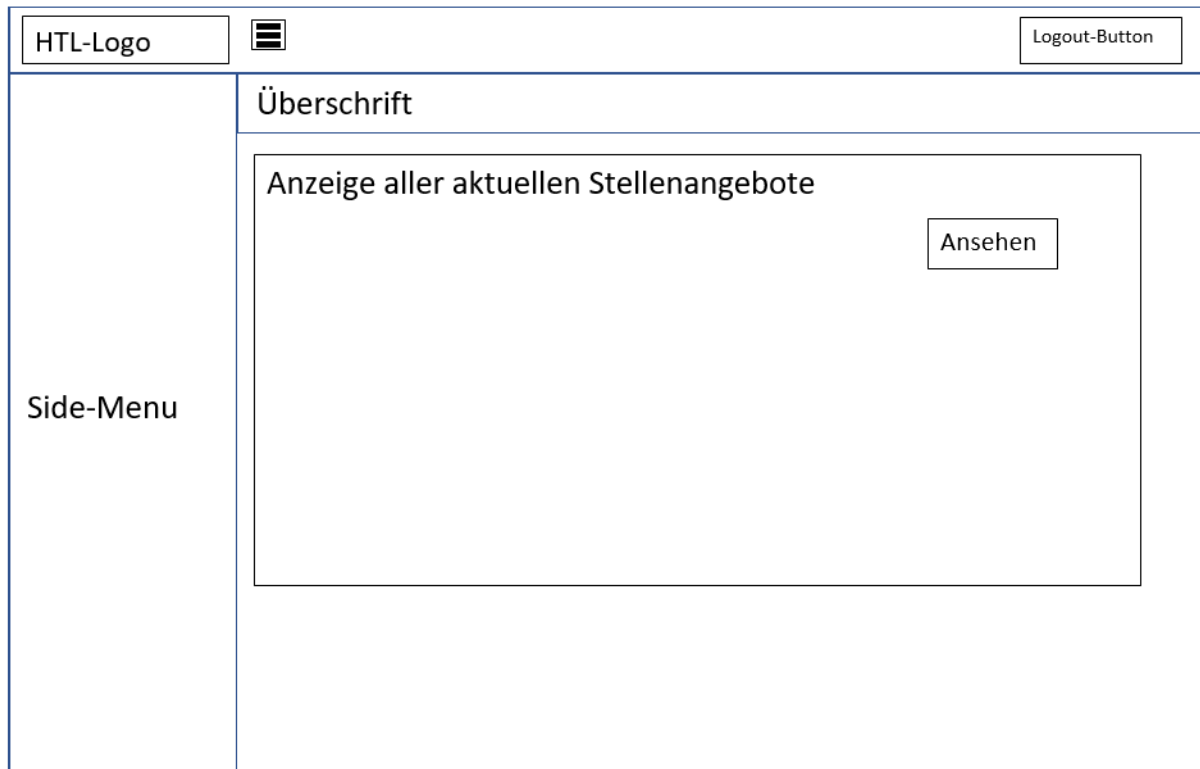


Abbildung 42: Webportal Entwurf Jobs - Offene Stelleninserate

Hier können die Benutzer sämtliche aktiven Stelleninserate einsehen. Diese werden in Form einer Tabelle dargestellt. Mit einem Klick auf den Button „Ansehen“ in der entsprechenden Tabellenzeile wird das PDF des Stelleninserates in einem neuen Tab im Internet-Browser geöffnet. Firmen sehen jedoch nur die eigenen Stellenangebote.

11.9.7 Jobs – Stellenangebote verwalten

Für Firmen

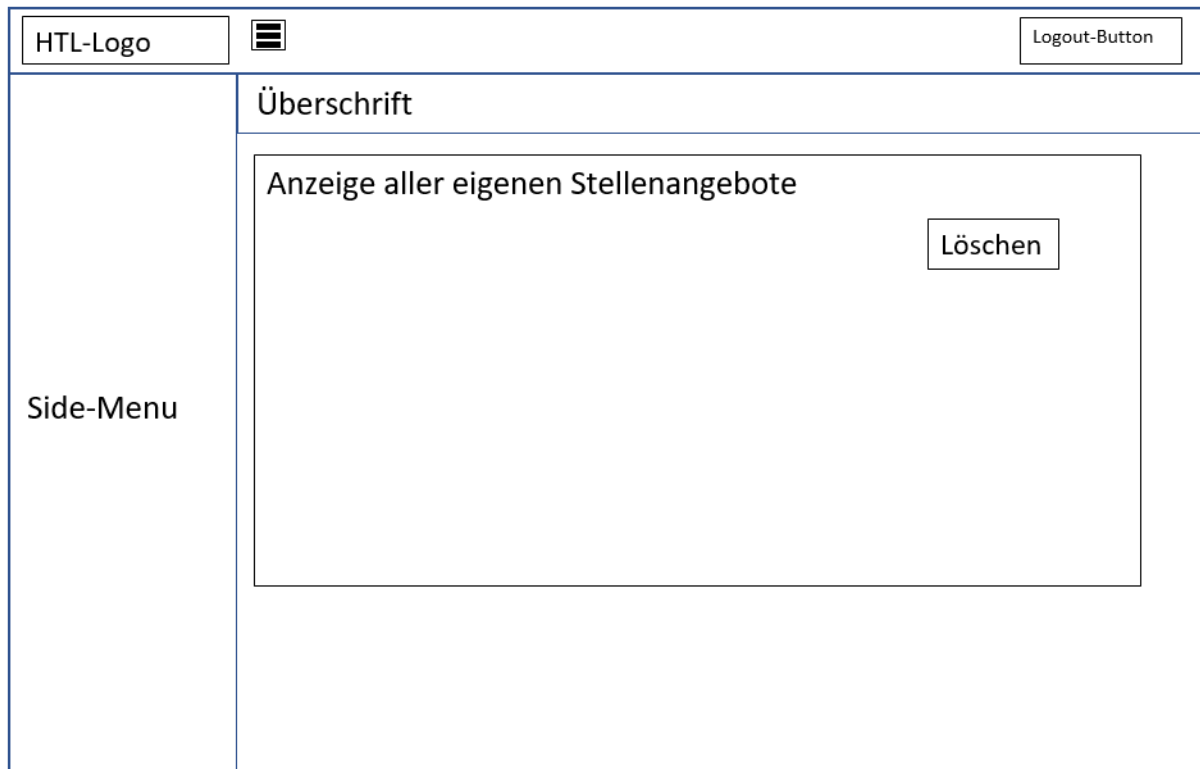
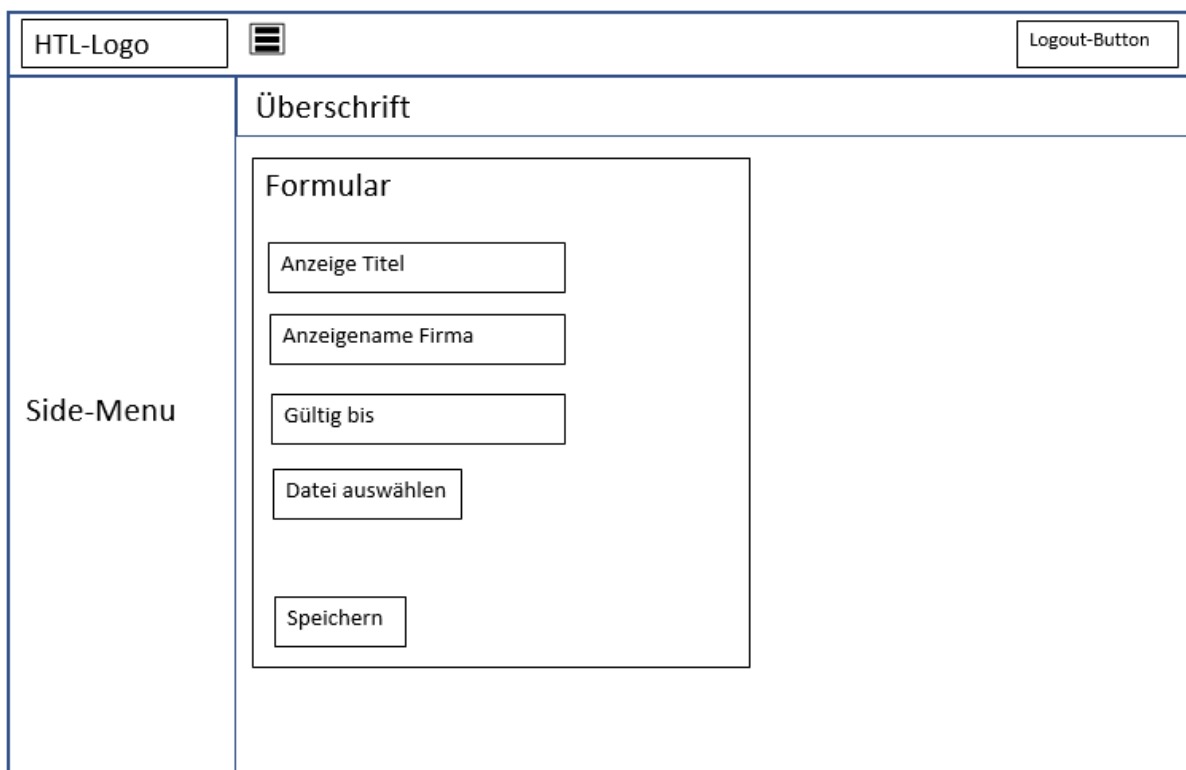


Abbildung 43: Webportal Entwurf Jobs - Stellenangebote verwalten (Administrator & Firma)

Hier können Firmen Stellenangebote ansehen und löschen, falls diese veraltet oder bereits vergeben sind. Firmen können nur ihre eigenen Stellenangebote ansehen sowie löschen.

11.9.8 Jobs – Stelleninserat hochladen

Administrator & Firma



HTL-Logo	☰	Logout-Button
Side-Menu	Überschrift	
	Formular	
	Anzeige Titel	
	Anzeigename Firma	
	Gültig bis	
	Datei auswählen	
	Speichern	

Abbildung 52: Webportal Entwurf Jobs - Stelleninserat hochladen (Administrator & Firma)

In diesem Formular können sowohl Administratoren als auch Firmen alle relevanten Daten für Stelleninserate eingeben. Die Gültigkeit gibt an, wie lange dieses Stelleninserat verfügbar und daher sichtbar sein soll. Dieses Datum kann händisch eingetragen oder per Kalender selektiert werden. Über den Button „Datei auswählen“ kann die gewünschte PDF-Datei für die Stellenbeschreibung ausgewählt werden.

11.9.9 Jobs – Administratorfunktionen

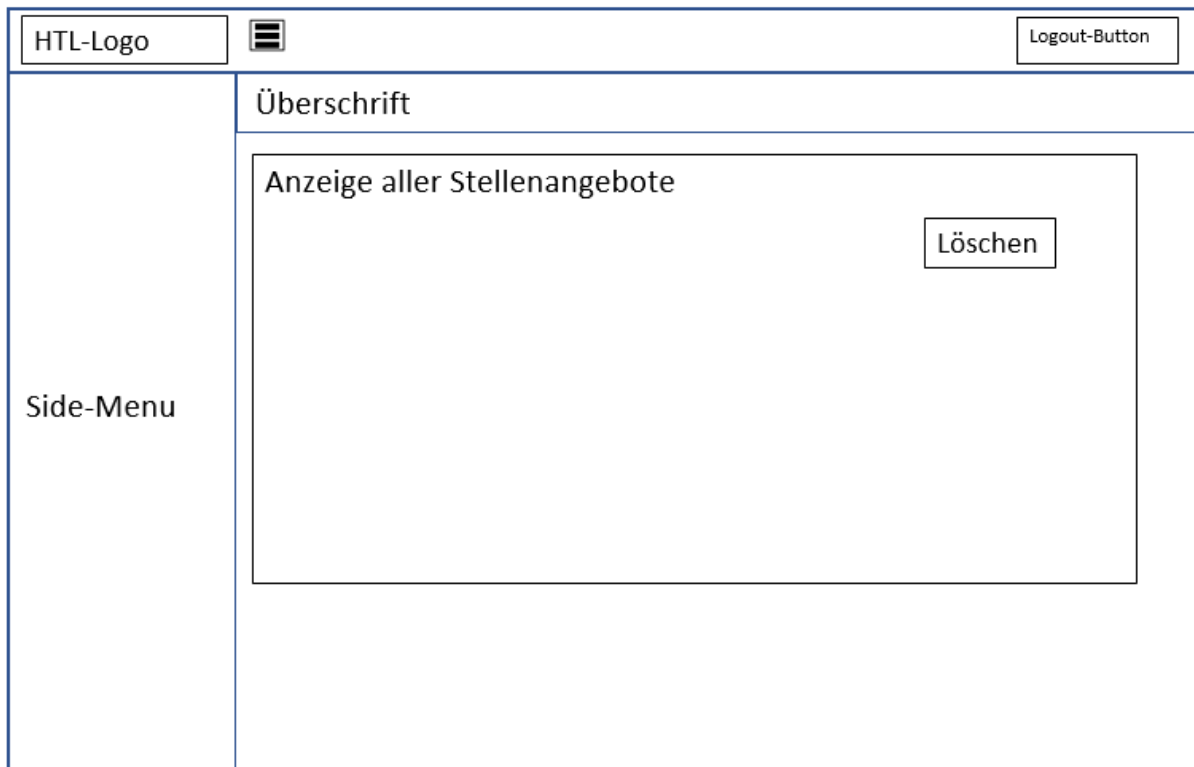


Abbildung 53: Webportal Entwurf Jobs - Administratorfunktionen

Hier haben Administratoren die Möglichkeit, alle Stelleninserate einzusehen. Falls ein bestimmtes Inserat gelöscht werden soll, kann dies über den „Löschen“-Button in der jeweiligen Tabellenzeile erledigt werden.

11.9.10 Events bzw. News – Übersicht

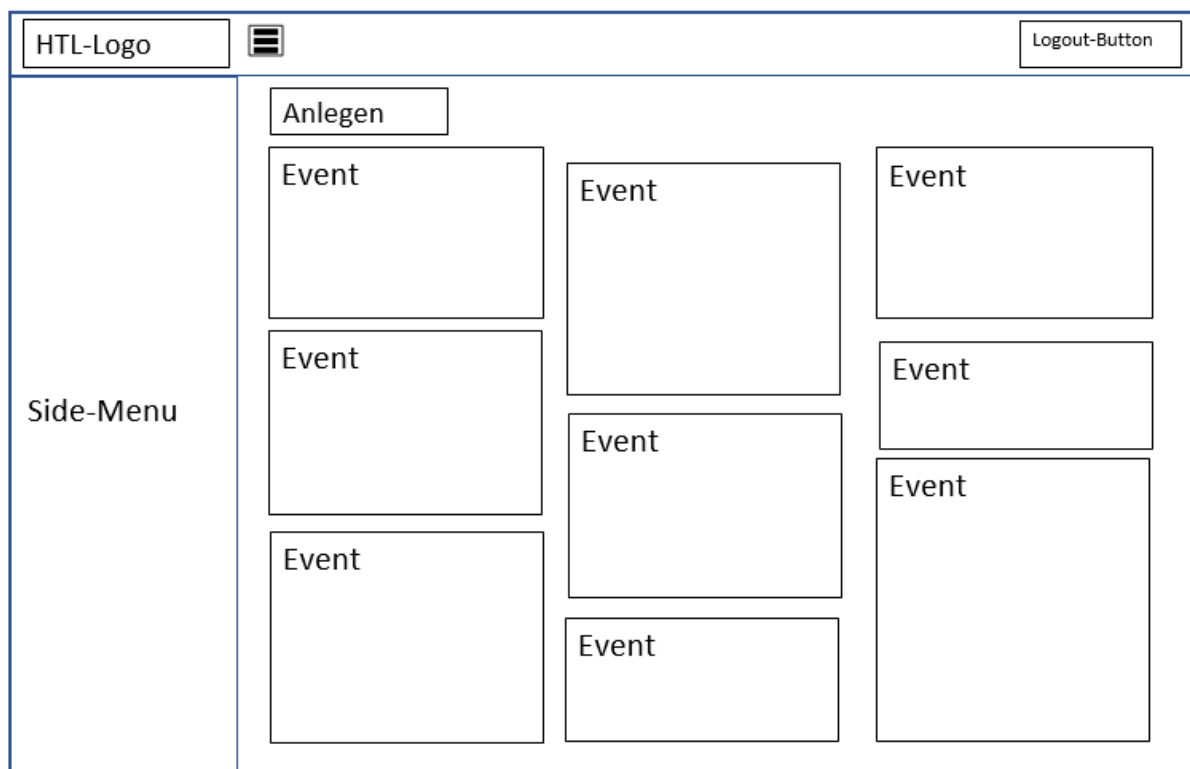


Abbildung 54: Webportal Entwurf Events- und News-Übersicht

Die folgenden Entwürfe unter diesem Punkt sind vom Aufbau her für Events und News ident. Der Unterschied liegt darin, dass Events über einen „Teilnehmen“-Button in der Detailansicht verfügen und die Felder „Veranstaltungsort“ und „Veranstaltungsdatum“ beinhalten.

In dieser Ansicht werden alle aktuellen Events bzw. News angezeigt. Diese sind zur besseren Übersichtlichkeit in einem Masonry-Grid-Layout dargestellt. Datum, Autor, eine kurze Beschreibung, die Anzahl der verfassten Kommentare sowie das Titelbild - sofern vorhanden - werden zusätzlich angezeigt.

11.9.11 Events bzw. News – Einzelansicht

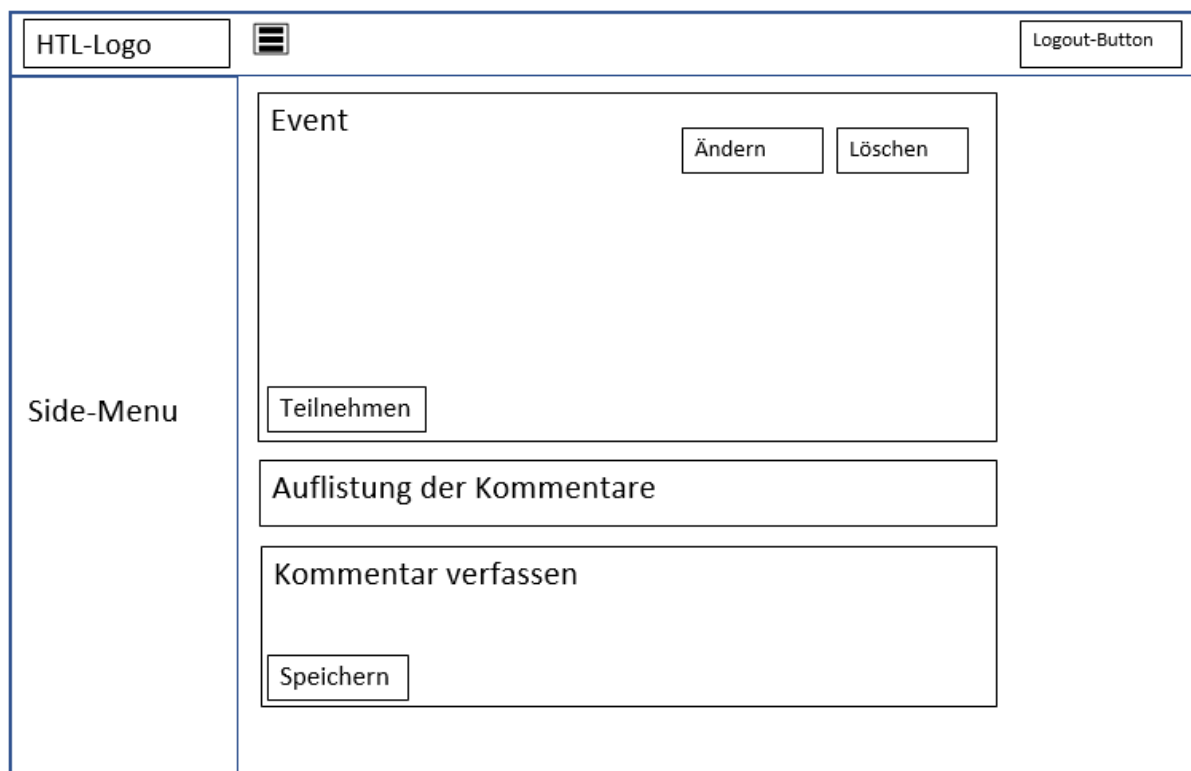


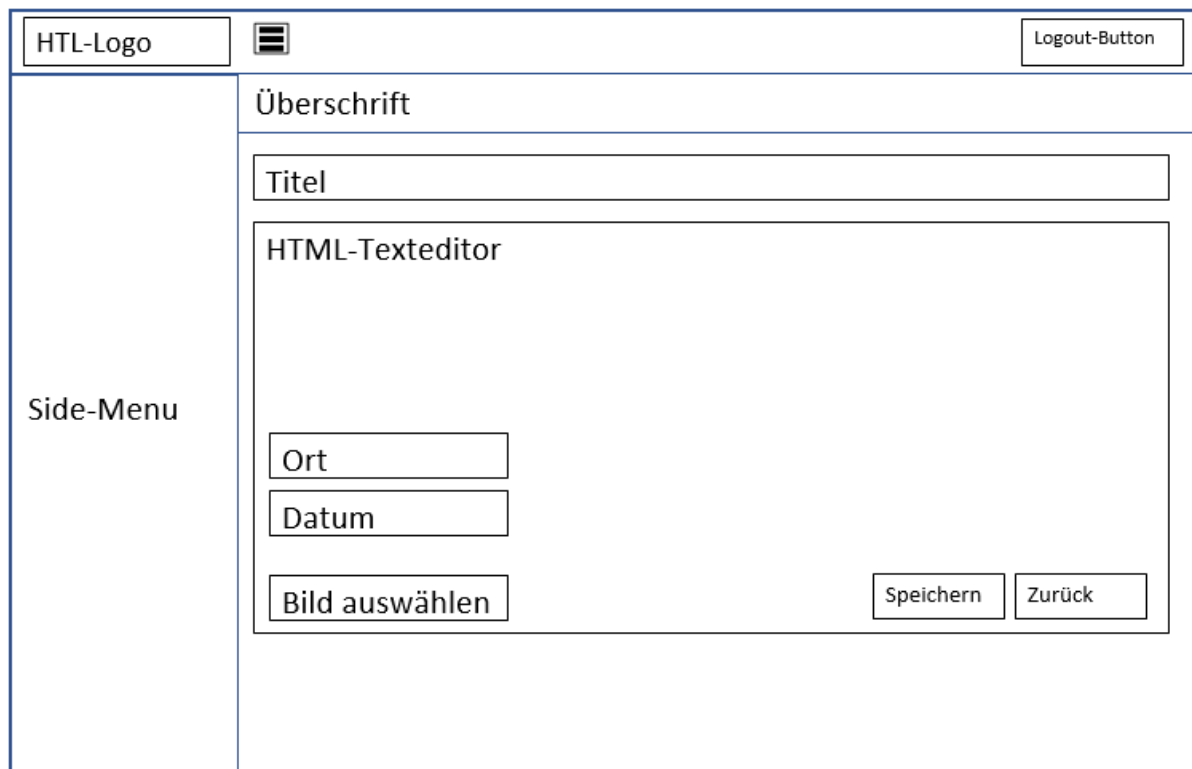
Abbildung 55: Webportal Entwurf Events bzw. News - Einzelansicht

In der Detailansicht wird das Event bzw. der News-Beitrag alleine angezeigt. Hier ist der Beschreibungstext in seiner vollen Länge sichtbar. Weiters können Absolventen die Events und News-Beiträge kommentieren und die Kommentare anderer ansehen. Jeder Benutzer kann später seine eigenen Kommentare löschen, Administratoren können alle Kommentare löschen.

Der „Teilnehmen“-Button ist nur bei den Events sichtbar. Wird dieser geklickt, erscheint der Name des Benutzers in der Teilnehmerliste und der Button verändert sich zu einem „Absagen“-Button. Wird dieser geklickt, so wird der Benutzer wieder aus der Teilnehmerliste entfernt.

Die Buttons „Ändern“ und „Löschen“ sind nur für Administratoren ersichtlich. So kann das Event bzw. der News-Beitrag einfach bearbeitet oder gelöscht werden. Um das Event bzw. den News-Beitrag bearbeiten zu können, wird man auf eine neue Seite weitergeleitet, wo den Administratoren ein HTML-Editor zur Verfügung gestellt wird.

11.9.12 Events bzw. News erstellen und bearbeiten




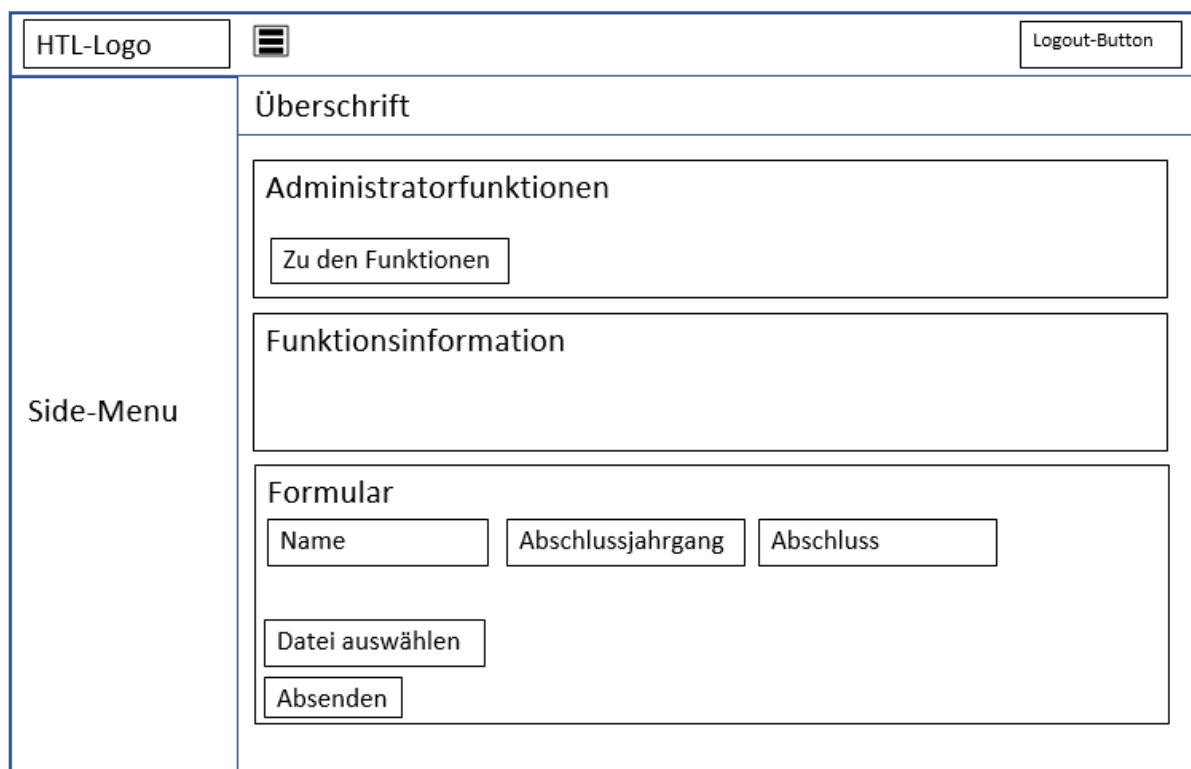
HTL-Logo		Logout-Button
Side-Menu	Überschrift	
	Titel	
	HTML-Texteditor	
	Ort	
	Datum	
	Bild auswählen	Speichern Zurück

Abbildung 56: Webportal Entwurf Events bzw. News erstellen und bearbeiten

Um Events bzw. News zu erstellen und zu bearbeiten wird den Administratoren eine Seite mit diversen Inputfeldern und einem HTML-Editor zur Verfügung gestellt. Die Eingabefelder „Ort“ und „Datum“ sind nur bei den Events ersichtlich. Für den Titel ist ein weiteres Eingabefeld verfügbar - und falls gewünscht kann ein Titelbild über den Button „Datei auswählen“ hochgeladen werden. Über den „Zurück“-Button gelangt man wieder auf die Events- bzw. News-Ausgangsseite.

11.9.13 Erfolgsgories



HTL-Logo	☰	Logout-Button
Side-Menu	Überschrift	
	Administratorfunktionen Zu den Funktionen	
	Funktionsinformation	
Formular		
Name		
Abschlussjahrgang		
Abschluss		
Datei auswählen		
Absenden		

Abbildung 57: Webportal Entwurf Erfolgsgories

Um als Absolvent eine Erfolgsgorie zu erstellen, wird ein Eingabeformular angeboten. Dort müssen einige Daten zur Ausbildung, der Werdegang des Absolventen sowie eine rückblickende Beschreibung über die Zeit an der HTL Perg angegeben werden. Weiters steht eine Funktion zum Hochladen eines Fotos bereit. Nach dem Ausfüllen wird aus den Daten eine PDF-Datei generiert, die ausgedruckt und in der HTL Perg ausgestellt werden kann.

Für Administratoren wird zusätzlich das Feld „Administratorfunktionen“ eingeblendet. Über den Button „Zu den Funktionen“ gelangt man zu diesen Funktionen.

11.9.14 Erfolgstories – Administratorfunktionen

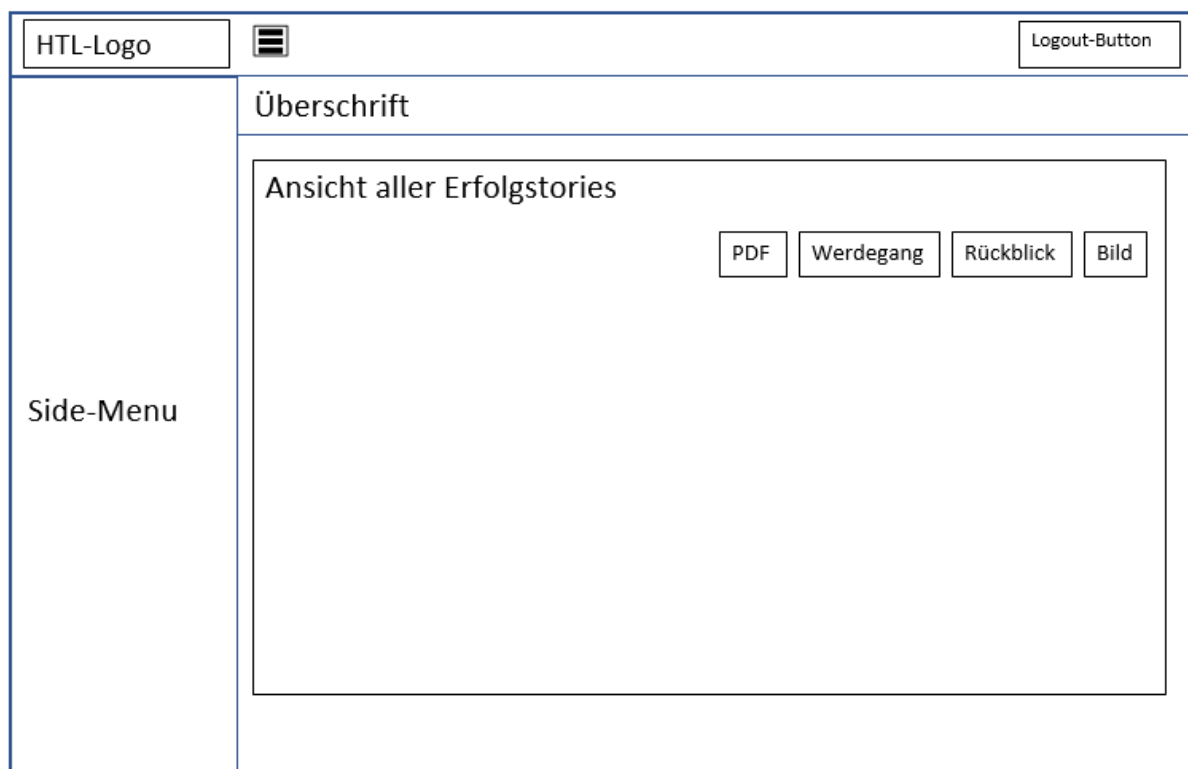
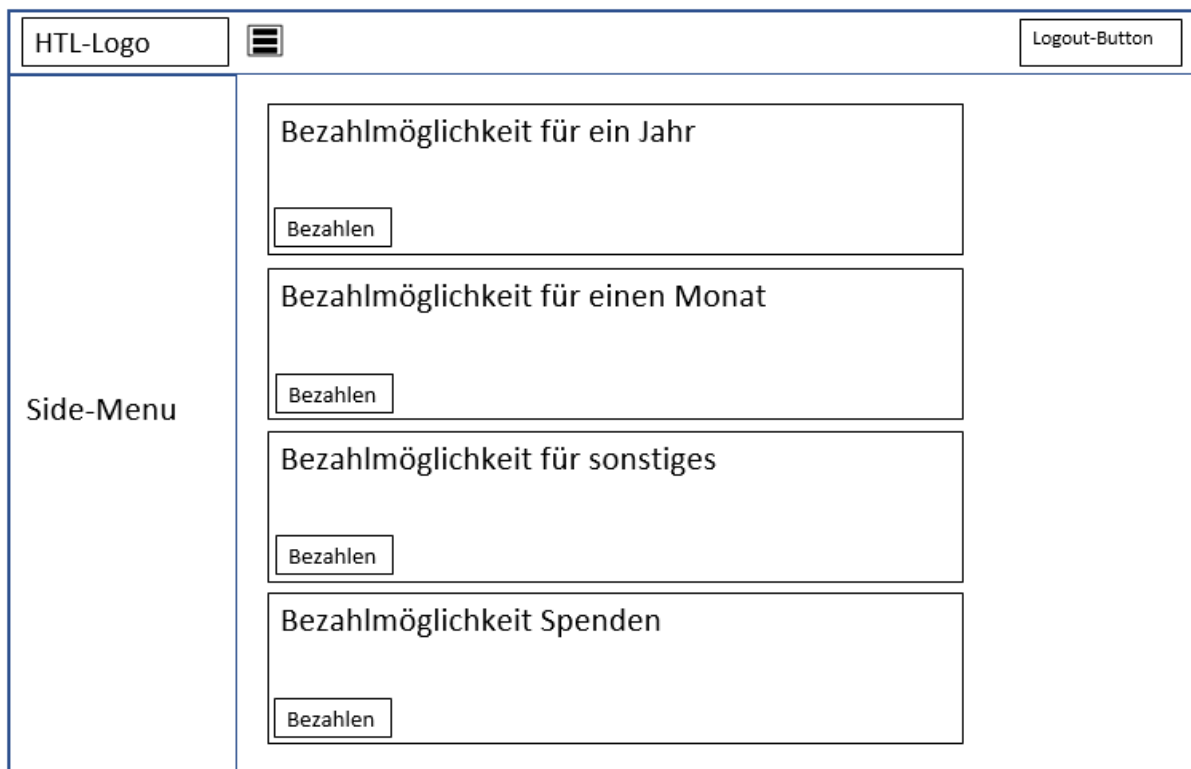


Abbildung 58: Webportal Entwurf Erfolgstories - Administratorfunktionen

Administratoren können alle Erfolgstories einsehen. Sie können sich die Informationen einzeln anzeigen lassen, das PDF in einem neuen Tab öffnen und speichern oder auch die Erfolgstory löschen. Weiters kann er mithilfe der zusätzlichen Buttons den jeweiligen Teil der Erfolgstory, welcher in der Beschriftung angedeutet wird, anzeigen.

11.9.15 Mitgliedsbeitrag



The image shows a wireframe of a web portal for membership contributions. At the top left is the HTL-Logo, followed by a hamburger menu icon. At the top right is a Logout-Button. On the left side, there is a vertical Side-Menu. The main content area contains four stacked boxes, each with a title and a 'Bezahlen' button below it:

- Bezahlungsmöglichkeit für ein Jahr
- Bezahlungsmöglichkeit für einen Monat
- Bezahlungsmöglichkeit für sonstiges
- Bezahlungsmöglichkeit Spenden

Abbildung 59: Webportal Entwurf Mitgliedsbeitrag

Hier können Benutzer ihren Mitgliedsbeitrag einzahlen. Abgewickelt wird dies über PayPal. Man kann den Mitgliedsbeitrag entweder mittels einer automatischen, jährlichen Abbuchung - damit man sich in Zukunft nicht mehr um die Zahlung kümmern muss - oder einer einmaligen Zahlung für ein Jahr begleichen. Weiters kann man auch sonstige Beträge einzahlen bzw. Spenden an den Absolventenverband senden. Über die „Bezahlen“-Buttons wird man direkt auf die jeweilige PayPal-Seite weitergeleitet, wo die Bezahlung bequem abgewickelt wird.

12 Implementierung

Im Kapitel Implementierung wird die Programmierung der Diplomarbeit in Teilschritte zerlegt und dargestellt. Dabei unterscheiden wir zwischen dem Backend (zur Datenaufbereitung und zur Datenhaltung) und dem Frontend (zur Anzeige und der Interaktion mit den Benutzern).

Im späteren Verlauf werden einige interessante Teilprobleme aufgegriffen und erläutert.

12.1 Backend

12.1.1 Datenmodell

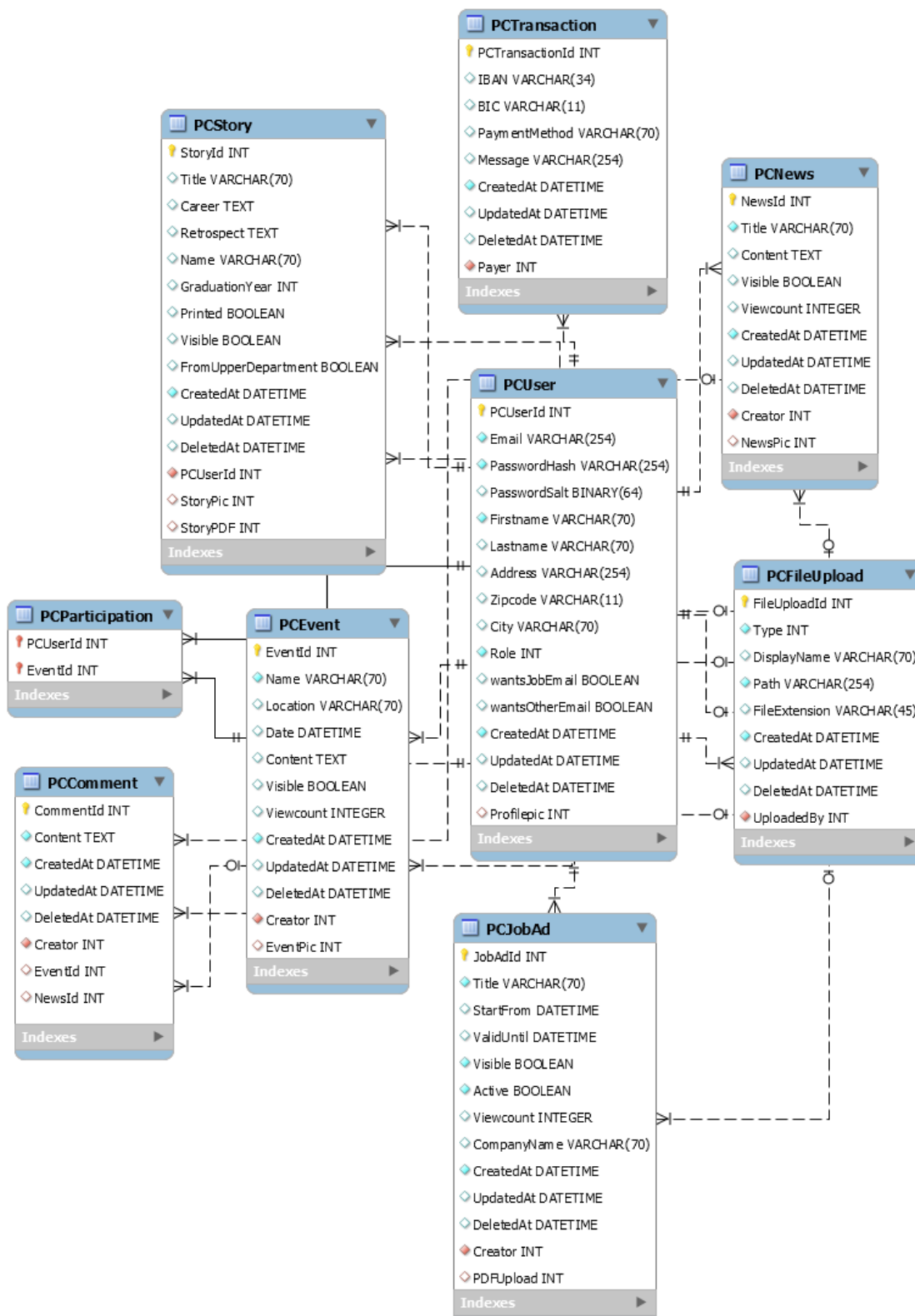


Abbildung 60: Datenmodell

12.1.2 Allgemeine Informationen

Im Folgenden werden einige kurze Informationen erläutert, die alle verwendeten Tabellen mit Ausnahme der Beziehungstabelle PCParticipation betreffen. Das Präfix „PC“ vor jedem Tabellennamen steht für „P@BS Connect“ und verhindert einerseits die Verwendung von Schlüsselwörtern in MySQL und Java und grenzt andererseits die Tabellen von den Joomla-Tabellen in der Datenbank ab.

Primärschlüssel

Alle Tabellen verfügen über einen Primärschlüssel mit der Bezeichnung <Entität>Id vom Typ INT, der beim Einfügen eines neuen Datensatzes automatisch inkrementiert wird.

Attribut *CreatedAt*

Jede Tabelle verfügt über das Attribut CreatedAt, welches angibt, wann ein Datensatz erstellt wurde.

Dieses Attribut kann beispielsweise dazu verwendet werden, Kommentare für eine News nach ihrem Erstellungsdatum gereiht anzuzeigen, um so eine chronologische Abfolge zu gewährleisten.

Attribut *UpdatedAt*

Im Attribut UpdatedAt lässt sich jeweils das letzte Änderungsdatum eines Datensatzes in jeder Tabelle ablesen (sofern eine Änderung zumindest einmalig durchgeführt wurde), um Änderungen terminlich nachverfolgen zu können.

Attribut *DeletedAt*

Bei einem Löschvorgang im System werden keine Datensätze physikalisch gelöscht. Es wird auf das Konzept der „soft-deletes“ gesetzt, das heißt es wird ein Attribut gesetzt, dass der Anwendung anzeigt, dass ein Datensatz als gelöscht anzusehen ist und somit dem Benutzer nicht präsentiert wird.

Das Attribut DeletedAt gibt den Zeitstempel der Löschung eines Datensatzes an. Ist es gesetzt, so wird dem Frontend der Datensatz nicht zur Verarbeitung und Anzeige übergeben.

Fremdschlüsselbeziehung zu PCFileUpload

Die meisten Entitäten verfügen über eine Fremdschlüsselbeziehung zur Tabelle PCFileUpload. Damit wird der Anhang einer Datei an den Datensatz realisiert. Die Dateien werden nicht direkt in der Datenbank gespeichert, sondern ganz gewöhnlich am Dateisystem des Servers abgelegt. Durch den Verweis des PCFileUpload-Datensatzes kann darauf in einer Entität zugegriffen werden.

12.1.3 Tabellenbeschreibungen

Im Folgenden werden die einzelnen Tabellen mit ihrer Funktion und den zugehörigen Attributen kurz erläutert. Ständig wiederkehrende Teile wie der Primärschlüssel oder die eben genannten Zeitstempel-Attribute werden nicht mehr erwähnt.

PCUser

In der Tabelle PCUser werden alle Benutzer von P@BS Connect abgespeichert. Dies umfasst Administratoren, Absolventen und Firmen. Die Tabelle kann als zentral angesehen werden und verfügt über Beziehungen zu sämtlichen anderen Tabellen.

Attribute

Name	Typ	Beschreibung
Email	VARCHAR (254)	Email-Adresse des Benutzers Muss eindeutig sein, es ist nicht möglich, dieselbe Adresse für zwei Benutzer zu verwenden
PasswordHash	VARCHAR (254)	Der „verhashte“ Wert von Passwort und Salt des Benutzers Mehr dazu in „Passwort-Hashing“
PasswordSalt	BINARY (64)	Zufällig generierter Salt zum Hashen des Passworts Mehr dazu in „Passwort-Hashing“
Firstname	VARCHAR (70)	Vorname eines Absolventen oder Administrators oder Firmenname einer Firma des Absolventenverbandes

Lastname	VARCHAR (70)	Nachname eines Absolventen oder Administrators oder Name des Ansprechpartners einer Firma des Absolventenverbandes
Address	VARCHAR (254)	Straße und Hausnummer
Zipcode	VARCHAR (11)	Postleitzahl
City	VARCHAR (70)	Ort
Role	INT	Rolle: 1 = Administrator 2 = Firma 3 = Absolvent

wantsJobEmail	BOOLEAN	Gibt Auskunft darüber, ob ein Absolvent die optionalen Benachrichtigungen zu Stellenangeboten per E-Mail erhalten möchte
wantsOtherEmail	BOOLEAN	Gibt Auskunft darüber, ob ein Absolvent die optionalen Benachrichtigungen zu neuen News und Events per E-Mail erhalten möchte
Profilepic	INT	Verweis auf das optionale Profilbild des Benutzers Funktion für künftige Entwicklungen, aktuell im Frontend nicht aktiv in Verwendung

Tabelle 2: Attribute PCUser

PCNews

In der Tabelle PCNews werden alle angelegten News-Beiträge abgespeichert.

Attribute

Name	Typ	Beschreibung
Title	VARCHAR (70)	Titel des News-Beitrags
Content	TEXT	Inhalt des Beitrags als HTML-Text Der Datentyp erlaubt die Speicherung großer Eingabemengen für lange Texte
Visible	BOOLEAN	Gibt an, ob der Beitrag für Benutzer in der Webanwendung sichtbar sein soll
Viewcount	INT	Anzahl der Aufrufe eines Beitrags Wird bei jedem Aufruf eines Benutzers erhöht und aktualisiert
Creator	INT	Verweis auf den PCUser, der den Beitrag erstellt hat (ein Administrator)
NewsPic	INT	Verweis auf das optionale Beitragsbild

Tabelle 3: Attribute PCNews

PCEvent

In der Tabelle PCEvent werden alle angelegten Events (Veranstaltungen) abgespeichert.

Attribute

Name	Typ	Beschreibung
Name	VARCHAR (70)	Name der Veranstaltung
Location	VARCHAR (70)	Ort der Veranstaltung
Date	DATETIME	Datum und Beginn-Uhrzeit der Veranstaltung

Content	TEXT	Inhalt des Veranstaltungstexts als HTML-Text Der Datentyp erlaubt die Speicherung großer Eingabemengen für lange Texte
Visible	BOOLEAN	Gibt an, ob die Veranstaltung für Benutzer in der Webanwendung sichtbar sein soll
Viewcount	INT	Anzahl der Aufrufe einer Veranstaltung Wird bei jedem Aufruf eines Benutzers erhöht und aktualisiert
Creator	INT	Verweis auf den PCUser, der die Veranstaltung angelegt hat (ein Administrator)
EventPic	INT	Verweis auf das optionale Veranstaltungsbild

Tabelle 4: Attribute PCEvent

PCComment

In der Tabelle PCComment werden alle angelegten Kommentare zu News-Beiträgen und Events abgespeichert.

Attribute

Name	Typ	Beschreibung
Content	TEXT	Inhalt des Kommentars als statischer Text ohne aufwändige Formatierung Der Datentyp erlaubt die Speicherung großer Eingabemengen für lange Texte
Creator	INT	Verweis auf den PCUser, der den Kommentar erstellt hat
EventId	INT	Verweis auf die zugehörige News, zu welcher der Kommentar erstellt wurde (sofern vorhanden)

NewsId	INT	Verweis auf die zugehörige Veranstaltung, zu welcher der Kommentar erstellt wurde (sofern vorhanden)
---------------	-----	--

Tabelle 5: Attribute PCComment

PCParticipation

PCParticipation ist eine Beziehungstabelle, welche die m:n-Relation zwischen PCUser und PCEvent abbildet. Sie wird dazu verwendet, die Teilnehmer einer Veranstaltung zu speichern und zu verwalten.

Die Primärschlüssel der beiden Tabellen werden hier verwendet, um die Zugehörigkeit (Teilnahme) eines Benutzers zu einer Veranstaltung abzubilden.

An einer Veranstaltung können somit viele Nutzer teilnehmen – und ein Nutzer kann an vielen Veranstaltungen teilnehmen.

Attribute

Name	Typ	Beschreibung
PCUserId	INT	Ein Primärschlüssel aus der Tabelle PCUser, der den Benutzer des Datensatzes als Teilnehmer angibt
PCEventId	INT	Ein Primärschlüssel aus der Tabelle PCEvent, der die Veranstaltung angibt, an welcher teilgenommen wird

Tabelle 6: Attribute PCParticipation

PCJobAd

In der Tabelle PCJobAd werden alle angelegten Stellenangebote abgespeichert.

Attribute

Name	Typ	Beschreibung
Title	VARCHAR (70)	Titel des Stellenangebots
StartFrom	DATETIME	Gültigkeitsbeginn der Stellenausschreibung
ValidUntil	DATETIME	Ende der Gültigkeit
Visible	BOOLEAN	Gibt an, ob das Stellenangebot für Benutzer in der Webanwendung sichtbar sein soll
Active	BOOLEAN	Gibt an, ob eine Benachrichtigung für das Stellenangebot noch aktiv ist, oder ob bereits eine E-Mail an interessierte Absolventen gesendet wurde
Viewcount	INT	Anzahl der Aufrufe eines Stellenangebots (für Administratoren und die Firma einsehbar, die das Angebot erstellt hat) Wird bei jedem Aufruf eines Benutzers erhöht und aktualisiert
CompanyName	VARCHAR (70)	Firmenname für die Anzeige beim Stellenangebot
Creator	INT	Verweis auf den PCUser, der die Veranstaltung angelegt hat (Firma oder Administrator)
PDFUpload	INT	Verweis auf die zugehörige PDF-Datei, in der die ausformulierte, firmengefertigte Stellenbeschreibung angegeben ist

PCStory

In der Tabelle PCStory werden alle angelegten HTL Perg-Erfolgsstories abgespeichert.

Attribute

Name	Typ	Beschreibung
Title	VARCHAR (70)	Titel der Erfolgsstory
Career	TEXT	Werdegang eines Absolventen Der Datentyp erlaubt die Speicherung großer Eingabemengen für lange Texte
Retrospect	TEXT	Rückblickende Aussagen eines Absolventen über die Ausbildungszeit an der HTL Perg Der Datentyp erlaubt die Speicherung großer Eingabemengen für lange Texte
Name	VARCHAR (70)	Name des Absolventen
GraduationYear	INT	Abschlussjahr
Printed	BOOLEAN	Gibt Auskunft darüber, ob die Erfolgsstory bereits ausgedruckt und in der HTL Perg ausgestellt wurde
Visible	BOOLEAN	Gibt an, ob die Erfolgsstory in der Webanwendung sichtbar sein soll
FromUpperDepartment	BOOLEAN	Gibt an, ob ein Benutzer die Höhere Abteilung oder die Fachschule der HTL Perg absolviert hat
PCUserId	INT	Verweis auf den PCUser (Absolvent), der die Erfolgsstory über sich verfasst hat
StoryPic	INT	Verweis auf das Bild des Absolventen
StoryPDF	INT	Verweis auf die PDF-Datei, die automatisch aus den eingegebenen Daten generiert wird

Tabelle 7: Attribute PCStory

PCFileUpload

In der Tabelle PCFileUpload werden Informationen über hochgeladene und gespeicherte Dateien abgelegt.

Attribute

Name	Typ	Beschreibung
Type	INT	Entitätstyp, für welchen die Datei hochgeladen wurde 1 = PCJobAd 2 = PCUser, PCStory (Bilder) 3 = PCNews 4 = PCEvent
DisplayName	VARCHAR (70)	Anzeigename der Datei
Path	VARCHAR (254)	Pfad zur Datei am Dateisystem des Servers
FileExtension	VARCHAR (45)	Dateiendung
UploadedBy	INT	Verweis auf den PCUser, der die Datei hochgeladen hat

Tabelle 8: Attribute PCFileUpload

PCTransaction

In der Tabelle PCTransaction werden Informationen zu Mitgliedsbeitragszahlungen gehalten.

Die Funktionalität ist im Backend für künftige Erweiterungen vorgesehen und wird derzeit vom Frontend nicht aktiv verwendet, da aktuell nur PayPal als Zahlungsmittel verwendet werden kann und hierzu keine Datenspeicherung von Nöten ist.

Attribute

Name	Typ	Beschreibung
IBAN	VARCHAR (34)	IBAN des Benutzers
BIC	VARCHAR (11)	BIC des Benutzers
PaymentMethod	VARCHAR (70)	Verwendete Bezahlmethode

Message	VARCHAR (254)	Überweisungs- / Transaktionstext
Payer	INT	Verweis auf den PCUser, der die Bezahlung angewiesen hat

Tabelle 9: Attribute PCTransaction

12.1.4 Datenbankoperationen

Im Folgenden werden die Möglichkeiten des Zugriffs (Operationen) auf die Datenbank sehr kurz erläutert.

Select

Mit einer Select-Operation können Daten aus der Datenbank ausgelesen werden. Durch die Angabe von Parametern kann die Auswahl gefiltert und über mehrere Tabellen angewandt werden.

Insert

Mit einem Insert können neue Datensätze in eine Datenbanktabelle eingefügt werden. Es ist auf die im Datenmodell festgelegten Regelungen zu achten, die den Datentyp, erlaubte Werte sowie Abhängigkeiten zu anderen Tabellen definieren.

Update

Mittels Update können bestehende Datensätze abgeändert und aktualisiert werden. Auch hier ist auf die Konsistenz und die Einhaltung definierter Regeln zu achten, damit die Manipulation erfolgreich ist.

Delete

Mit der Delete-Operation können Datensätze gelöscht werden und stehen fortan nicht mehr zur Verfügung. Es ist auf die Abhängigkeiten zu anderen Tabellen zu achten, die eine verkettete Löschung bewirken können.

12.1.5 REST-API

Die Dokumentation der gesamten REST-API mit ihren Schnittstellen, Anwendungsbeschreibungen und den zu erwartenden Antworten kann im Anhang nachgelesen werden.

Grundsätzlich ist die API dazu da, http-Aufrufe von der Clientanwendung entgegenzunehmen und daraus resultierend Datenbankoperationen durchzuführen oder Daten für den Client aufzubereiten.

Über die API kann das Frontend also Daten erhalten und auch senden, um so die Geschäftslogik abzubilden und zu realisieren.

Im Folgenden werden die oft eingesetzten Standardoperationen angeführt.

GET all

Mit der http-Methode GET können Daten abgefragt werden. *GET all* liefert hierbei sämtliche Einträge zu einer Entität.

GET by Id

Mittels *GET by Id* werden die Daten eines einzelnen Objekts / Datensatzes aufgrund seiner identifizierenden Id abgefragt.

POST

Mit der *POST*-Operation werden Daten an den Server gesendet, um eine neue Entität mit den übergebenen Daten anzulegen und zu persistieren.

PUT

Die *PUT*-Operation wird für Update-Vorgänge eingesetzt, um eine Entität aufgrund von übergebenen Daten zu aktualisieren.

DELETE

Mit einem *DELETE*-Aufruf wird die Löschung eines Objekts bzw. Datensatzes gefordert.

12.1.6 Statuscodes

Jeder API-Aufruf wird - neben den Nutzdaten je nach Operation (sofern zutreffend) – mit einem http-Statuscode und zusätzlich in manchen Fällen mit einem Beschreibungstext beantwortet.

Oftmals verwendete Statuscodes (Auszug)

Code	Bedeutung
200	OK Die Ausführung war erfolgreich, die zu erwartenden Daten stehen bereit und wurden übermittelt (sofern zutreffend)
201	Created Ein Objekt / Datensatz wurde erfolgreich angelegt
400	Bad Request Die Anfrage ist fehlerhaft (z.B. ungültige Nutzdaten)
401	Unauthorized Der Aufrufende konnte sich nicht ausreichend authentifizieren, eine Antwort wird aus Sicherheitsgründen verweigert
403	Forbidden Der Client besitzt keine Rechte, auf diese Ressource zuzugreifen (eventuell wurde eine als https konfigurierte URL per http aufgerufen)
404	Not Found Die gewünschte Ressource konnte nicht gefunden werden (hier ist kein Fehlen von Daten gemeint, sondern die aufgerufene Ressource ist nicht existent)
405	Method not allowed Die benutzte http-Methode ist für diesen Aufruf ungültig
415	Unsupported Media Type Der Anfrageninhalt wurde mit einem unerlaubten Mediatyp übermittelt
500	Internal Server Error Sammel-Statuscode für nicht erwartete Serverfehler

Tabelle 10: Oftmals verwendete http-Statuscodes (vgl. wikipedia.org, 2017)

12.1.7 Zugriff auf die Datenbank

Zur Kommunikation mit der Datenbank wird mit Data Access Objects (DAO) gearbeitet. Eine Erklärung der Technologie ist im Grundlagen-Kapitel zu finden.

Ein generischer DaoManager implementiert den Zugriff mittels eines EntityManagers übergeordnet für alle Entitäten. Dies umfasst die Standardoperationen, die beinahe in jedem Service benötigt werden.

```
public abstract class DaoManager <T> implements DaoInterface<T> {
    private Class<T> type;

    @PersistenceContext
    protected EntityManager entityManager;

    public DaoManager(Class<T> type) {
        this.type=type;
    }

    @Override
    public T persist(T entity) {
        this.entityManager.persist(entity);
        this.entityManager.flush();

        return entity;
    }

    @Override
    public void remove(T entity) {
        this.entityManager.remove(this.entityManager.contains(entity)? entity: this.entityManager.merge(entity));
    }

    @Override
    public T findById(int id) {
        return this.entityManager.find(this.type, id);
    }

    @Override
    public T update(T entity) {
        return this.entityManager.merge(entity);
    }

    public List<T> findAll(){
        String queryName=this.type.getSimpleName()+".findAll";

        return (List<T>)entityManager.createNamedQuery(queryName, this.type).getResultList();
    }
}
```

Die einzelnen Entitäten erben von diesem DaoManager, nutzen dessen Methoden und implementieren gegebenenfalls zusätzliche Funktionen zum Datenbankzugriff, sofern benötigt.

```
@Named
@Stateless
public class PCUserDao extends DaoManager<PCUser> {
    public PCUserDao() {
        super(PCUser.class);
    }

    public PCUser findByEmail(String email){
        List<PCUser> pcusers = this.entityManager.createQuery("SE-
LECT u FROM PCUser u WHERE u.email = :email AND u.deletedAt
IS NULL").setParameter("email", email).getResultList();
        if(pcusers.size() == 0) {
            return null;
        } else {
            return pcusers.get(0);
        }
    }
}
```

12.1.8 Kommunikation per REST-API

Für jede einzelne Entität gibt es ein Service, welches REST-Anfragen entgegennimmt, die Daten aufbereitet und an den Anfragenden übermittelt.

Im folgenden Code-Ausschnitt werden beispielhaft einige Service-Methoden der Entität PCEvent dargestellt, die als REST-Endpunkte aufrufbar sind.

```
@Path("/event")
@Consumes(MediaType.APPLICATION_JSON)
@Produces(MediaType.APPLICATION_JSON)
public class PCEventService {
    @Inject
    private PCEventDao pceventDao;
    @Inject
    private PCCommentDao pccommentDao;
    @Inject
    private PCUserDao;

    @GET
    @Secured
    public List<PCEvent> getAll() {
        return this.pceventDao.findAll();
    }

    @POST
    @Secured
    public Response createEvent(PCEvent pcevent) {
        try {
            pcevent.setCreatedAt(new Date());
            this.pceventDao.persist(pcevent);
        } catch (javax.persistence.PersistenceException e) {
            return Response.status(400).type("text/plain").entity("Fehler bei Event: " + pcevent.getName()).build();
        }

        try {
            new Thread(new Runnable() {
                public void run() {
                    notificationNewEvent(pcevent);
                }
            }).start();
        } catch (Exception e) {
            e.printStackTrace();
            return Response.status(500).type("text/plain").entity("Email zu Event: " + pcevent.getName() + " konnte nicht gesendet werden").build();
        }

        return Response.status(200).entity(pcevent).build();
    }
}
```

```
@PUT
@Path("/{id}/participation/{pcuserId}")
@Secured
public Response participate(@PathParam("id") int id, @PathParam("pcuserId") int pcuserId) {
    PCEvent e = this.pceventDao.findById(id);
    PCUser u = this.pcuserDao.findById(pcuserId);

    if(e == null || u == null) {
        return Response.status(404).build();
    }

    if(!u.getPevents2().contains(e)) {
        u.getPevents2().add(e);
        this.pcuserDao.update(u);
    }

    return Response.status(200).build();
}

@POST
@Path("/{id}/comment")
@Secured
public Response addComment(@PathParam("id") int id, PCComment pccomment) {
    PCEvent e = this.pceventDao.findById(id);

    if(e == null) {
        return Response.status(404).build();
    }

    try {
        pccomment.setCreatedAt(new Date());
        pccomment.setPcevent(e);
        this.pccommentDao.persist(pccomment);
        e.addPccomment(pccomment);
        this.pceventDao.update(e);
    } catch (javax.persistence.PersistenceException x) {
        return Response.status(400).type("text/plain").entity("Fehler beim Hinzufügen von Kommentar mit Datum: " + pccomment.getCreatedAt() + " für Event: " + e.getName()).build();
    }

    return Response.status(200).build();
}
```

12.1.9 Authentifizierung

Damit sich Benutzer in der Webanwendung anmelden können, ist eine serverseitige Authentifizierung notwendig.

Dies wird in der Diplomarbeit mittels JSON Web Tokens realisiert, welche bereits im Kapitel Grundlagen erklärt wurden.

Beim Einloggen übermittelt der Client die eingegebenen Nutzerdaten des aktuellen Benutzers mit einem REST-Request und möchte sich damit authentifizieren. Die Daten werden am Server geprüft und bei korrekter Eingabe wird ein JSON Web Token ausgestellt und retourniert.

JSON Web Token

Der folgende Code-Ausschnitt zeigt die Authentifizierung eines Benutzers aufgrund seiner E-Mail und seines Passworts. Aufgrund der Daten wird ein zeitlich begrenzter JWT erstellt, der dann an den Client gesendet wird und für künftige Anfragen verwendbar ist.

```
@Inject
private PCUserDao pcuserDao;

@POST
@Path("/login")
@Consumes("application/x-www-form-urlencoded")
public Response authenticatePCUser(@FormParam("email") String email, @Form-
Param("password") String password) {
    long ttl = 3600000;

    PCUser pcuser;
    String jwt;

    try {
        pcuser = authenticate(email, password);
        jwt = generateJWT(Integer.toString(pcuser.getPCUserId()),
            "pabs.htl-perg.ac.at", email, pcuser.getRole(), ttl);
    } catch(Exception e) {
        return Response.status(Response.Status.UNAUTHOR-
            IZED).build();
    }

    long exp = System.currentTimeMillis() + ttl;
    return Response.status(200).entity(pcuser).header("JWT",
        jwt).header("ValidUntil", new Date(exp)).build();
}
```

```
private PCUser authenticate(String email, String password) throws Exception
{
    PCUser u;

    u=pcuserDao.findByEmailAndPassword(email, password);

    if(u == null) {
        throw new Exception();
    }

    return u;
}
```

```
private String generateJWT(String id, String issuer, String subject, int
role, long ttlMillis) {
    SecretKey key = new SecretKey();
    Key signKey=key.getKey();

    String roleToAssign = "";
    if(role == 1) {
        roleToAssign = "Admin";
    } else {
        if(role == 2) {
            roleToAssign = "Firma";
        } else {
            if(role == 3) {
                roleToAssign = "Absolvent";
            }
        }
    }

    long nowMillis=System.currentTimeMillis();
    Date now=new Date(nowMillis);

    JwtBuilder builder=Jwts.builder().setId(id)
        .setIssuedAt(now)
        .setSubject(subject)
        .setIssuer(issuer)
        .claim("Role", roleToAssign)
        .signWith(SignatureAlgo-
rithm.HS256,signKey);

    if(ttlMillis >= 0) {
        long expMillis=nowMillis + ttlMillis;
        Date exp = new Date(expMillis);
        builder.setExpiration(exp);
    }
    return builder.compact();
}
```

Authentifizierungsfilter

REST-Endpunkte, die vor unautorisiertem Zugriff zu schützen sind, werden mit einer eigens erstellten Secured-Annotation versehen. Ist diese Annotation auf eine Methode gesetzt, so kann diese Methode erst dann ausgeführt werden, wenn ein Authentifizierungsfilter passiert wurde, der den vom Client übermittelten JWT auf seine Gültigkeit kontrolliert.

Der folgende Code-Ausschnitt zeigt einen Teil des Authentifizierungsfilters, der zuerst überprüft, ob ein JSON Web Token mit der Client-Anfrage versendet wurde. Ist dies der Fall, wird der übermittelte Token kontrolliert. Wenn alles in Ordnung ist, wird die Anfrage weitergeleitet und die gewünschte Service-Methode ausgeführt, damit die vom Client angefragten Daten an diesen gesendet werden können.

```
@Secured
@Provider
@Priority(Priorities.AUTHENTICATION)
public class AuthenticationFilter implements ContainerRequestFilter {

    @Override
    public void filter(ContainerRequestContext requestContext) throws IOException {
        String authorizationHeader=requestContext.getHeaderString(Headers.AUTHORIZATION);

        if(authorizationHeader==null || !authorizationHeader.startsWith("Bearer ")){
            throw new NotAuthorizedException("Authorization header must be provided");
        }

        String token=authorizationHeader.substring("Bearer ".length()).trim();

        try {
            String email=validate(token);
            SecurityContext securityContext=requestContext.getSecurityContext();
            Authorizer authorizer=new Authorizer(email, securityContext.isSecure());
            requestContext.setSecurityContext(authorizer);
        } catch (SignatureException e){
            requestContext.abortWith(Response.status(Response.Status.UNAUTHORIZED).entity("Sign wrong!").build());
        } catch (ExpiredJwtException e) {
            requestContext.abortWith(Response.status(Response.Status.UNAUTHORIZED).entity("JWT expired!").build());
        } catch (Exception e) {
            requestContext.abortWith(Response.status(Response.Status.UNAUTHORIZED).entity("Exception!").build());
        }
    }
}
```

```
}  
}
```

Secured-Annotation

Der folgende Code zeigt die eigens erstellte Annotation, die beispielsweise Methoden markiert, die schützenswert sind, damit der Authentifizierungsfilter aktiv wird.

```
@NameBinding  
@Retention(RetentionPolicy.RUNTIME)  
@Target({ElementType.TYPE, ElementType.METHOD})  
public @interface Secured { }
```

12.1.10 Passwort-Hashing

Um die Sicherheit des Systems zu erhöhen, werden Passwörter nicht im Klartext abgelegt, sondern als Hash gespeichert.

Hash

Auf das eingegebene Passwort eines Benutzers bei der Registrierung wird eine kryptologische Hashfunktion angewandt. Dadurch entsteht der Hash des Passworts, der anstelle des Klartexts abgespeichert wird und somit nicht mehr einfach auszulesen ist.

Übermittelt der Benutzer beim Login später wiederum sein Passwort, so wird auch daraus der Hash mit derselben Funktion errechnet und mit dem in der Datenbank gespeicherten Wert verglichen. Sind die Werte ident, so gilt die Eingabe als gültig und der Benutzer wird ins System gelassen.

Salt

Um eine erhöhte Sicherheit zu gewährleisten, wird der Hash nicht nur aus dem übermittelten Passwort errechnet. Zusätzlich wird eine zufällig generierte Zeichenfolge, der sogenannte Salt, an das Passwort gehängt, und das Gesamtkonstrukt wird an die Hashfunktion übergeben.

Hashfunktionen erzeugen aus identen Passwörtern idente Hashes. Wenn also ein Angreifer Zugriff auf die Hashes bekommt und dort viele gleiche Werte vorfindet, kann er darauf schließen, dass diese Benutzer alle dasselbe Passwort verwendet

haben und oftmals verwendete Kennwörter einfach ausprobieren, bis er das korrekte gefunden hat.

Ein Salt erschwert das Knacken des Passworts mithilfe einer Rainbow-Tabelle (Tabelle mit sehr vielen häufig verwendeten Passwörtern, die für eine Brute-Force-Attacke verwendet werden), da das bloße Auslesen und eine Analyse der Hash-Werte nicht ausreichen, um auf das Klartext-Passwort zu schließen. Durch den für jeden Nutzer unterschiedlichen, zufälligen Salt ist der Hash-Wert auch bei identen Passwörtern unterschiedlich. (vgl. wikipedia.org, 2017)

Implementierung

Der folgende Code-Ausschnitt zeigt die Funktionen zum Hashing von Passwörtern am Server. Hierzu wird die Funktion PBKDF2 (Password-Based Key Derivation Function 2) mithilfe eines SHA512-Algorithmus verwendet. Es wird mithilfe eines 512 Bit langen Schlüssels in 1000 Iterationen kodiert, was dem aktuellen Sicherheitsstandard entspricht.

```
private static final String ALGORITHM="PBKDF2WithHmacSHA512";
private static final int ITERATIONS=1000;
private static final int KEY_LENGTH=512;

public static Object[] hash(String password) throws NoSuchAlgorithmException, InvalidKeySpecException {
    byte[] salt = generateSalt();
    byte[] hashedPassword = hashPassword(password, salt);

    Object[] object=new Object[2];
    object[0]=hashedPassword;
    object[1]=salt;

    return object;
}

private static byte[] hashPassword(String password, byte[] salt)
throws NoSuchAlgorithmException, InvalidKeySpecException {
    SecretKeyFactory secretKeyFactory = SecretKeyFactory.getInstance(ALGORITHM);

    PBEKeySpec spec = new PBEKeySpec(password.toCharArray(), salt, ITERATIONS,KEY_LENGTH);

    SecretKey secretKey = secretKeyFactory.generateSecret(spec);
    byte[] res = secretKey.getEncoded();

    return res;
}
```

```
private static byte[] generateSalt() throws NoSuchAlgorithmException {
    SecureRandom random = SecureRandom.getInstance("SHA1PRNG");

    byte[] salt=new byte[8];
    random.nextBytes(salt);

    return salt;
}
```

12.1.11 Generierung der HTL-Erfolgsstories

Die Generierung der PDF-Dateien erfolgt mithilfe der iText-Bibliothek. Als erstes müssen dafür alle benötigten Elemente der Bibliothek importiert werden, um diese verwenden zu können:

```
import com.itextpdf.text.BadElementException;
import com.itextpdf.text.BaseColor;
import com.itextpdf.text.Document;
import com.itextpdf.text.DocumentException;
import com.itextpdf.text.Element;
import com.itextpdf.text.Font;
import com.itextpdf.text.Image;
import com.itextpdf.text.PageSize;
import com.itextpdf.text.Paragraph;
import com.itextpdf.text.Utilities;
import com.itextpdf.text.pdf.BaseFont;
import com.itextpdf.text.pdf.PdfPCell;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
```

PDF-Dokument erzeugen und Parameter für den PDF-Export einstellen

Zu den Parametern zählen der Speicherort, der Dateiname und die Seitengröße. Der Dateiname beinhaltet immer das aktuelle Datum (inklusive Uhrzeit) im Format `_yyyy_MM_dd_HH_mm_ss`, damit ein eindeutiger Dateiname gewährleistet ist.

```
DateFormat dateFormat = new SimpleDateFormat("_yyyy_MM_dd_HH_mm_ss");
Date date = new Date();
String filename=
this.baseUrl+"story/"+uploadedBy+dateFormat.format(date)+"_"+successstory.
pdf";
```

```
Document document = new Document();
PdfWriter.getInstance(document, new FileOutputStream(filename));
```

```
document.open();
document.setPageSize(PageSize.A3);
document.setPageSize(document.getPageSize().rotate());
document.newPage();
```

Aufteilung der DIN A3 Seite

Ein weiterer wichtiger Schritt ist die Aufteilung der Seite in einzelne Sektoren. Die Grundlage dafür ist der nachfolgende Entwurf einer HTL-Erfolgsstory.



Abbildung 61: Entwurf HTL-Erfolgsstory

Für die Aufteilung der Seite werden sogenannte „PDFTables“ der iText-Bibliothek verwendet. Die Gesamtbreite beträgt 400 Millimeter. Da iText mit Bildpunkten arbeitet, müssen diese mit einer geeigneten Methode umgerechnet werden. Zusätzlich sollen die einzelnen Zeilen keinen Rahmen haben.

Jeder einzelne Bereich der Skizze bildet eine eigene Zelle. In der jeweiligen Tabellenzelle werden danach in eigenen Methoden die Zelleninhalte zusammengestellt. Die Parameter der Methoden sind immer die Zellenbreite und Länge anteilmäßig an der Gesamtlänge und Breite der Tabelle und eventuell benötigte Informationen für die Zellen.

```
PdfPTable table = new PdfPTable(200);
table.setTotalWidth(Utilities.millimetersToPoints(400));
table.setLockedWidth(true);
table.getDefaultCell().setBorder(0);
```

```
//Streifen (grün oder gelb)
table.addCell(getCell_green_or_yellow_bar(2,75,type));
//Abstand - Zelle ohne Inhalt
table.addCell(getCell_white(1,75));
//HTL Logo
table.addCell(getCell_htl_logo(30,75,this.baseUrl+"general/htlperglogo.png"
));

//Überschrift
table.addCell(getCell_header(60,75,type));
//Abstand - Zelle mit grauem Hintergrund
table.addCell(getCell_gray(107,75));
//Abstand - Zelle ohne Inhalt
table.addCell(getCell_white(200,10));
//Bild
table.addCell(getCell_pic(60,490,path_picture));
//Daten (Name, Schultyp, Abschlussjahr, Werdegang)
table.addCell(getCell_data(140,490,type,year,name.toUpperCase(),story));
//Abstand - Zelle ohne Inhalt
table.addCell(getCell_white(200,10));
//Rückblickende Aussage
table.addCell(getCell_review(200, 185,review));

table.completeRow();
document.add(table);

document.close();
```

In den nächsten Unterpunkten wird die Gestaltung der einzelnen Zellen auszugsweise näher erläutert. Es werden aber nicht alle einzeln beschrieben, da diese teilweise relativ ähnlich sind.

Zelle mit dem Bild der HTL-Erfolgsstory

Als erstes müssen allgemeine Einstellungen an der Zelle vorgenommen werden, wie beispielsweise die Zellenbreite und -höhe. Danach wird das Bild, welches bereits vorher am Server (Pfad zum Bild ist in der Variable `pic_path` hinterlegt) gespeichert wurde, mittig in der Zelle eingefügt.

```
private PdfPCell getCell_pic(int cells, int height, String pic_path) throws
IOException, BadElementException {

    PdfPCell cell = new PdfPCell();
    cell.setColspan(cells);
    cell.setUseAscender(true);
    cell.setUseDescender(true);
    cell.setBorder(0);
    cell.setFixedHeight(height);

    cell.setHorizontalAlignment(Element.ALIGN_CENTER);
    cell.setVerticalAlignment(Element.ALIGN_MIDDLE);

    Image image1 = Image.getInstance(pic_path);

    cell.addElement(image1);

    return cell;
}
```

Zelle mit dem grauen Hintergrund

Auch hier werden wieder allgemeine Einstellungen der Zelle vorgenommen. Zusätzlich wird die Hintergrundfarbe der Zelle auf den Grauton gesetzt.

```
private PdfPCell getCell_gray(int cells, int height) {

    PdfPCell cell = new PdfPCell();
    cell.setColspan(cells);
    cell.setUseAscender(true);
    cell.setUseDescender(true);
    cell.setBorder(0);
    cell.setFixedHeight(height);

    cell.setBackgroundColor(new BaseColor(192, 192, 192));

    return cell;
}
```

Zelle mit dem Namen

Wiederum werden allgemeine Einstellungen der Zelle vorgenommen. Für den Namen selbst muss die verwendete Schriftart „Calibri“ vom zugehörigen „.ttf“-File importiert werden. Danach wird noch die Schriftgröße, etc. festgelegt und der Text in einem neuen Paragraph eingefügt.

```
private PdfPCell getCell_name(int cells, int height, String name) throws
IOException, DocumentException {
    PdfPCell cell = new PdfPCell();
    cell.setColspan(cells);
    cell.setUseAscender(true);
    cell.setUseDescender(true);
```

```
cell.setBorder(0);
cell.setFixedHeight(height);
cell.setVerticalAlignment(Element.ALIGN_MIDDLE);

BaseFont calibri =
BaseFont.createFont(this.baseUrl+"general/Calibri.ttf",BaseFont.WIN
ANSI, BaseFont.NOT_EMBEDDED);

Font font = new Font(calibri, 30,Font.BOLD);

Paragraph p = new Paragraph(String.format(" "+name),font);

cell.addElement(p);

return cell;
}
```

12.1.12 Dateiupload

Für die PDF-Files der Stelleninserate, sowie für die Bilder der News, Events und HTL-Erfolgsgeschichten ist ein Dateiupload notwendig.

POST-Methode des Web-APIs

Zunächst werden die benötigten Parameter (Dateityp und die Id des Benutzers, der die Datei hochladen will) aus dem Body der http-POST Anfrage ausgelesen. Sollten diese im Body nicht gefunden werden, wird der ein Fehler mit http-Statuscode 400 (Bad Request) zurückgesendet.

Für die Dateitypen gelten folgende Regeln:

- 1 → PDF eines Jobinsetates
- 2 → Bilder der HTL-Erfolgsgeschichten und Profilbilder
- 3 → Bilder von News
- 4 → Bilder der Events

```
try{

    type = Integer.parseInt(input.getFormDataPart("type", String.class, null));

    uploadedBy = Integer.parseInt(input.getFormDataPart("uploadedBy",
String.class, null));
}catch(Exception e){
    return Response.status(Response.Status.BAD_REQUEST).build();
}
```

Ist das Auslesen erfolgreich wird die Methode zur Speicherung der Datei aufgerufen:

```
this.fileUpload.fileUpload(input_file, type, user);
```

Bestimmung des Speicherortes und des Dateinamens

Je nach Typ wird die Datei in einem anderen Ordner auf dem Server gespeichert. Für die Speicherung am Server wird ein eigener, eindeutiger Dateiname generiert.

Dafür ist die Bestimmung des ursprünglichen Dateinamens und der Dateiendung notwendig, da der Server aus sicherheitstechnischen Gründen nur Bilddateien und PDF-Dateien akzeptiert.

Dateinamenbestimmung

```
private String parseFileName(MultivaluedMap<String, String> headers) {  
  
    String[] contentDispositionHeader = headers.getFirst("Content-  
Disposition").split(";");  
  
    for (String name : contentDispositionHeader) {  
        if ((name.trim().startsWith("filename"))) {  
            String[] tmp = name.split("=");  
  
            String fileName = tmp[1].trim().replaceAll("\\\"", "");  
  
            return fileName;  
        }  
    }  
  
    return "randomName";  
}
```

Bestimmung der Dateinamenerweiterung

```
private String getFileExtension(String filename){  
    int i = filename.lastIndexOf('.');  
    if (i > 0) {  
        filename = filename.substring(i+1);  
    }  
  
    return filename;  
}
```

Speicherung der Datei am Server

Der wichtigste Schritt des Dateiuploads ist die Speicherung am Server. Dafür wird der Methode der notwendige InputStream (die zu speichernde Datei) und der vollständige Pfad (inklusive neuem Dateinamen) übergeben. Die Methode speichert dann mithilfe eines OutputStreams den InputStream am Server. Ist die Speicherung erfolgreich wird „true“ zurückgegeben. Sollte die Speicherung nicht erfolgreich sein, so wird „false“ zurückgegeben.

```
private boolean saveFile(InputStream uploadedInputStream, String
serverLocation) {

    File file=new File(serverLocation);

    try {
        OutputStream outputStream = new FileOutputStream(file);
        int read = 0;
        byte[] bytes = new byte[1024];

        outputStream = new FileOutputStream(new
File(serverLocation));
        while ((read = uploadedInputStream.read(bytes)) != -1) {
            outputStream.write(bytes, 0, read);
        }

        outputStream.flush();
        outputStream.close();

    } catch (IOException e) {

        return false;
    }

    return true;
}
```

12.1.13 Mail-System

Das Mail-System wird zur Benachrichtigung der Benutzer per E-Mail benötigt. Dafür ist es als erstes notwendig, eine sichere (verschlüsselte) Verbindung zum Mailserver einzurichten. Da eine Office 365 E-Mail-Adresse als Absenderadresse in der Diplomarbeit verwendet wird, ist eine Verbindung zum SMTP-Server von Microsoft notwendig. Dort ist eine Authentifizierung mit E-Mail und Passwort von Nöten:

```
Properties props = new Properties();
props.put("mail.smtp.host", "smtp.office365.com");
props.put("mail.smtp.socketFactory.port", "587");
props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory")
props.put("mail.smtp.auth", "true");
props.put("mail.smtp.port", "587");
props.put("mail.smtp.starttls.enable", "true");
```

```
Session session = Session.getInstance(props,
new javax.mail.Authenticator() {
    @Override
    protected PasswordAuthentication getPasswordAuthentication() {
        return new PasswordAuthentication("pabs_admin@htl-
perg.ac.at", "<Passwort>");
    }
});
```

Im nächsten Schritt wird schlussendlich die Mail versendet. Dafür werden die Absenderadresse und der Absendername festgelegt. Danach werden die Empfänger ausgewählt und der Betreff festgelegt. Zum Schluss fügt man noch den Body an und versendet die E-Mail.

```
try {
    Message message = new MimeMessage(session);
    message.setFrom(new InternetAddress("pabs_admin@htl-perg.ac.at", abs));

    message.addRecipients(Message.RecipientType.TO, InternetAddress.parse(
        "pabs_admin@htl-perg.ac.at"));

    for(int i=0; i<empf.size();i++){
        message.addRecipients(Message.RecipientType.BCC, InternetAddress
            .parse(empf.get(i)));
    }

    message.setSubject(betreff);

    message.setContent(text, "text/html; charset=utf-8");

    Transport.send(message);

} catch (MessagingException e) {
    System.out.println(e.getMessage());
} catch (UnsupportedEncodingException ex) {
    System.out.println(ex.getMessage());
}
```

12.2 Frontend

12.2.1 Login

Um das Webportal benutzen zu können, muss sich der User zuerst authentisieren. Dafür gibt er seine E-Mail-Adresse und sein Passwort ein. Bestätigt man seine Eingabe, wird im Hintergrund die Authentifizierung durchgeführt. Dafür wird ein http-POST-Request an den Server geschickt. Ist die Authentifizierung erfolgreich abgeschlossen, so werden die Nutzerdaten und der erhaltene Authentifizierungstoken lokal im Browser gespeichert.

```
var data: Login {
    email: value.email,
    password: value.password
}

this.loginService.authenticate(data).subscribe(response => {
    var payload = response.json();
    var headers = response.headers;

    var user: User = {

        address: payload.address,
        city: payload.city,
        email: payload.email,
        firstname: payload.firstname,
        passwordHash: payload.passwordHash,
        lastname: payload.lastname,
        role: payload.role,
        wantsJobEmail: payload.wantsJobEmail,
        wantsOtherEmail: payload.wantsOtherEmail,
        zipcode: payload.zipcode,
        profilePic: payload.zipcode,
        pcuserId: payload.pcuserId,
        token: headers.get('JWT'),
        token_validUntil: new Date(headers.get("ValidUntil"))
    }

    this.localStorage.set("user", JSON.stringify(user));
}
```

12.2.2 Dateiupload

Um eine Datei am Server zu speichern, wird ein http-POST-Request abgesetzt. Zur Authentifizierung wird im http-Header der aktuelle JSON-Web-Token des Benutzers eingefügt und um ein File hochladen zu können, muss das Headerfeld „Enctype“ noch auf den Wert „multipart/form-data“ gesetzt werden. Im Body werden die Datei selbst und weitere vom Server benötigte Informationen angefügt.

```
uploadFile(data:JobAdUpload,user: User){
    this.baseUrl=this.localStorage.get("baseUrl"+"pcfileupload");

    var headers = new Headers();
    headers.append('Authorization', 'Bearer '+user.token);
    headers.append('Enctype', 'multipart/form-data');

    var form_data = new FormData();

    form_data.append("file",data.jobad);
    form_data.append("uploadedBy",user.pcuserId)
    form_data.append("type",1);

    return this.http
        .post(this.baseUrl,form_data,{ headers: headers});
}
```

12.2.3 Mitgliedsbeitrag

Die Bezahlung des Mitgliedbeitrags wird mithilfe von PayPal abgewickelt. Die Voraussetzung dafür ist ein PayPal-Geschäftskonto. Nach der Erstellung eines Bezahlungslinks im Administrationsinterface von PayPal leitet man nur noch auf die Bezahlungs-Seite weiter. Der Zahlungsprozess an sich wird von PayPal abgewickelt.

12.2.4 CSV-Import von neuen Benutzern

Um mehrere Benutzer auf einmal in das System hinzufügen zu können, wird ein CSV-Import implementiert. Da der Übergabeparameter mit „any“ deklariert wurde, ist es möglich, der Methode eine CSV-Datei zu übergeben. Wird eine Datei über das Webinterface ausgewählt, so wird diese eingelesen und der Separator der CSV-Datei angeführt. Danach wird die Länge des resultierenden, mehrdimensionalen Ergebnisarrays geprüft. Sind Werte enthalten, so werden die Benutzer aus dem Ergebnisarray in der Datenbank gespeichert. Ist das Array allerdings leer, so wird eine

dementsprechende Meldung ausgeben. Dies geschieht mithilfe der „noteHelp“-Variable. Damit wird kann auch auf andere Fehler reagiert werden.

```
upload(inputValue:any):void {
  var data = null;
  var file:File = inputValue.files[0];
  var reader:FileReader = new FileReader();
  var that = this;
  var noteHelp: number;

  reader.onloadend = function (e) {
    var csvData = reader.result;
    data = $csv.toArrays(csvData, {separator:";"});

    if (data && data.length > 0) {
      var users = [];
      for(var i=0; i<data.length; i++)
        users.push(new User(data[i][0], data[i][1], data[i][2],
          data[i][3], data[i][4], data[i][5], data[i][6], data[i][7]));

      that.profilService.createUsers(users, that.user.token).subscribe(response => {
        noteHelp=1;
      }, error => {
        noteHelp=2;
      });
    } else {
      noteHelp = 3;
    }
  };

  reader.onerror = function () {
    noteHelp=4;
  };
  this.note=noteHelp;
  reader.readAsText(file);
}
```

13 Ergebnis

Das Ergebnis der Diplomarbeit P@BS Connect ist eine Webapplikation. Das Webportal kann auch auf mobilen Geräten genutzt werden, da das Frontend responsive gestaltet ist. Dadurch wird die Benutzerfreundlichkeit deutlich erhöht. Die Entwürfe für das Design des Webportals sind im Punkt Planung und Realisierung bereits angeführt worden.

13.1 Einstieg in Webportal

Das Webportal ist unter <http://pabs.htl-perg.ac.at> aufrufbar. Der informelle Teil des Webportals ist nun ersichtlich. Verschiedene Informationen zum Absolventenverband können in mehreren Menüpunkten eingesehen werden.

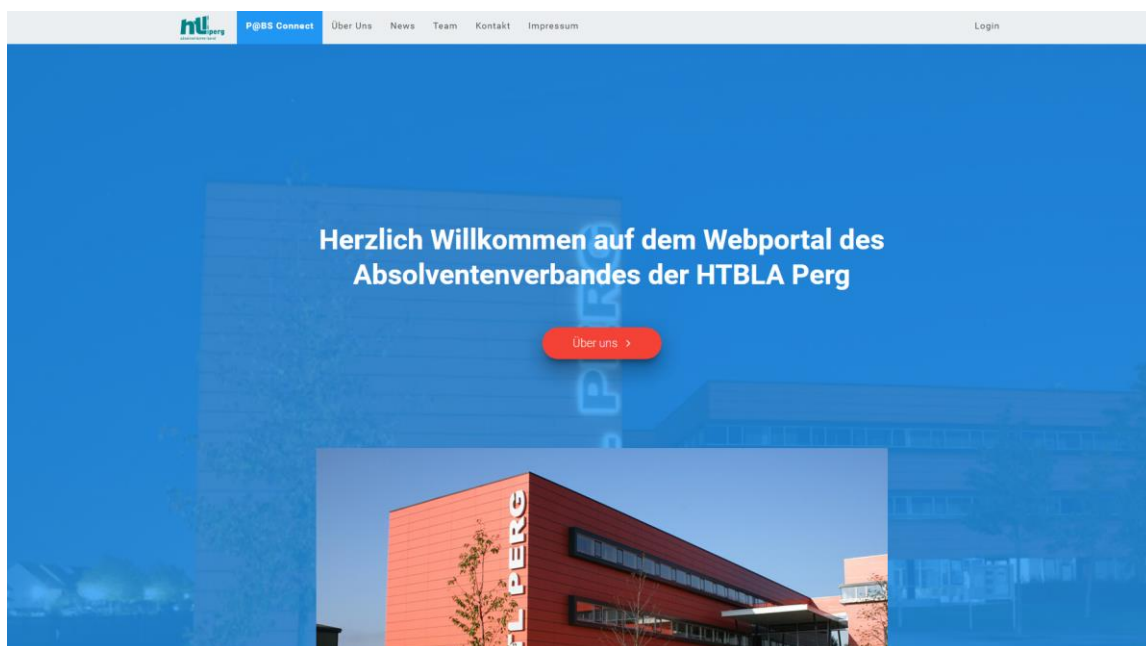


Abbildung 62: Einstieg in das Webportal

Durch einen Klick auf „Login“ wird man auf die Login-Page des internen Bereichs weitergeleitet. Dort kann man sich mit seiner E-Mail-Adresse und seinem Passwort anmelden. Sobald man sich erfolgreich authentifiziert hat, kommt man zur Startseite des internen Bereichs.

Je nach zugewiesener Rolle des Nutzers verfügt der interne Bereich über unterschiedliche zur Verfügung stehende Funktionen, da gewisse Funktionalitäten bestimmten Rollen vorbehalten sind.

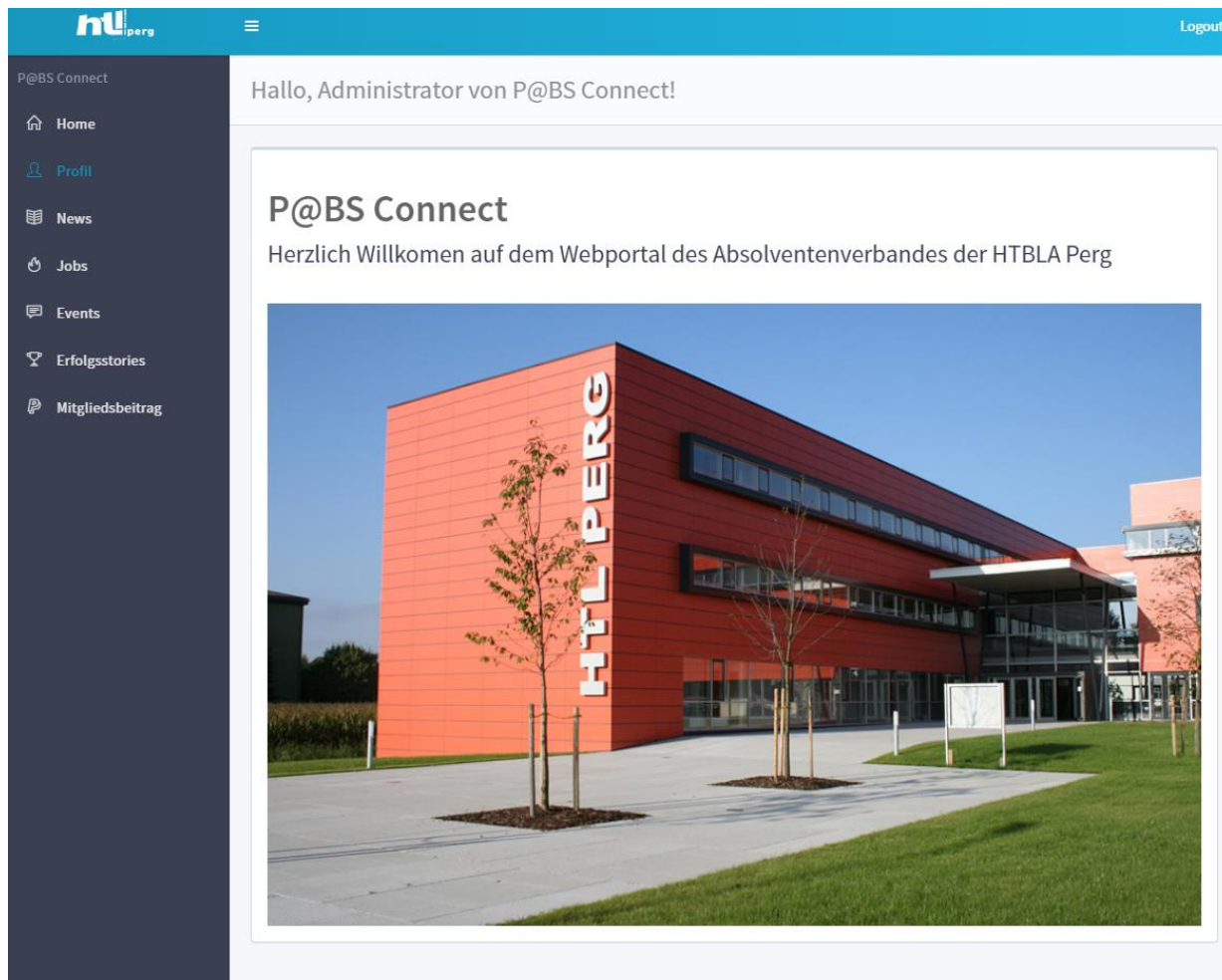


Abbildung 63: Startseite interner Bereich

Durch einen Klick im seitlichen Menü gelangt man zu den Funktionen des jeweiligen Bereichs.

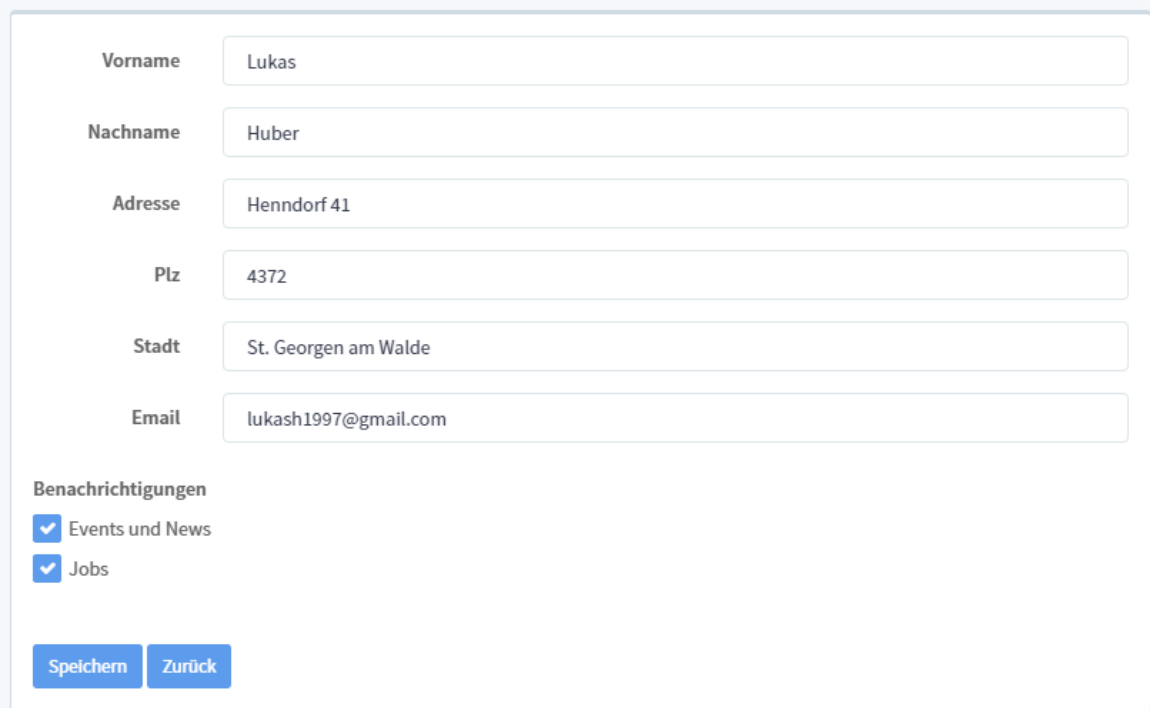
13.2 Funktionen aus Absolventensicht

Der Absolvent hat folgende Funktionen zur Verfügung:

- Bearbeiten der eigenen Stammdaten
- Einsehen und Kommentieren von News
- Einsehen von aktuellen Jobinseraten
- Einsehen und Kommentieren von Events
- Hochladen einer Erfolgsstory
- Bezahlung des Mitgliedbeitrages

13.2.1 Bearbeiten der Stammdaten

Im folgenden Formular kann der Absolvent seine Stammdaten bearbeiten und per Mausclick speichern. Zusätzlich kann er auswählen, ob er bei neuen Jobinseraten oder Events und News benachrichtigt werden will.



Vorname	Lukas
Nachname	Huber
Adresse	Henndorf 41
Plz	4372
Stadt	St. Georgen am Walde
Email	lukash1997@gmail.com

Benachrichtigungen

Events und News

Jobs

Abbildung 64: Stammdatenverwaltung

13.2.2 Einsehen und Kommentieren von News

Absolventen haben die Möglichkeit, unter dem Menüpunkt „News“ im Side-Menu alle aktuellen News einzusehen. Dort wird dem Benutzer eine News-Übersicht in einem sogenannten Masonry-Grid-Layout angezeigt. Weiters bekommt er zusätzliche Informationen wie Autor und Upload-Datum.



Abbildung 65: News-Übersicht

Klickt ein Benutzer nun auf eine bestimmte News, so wird ihm diese im Detail angezeigt. Dort wird der volle Nachrichtentext angezeigt. Zusätzlich werden nun sämtliche Kommentare anderer Benutzer angezeigt. Falls der Nutzer möchte, kann er nun selbst einen Kommentar verfassen, der für andere Nutzer sichtbar wird.

Die Sieger des P@BS Project Awards stehen fest

von Administrator von P@BS Connect 01.04.2017 18:11 9 Aufrufe

Wie jedes Jahr fand auch heuer wieder der P@BS Project-Award statt.

Die Sieger der jeweiligen Kategorien Elevator-Pitch: PocketTranslator
 Innovation: TrackMyTrip
 Technik: GoWest
 Wirtschaft: EasyGym

1 Kommentar

Administrator von P@BS Connect
 Super 01.04.2017 18:38:52

Kommentar verfassen

OK

Abbildung 66: News-Detailansicht

13.2.3 Einsehen von aktuellen Jobinseraten

Unter dem Menüpunkt Jobs kann der Absolvent offene Stelleninsetate ansehen. Dabei werden die offenen Stelleninsetate in einer Tabelle nach Aktualität aufgelistet.

Nummer	Bezeichnung	Firma	Stellenangebot
1	Hardware/Software Techniker(in)	Communications Data Engineering GmbH	Ansehen
2	C++ Developer – Vollzeit / Teilzeit	SLM Solutions Software GmbH	Ansehen
3	Backend Developer – Vollzeit / Teilzeit	SLM Solutions Software GmbH	Ansehen
4	Application Developer WPF – Vollzeit / Teilzeit	SLM Solutions Software GmbH	Ansehen
5	Application Developer – Vollzeit / Teilzeit	SLM Solutions Software GmbH	Ansehen
6	3D Developer – Vollzeit / Teilzeit	SLM Solutions Software GmbH	Ansehen
7	Technical Analyst (w/m)	Fabasoft	Ansehen
8	Infrastructure Consultant (w/m)	Fabasoft	Ansehen

Abbildung 67: offene Jobinsetate

Durch das Klicken auf den „Ansehen“-Button kann man sich die zugehörige PDF-Datei ansehen.

13.2.4 Einsehen, Kommentieren von und Teilnehmen an Events

Das Einsehen der Events ist sehr ähnlich aufgebaut wie bei den News. Es gibt jedoch kleine Unterschiede für Absolvent und Administrator. Zum Beispiel werden Veranstaltungsort sowie Veranstaltungsdatum angezeigt.



Abbildung 68: Event-Übersicht

Weiters haben die Benutzer die Möglichkeit, an den Events durch einen einfachen Klick auf einen Button teilzunehmen. Diese Zusage kann später wieder revidiert werden.

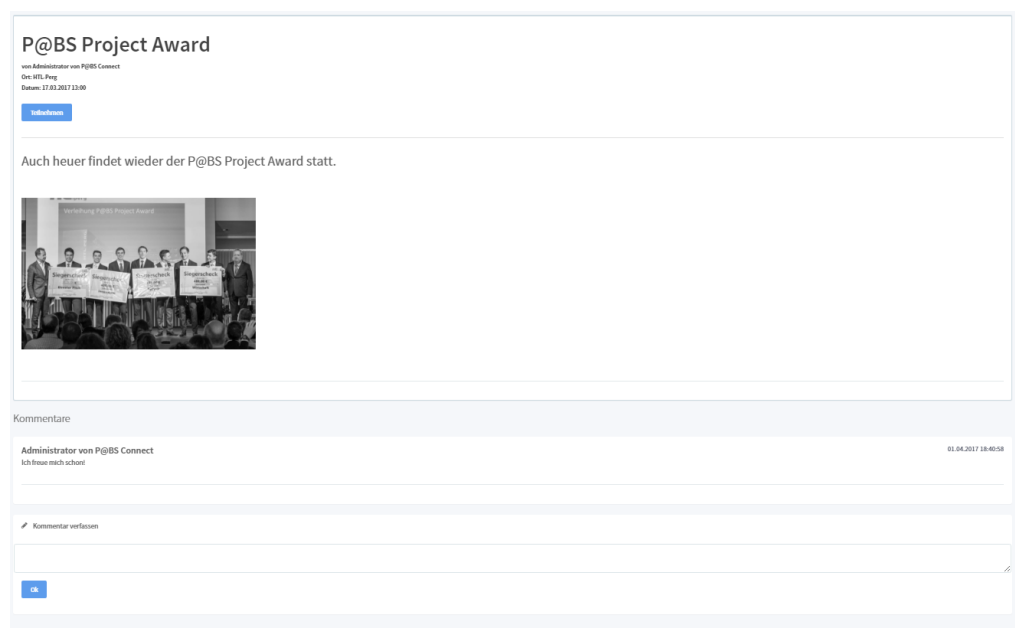


Abbildung 69: Event-Detailansicht

13.2.5 Hochladen einer Erfolgsstory

Im Bereich der Erfolgsstories ist ein Formular für den Absolventen ersichtlich. Um eine neue Erfolgsstory zu erstellen, muss dieses ausgefüllt und ein Bild hochgeladen werden. Beim Klicken auf den „Absenden“-Button generiert das System automatisch eine PDF-Datei und benachrichtigt den Administrator.

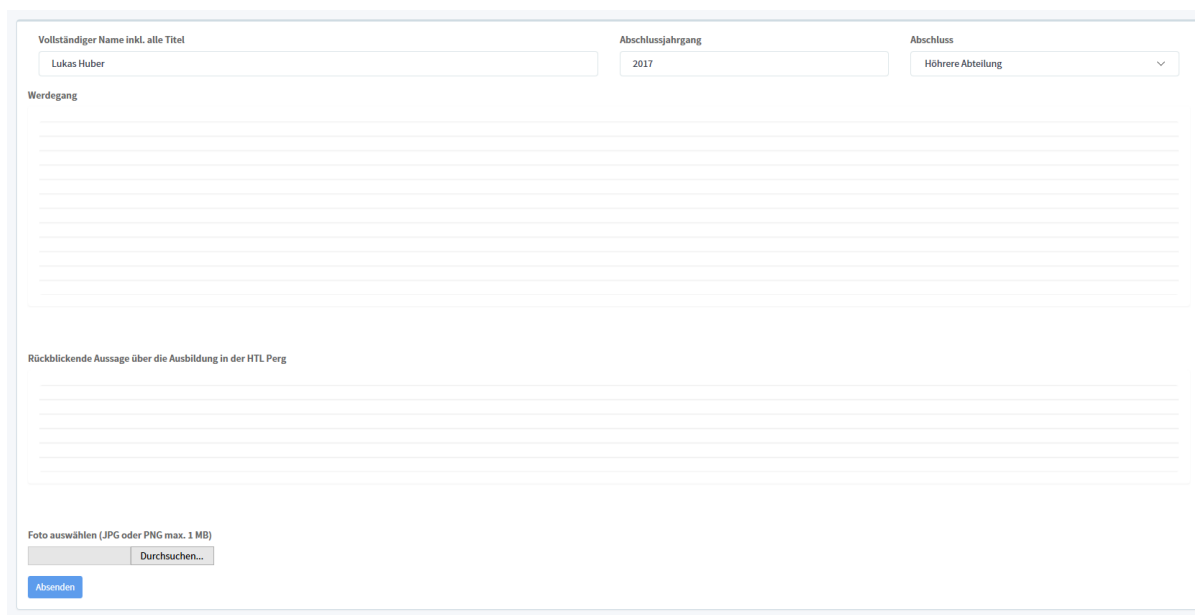


Abbildung 70: Hochladen einer Erfolgsstory

13.2.6 Bezahlung des Mitgliedbeitrages

Im Bereich „Mitgliedsbeitrag“ können Benutzer ihren Mitgliedsbeitrag bezahlen. Bei der Bezahlung kann man sich zwischen folgenden Zahlungsarten entscheiden:

- Bezahlung des Mitgliedbeitrages für ein Jahr
- Automatische jährliche Bezahlung des Mitgliedbeitrages
- Sonstige Forderung bezahlen
- Spenden

Hat man sich für eine Option entschieden, wird man automatisch auf die Bezahlseite von PayPal weitergeleitet und kann dort durch die Eingabe der notwendigen Daten seinen Mitgliedsbeitrag bzw. eine eventuelle Spende bezahlen.

13.3 Funktionen aus Firmensicht

Eine Firma hat folgende Funktionen zur Verfügung:

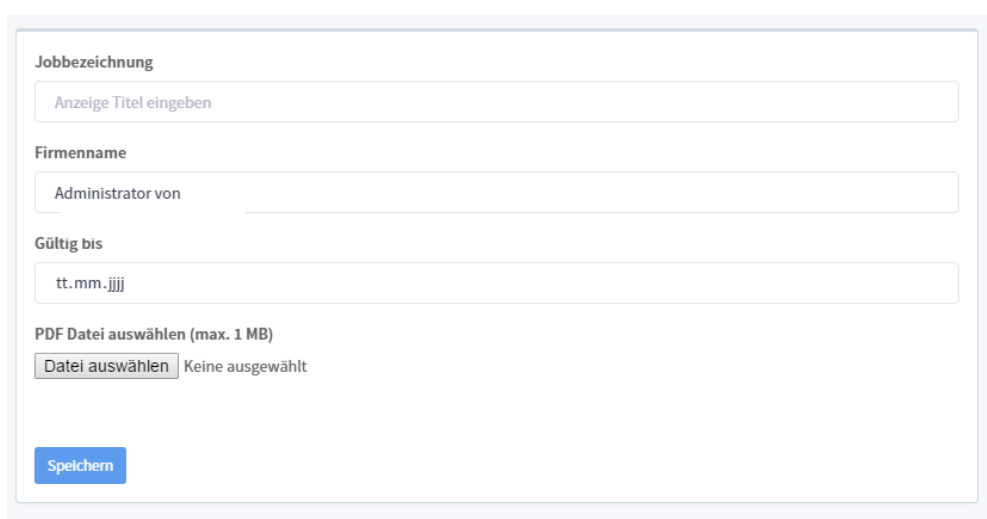
- Bearbeiten der eigenen Stammdaten
- Einsehen und Kommentieren von News
- Hochladen von Jobinseraten
- Verwalten der persönlichen Jobinserte
- Einsehen und Kommentieren von Events
- Bezahlung des Mitgliedbeitrages

Die Funktionen Bearbeiten der Stammdaten, Einsehen und Kommentieren von News, Einsehen und Kommentieren von Events und die Bezahlung des Mitgliedbeitrages unterscheiden sich nicht oder nur sehr geringfügig von der Absolventensicht und werden daher nicht noch einmal näher erläutert.

13.3.1 Hochladen von Jobinseraten

Um ein Jobinserat hochzuladen, muss man das zugehörige Formular ausfüllen und eine PDF-Datei mit der Stellenbeschreibung hochladen.

Danach werden alle Absolventen, die dies wünschen, einmal täglich per E-Mail über neue Stellenangebote informiert.



The screenshot shows a web form for uploading a job posting. It contains the following fields and elements:

- Jobbezeichnung:** A text input field with the placeholder text "Anzeige Titel eingeben".
- Firmenname:** A text input field with the placeholder text "Administrator von".
- Gültig bis:** A date input field with the placeholder text "tt. mm. jjjj".
- PDF Datei auswählen (max. 1 MB):** A file selection area with a "Datei auswählen" button and the text "Keine ausgewählt".
- Speichern:** A blue button at the bottom left of the form.

Abbildung 71: Stelleninserat hochladen

13.3.2 Verwalten der persönlichen Jobinserte

Sollte ein Stelleninserat vorzeitig nicht mehr aktuell sein, kann man es durch einen Klick auf den „Löschen“-Button entfernen.

Weiters ist für die Firmen ersichtlich, wie viele Absolventen sich ein Inserat bereits angesehen haben.

Aktuelle Stelleninserte						
Nummer	Bezeichnung	Firma	Link zum Stellenangebot	Gültig bis	Ansichten	Löschen
1	Infrastructure Consultant (w/m)	Administrator von	Stellenanzeige	31.03.2017	0	Löschen
2	Technical Analyst (w/m)	Administrator von	Stellenanzeige	31.03.2017	1	Löschen
3	3D Developer – Vollzeit / Teilzeit	Administrator von	Stellenanzeige	31.03.2017	1	Löschen
4	Application Developer – Vollzeit / Teilzeit	Administrator von	Stellenanzeige	31.03.2017	2	Löschen
5	Application Developer WPF – Vollzeit / Teilzeit	Administrator von	Stellenanzeige	31.03.2017	0	Löschen
6	Backend Developer – Vollzeit / Teilzeit	Administrator von	Stellenanzeige	31.03.2017	2	Löschen
7	C++ Developer – Vollzeit / Teilzeit	Administrator von	Stellenanzeige	31.03.2017	2	Löschen
8	Hardware/Software Techniker(in)	Administrator von	Stellenanzeige	30.04.2017	2	Löschen

Abbildung 72: Verwalten der persönlichen Stelleninserte

13.4 Funktionen aus Administrator-Sicht

Als Administrator kann das gesamte Webportal verwaltet werden. Man hat daher folgende Funktionen zur Verfügung:







- Bearbeiten der eigenen Stammdaten
- Übersicht über alle Benutzer
- Bearbeiten einzelner Benutzer
- Löschen von Benutzern
- Anlegen von neuen Benutzern
- Anzeigen aller Benutzer-E-Mails in einem Textfeld
- Importieren von neuen Benutzern mittels einer CSV-Datei
- Einsehen und Kommentieren von News
- Erstellen von neuen News
- Bearbeiten und Löschen von News
- Hochladen von Jobinseraten
- Verwalten aller Jobinserate
- Einsehen und Kommentieren von Events
- Erstellen von neuen Events
- Bearbeiten und Löschen von Events
- Verwalten der HTL-Erfolgsstories
- Bezahlung des Mitgliedbeitrages

Die Funktionen Bearbeiten der Stammdaten, Einsehen und Kommentieren von News sowie Einsehen, Kommentieren von und Teilnehmen an Events und die Bezahlung des Mitgliedbeitrages unterscheiden sich nur sehr gering von der Absolventensicht und werden daher nicht noch einmal näher erläutert.

13.4.1 Übersicht über alle Benutzer

Der Administrator hat die Möglichkeit, sich alle im System existierenden Benutzer in einer Tabelle anzeigen zu lassen. Dort kann er alle relevanten Daten dieser einsehen. Weiters kann er gewünschte Benutzer bearbeiten oder löschen bzw. ihr Passwort zurücksetzen, falls ein Nutzer sein Passwort vergessen hat.

Der Administrator hat die Möglichkeit, die Benutzer nach mehreren Selektionskriterien zu filtern, um schnell den gewünschten Eintrag zu finden.

ID	Vorname	Nachname	Email	Adresse	Stadt	Plz	
2	Daniel	Tüchler	daniel_tuechler@hotmail.com	Mühlenweg 17	Schwertberg	4311	 
3	Lukas	Huber	lukash1997@gmail.com	Henndorf 41	St. Georgen am Walde	4372	 
6	Administrator von	P@BS Connect	pabs_admin@htl-perg.ac.at	Machlandstraße 48	Perg	4320	 

Suchen Suchen [Alle Emails](#) [Absolventenemails](#) [Firmenmails](#)

Abbildung 73: Administrator-Benutzerübersicht

13.4.2 Bearbeiten einzelner Benutzer

Über das „Bearbeiten“-Icon gelangt der Administrator von der Übersichts-Tabelle zur Bearbeitungssicht für einen bestimmten Nutzer. Dort können über ein Formular alle Daten des Benutzers abgeändert werden. Dieses Formular ist gleich aufgebaut wie die Stammdaten-Ansicht für das persönliche Profil eines Nutzers. Hier kann vom Namen bis zum Passwort alles durch den Administrator geändert werden.

13.4.3 Löschen von Benutzern

Um einen Benutzer zu löschen, muss der Administrator zuerst zur Ansicht wechseln, in welcher alle aktiven Nutzer im System angezeigt werden. Das Löschen erfolgt nun durch einen einfachen Klick auf das entsprechende Symbol in der Tabellenzeile. Der Benutzer wird daraufhin in der Datenbank als gelöscht markiert und kann somit im Notfall manuell wieder aktiviert werden - die Daten gehen nicht verloren.

13.4.4 Anlegen von neuen Benutzern

Um einen neuen Benutzer anzulegen, muss der Administrator in den Bereich der Administratorfunktionen im Profil-Bereich navigieren. Dort wird er bei Klick auf den Button „Neuer User“ zu einem Formular weitergeleitet, in welchem er die Daten des neuen Nutzers eingeben kann. Die Eingabe ist gleich aufgebaut wie das Formular zur Pflege der eigenen Stammdaten.

13.4.5 Importieren von neuen Benutzern

Ein Administrator hat die Möglichkeit, viele Benutzer auf einmal mittels einer CSV-Datei zu importieren und im System zu registrieren.

Dies ist vor allem am Ende eines Schuljahres wichtig, wenn sämtliche neuen Absolventen automatisch Mitglieder des Absolventenverbands werden und ein Benutzerkonto erhalten sollen.

13.4.6 Anzeigen aller Benutzer-E-Mails in einem Textfeld

Auf Wunsch des Kunden wurde eine einfache Funktion implementiert, welche ihm die E-Mail-Adressen aller Absolventen, Firmen oder aller gespeicherten Benutzer in einem Textfeld anzeigt. Diese kann er nun in sein E-Mail-Programm kopieren, um damit weiterzuarbeiten.

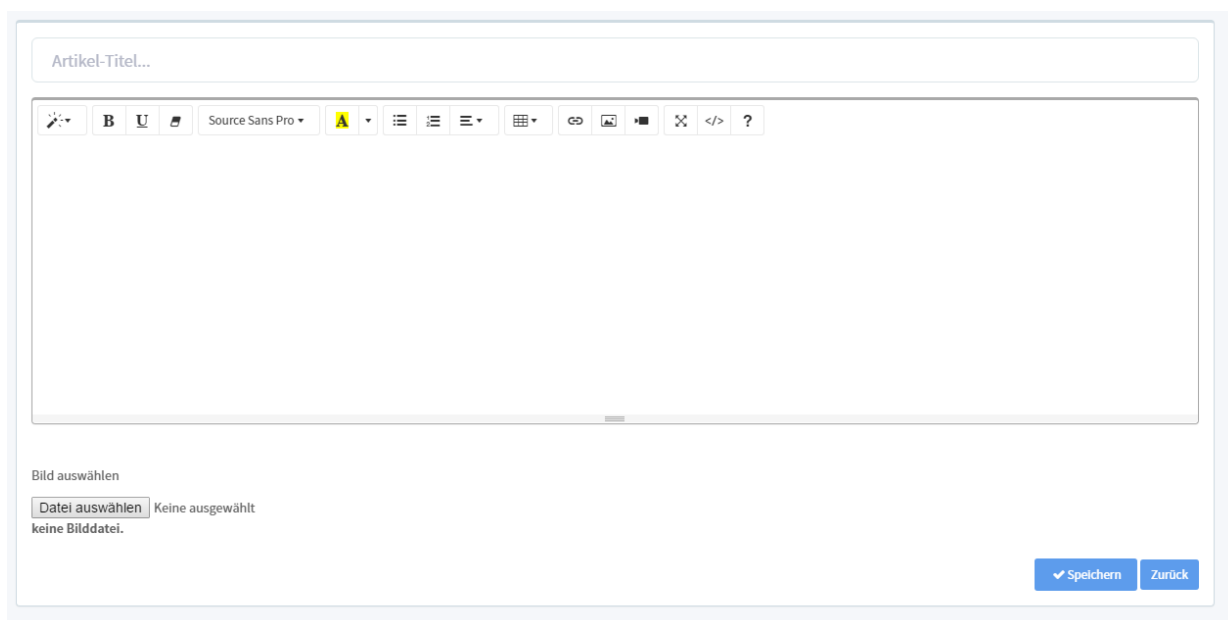


```
daniel_tuechler@hotmail.com; lukash1997@gmail.com; pabs_admin@htl-perg.ac.at;
```

Abbildung 74: Anzeigen aller Benutzer-E-Mails

13.4.7 Erstellen von neuen News

Zum Erstellen von News-Beiträgen wird dem Administrator ein HTML-Editor für den Inhalt, ein Feld für den Titel sowie eine Möglichkeit zum Hochladen eines Titelbilds zur Verfügung gestellt.



The screenshot shows a web interface for creating a new news article. At the top, there is a text input field labeled "Artikel-Titel...". Below this is a rich text editor with a toolbar containing icons for bold, underline, text color, background color, bulleted list, numbered list, indent, outdent, link, unlink, image, video, fullscreen, source code, and help. The editor area is currently empty. Below the editor, there is a section titled "Bild auswählen" with a button labeled "Datei auswählen" and the text "Keine ausgewählt" and "keine Bilddatei.". At the bottom right, there are two buttons: "Speichern" and "Zurück".

Abbildung 75: Erstellen von neuen News

13.4.8 Bearbeiten und Löschen von News

Der Administrator hat zusätzlich die Möglichkeit, News zu bearbeiten und auch zu löschen.

13.4.9 Erstellen von neuen Events

Das Erstellen von Events ist für den Administrator fast ident zum Erstellen von News. Es stehen bloß weitere Input-Felder für Veranstaltungsort und Veranstaltungsdatum und -uhrzeit zur Verfügung. Das Datum kann entweder manuell eingegeben oder über einen Kalender selektiert werden.

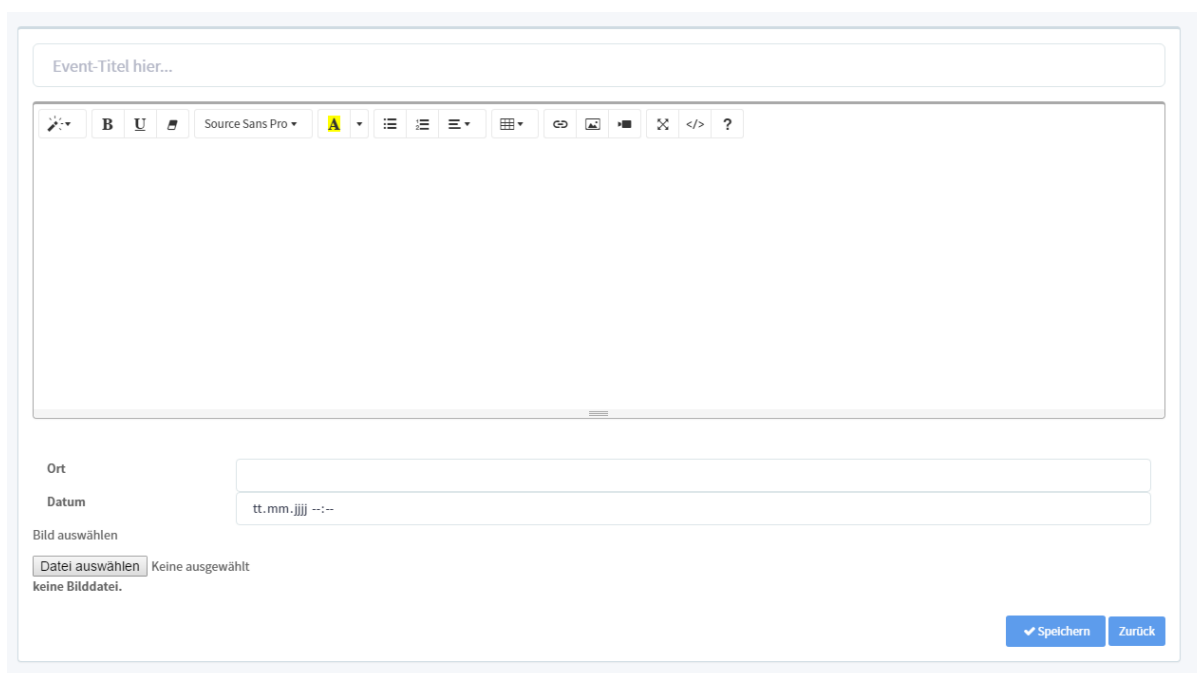


Abbildung 76: Erstellen von neuen Events

13.4.10 Bearbeiten und Löschen von Events

Der Administrator hat zusätzlich die Möglichkeit, Events zu bearbeiten und auch zu löschen.

13.4.11 Verwalten der Erfolgsgories

Als Administrator kann man alle Erfolgsgories aller Absolventen verwalten. Die wichtigste Funktion in diesem Bereich ist der Download der PDF-Dateien der generierten HTL-Erfolgsgories.

Nummer	Name	Erstellt am	Abschlussjahr	PDF	Werdegang	Rückblick	Bild
1	PATRICK NEULINGER & PATRICK MARKSTEINER	25.03.2017	2011	PDF	Werdegang	Rückblick	Bild
2	Sara Kragl	25.03.2017	2012	PDF	Werdegang	Rückblick	Bild

Abbildung 77: Verwaltung der Erfolgsgories

13.5 Benachrichtigungen

Absolventen und Administratoren erhalten Benachrichtigungen per E-Mail. Absolventen erhalten diese E-Mail-Notifikationen nur auf Wunsch.

Ein Administrator wird nach dem erfolgreichen Upload einer HTL-Erfolgsgorie benachrichtigt und kann dann die PDF-Datei ausdrucken.



P@BS Absolventenverband
der HTBLA Perg

Lieber Administrator!

Eine neue Erfolgsgorie wurde hochgeladen:

[online ansehen](#)

Copyright © 2017 P@BS Absolventenverband der HTBLA Perg

Abbildung 78: E-Mail-Benachrichtigung HTL-Erfolgsgorie

Der Absolvent wird bei neuen Jobinseraten benachrichtigt, sofern er dies wünscht. Hierbei prüft das System jeden Tag zu einer bestimmten Uhrzeit, ob neue Stelleninserate verfügbar sind. Sollte es neue Inserate geben, wird das Mail-System aktiv und sendet E-Mail-Benachrichtigungen an die entsprechenden Absolventen aus. Innerhalb der E-Mail sind die neuen Stelleninserate in einer Tabelle aufgelistet und können direkt aus der E-Mail eingesehen werden.



**P@BS Absolventenverband
der HTBLA Perg**

Liebe Absolventen!

Es sind neue Stelleninserate verfügbar:

Abbildung 79: Auszug E-Mail-Benachrichtigung Jobinserate

Nach der Erstellung von neuen News-Beiträgen oder Veranstaltungen werden ebenfalls alle interessierten Absolventen benachrichtigt.



**P@BS Absolventenverband
der HTBLA Perg**

Liebe P@BS-Mitglieder!

Ein neuer Newseintrag wurde erstellt:

Die Sieger des P@BS Project Awards stehen fest
von Administrator von P@BS Connect

[Zur Newsübersicht](#)

Abbildung 80: E-Mail-Benachrichtigung neuer Newseintrag

14 Resümee

In der Planung und Dokumentation der Diplomarbeit wurden die bisherigen Erfahrungen aus dem Projektunterricht berücksichtigt. Dadurch war eine rasche Durchführung der anstehenden Planungsaufgaben möglich. Fixpunkte wie das Abhalten unseres Kick-Off-Meetings und das Verfassen der Projektdefinition gingen leicht von der Hand.

Die Kommunikation mit dem Auftraggeber verlief reibungslos und einwandfrei über den Obmann Dipl.-Ing. Gerhard Zweimüller. Er stand jederzeit zur Verfügung und die Antworten auf etwaige Fragen konnten in kurzer Zeit abgeklärt werden. Durch regelmäßige Statuspräsentationen nach dem Unterricht wurde dem Auftraggeber der Verlauf und die Entwicklungen des Projekts präsentiert und es konnte zwischenzeitliches Feedback eingebracht werden.

Für die Durchführung der Diplomarbeit war das Einarbeiten in neue Technologien notwendig. Durch Selbststudium mithilfe von Dokumentation und Community-Foren im Internet und durch die Hilfe von Herrn Prof. Dipl.-Ing. Christian Aberger verlief dies ohne größere Probleme.

Hauptsächlich waren davon folgende Technologien betroffen:

- JWT und Authentifizierung
- iText
- Joomla und Joomla-Erweiterungen

Durch Meetings mit potentiellen Benutzern des Webportals - sowohl Absolventen als auch P@BS-Mitgliedsfirmen - haben wir Feedback zu den geplanten Funktionen eingeholt. So wurde der ursprünglich angedachte Funktionsumfang erweitert und nicht benötigte Funktionsbereiche vom Projekt ausgenommen.

Im Verlauf der Diplomarbeit hat sich herausgestellt, dass der Einsatz von Joomla für das gesamte Webportal nicht zweckmäßig ist. Da wir dies rechtzeitig erkannt haben, wurde im Rahmen eines Change-Requests auf Angular 2 umgestellt. Wir haben uns für diese Technologie entschieden, da wir aufgrund unserer Erfahrungen aus dem Programmierunterricht diese Technologie für den geplanten Einsatzbereich als ideal angesehen haben. Der Change-Request ist im Anhang ersichtlich.

15 Aufgabenverteilung

Im folgenden Punkt ist festgehalten, wer welche Kapitel der Diplomarbeit verfasst hat. Teilweise wurden Kapitel von mehreren Personen verfasst, da der jeweilige Bereich in die Zuständigkeit mehrerer Diplomanden fällt.

15.1 Lukas Huber

- Stakeholder
 - o Lukas Huber
- Einleitung
- Grundlagen
 - o Verwendete Technologien
 - o Verwendete Bibliotheken und Plug-Ins
 - o Sonstige verwendete Software
- Planung und Realisierung
 - o Meilensteine
 - o Architektur
- Implementierung
 - o Backend
 - Generierung der HTL-Erfolgsstories
 - Dateiupload
 - Mail-System
 - o Frontend
- Ergebnis
 - o Einstieg ins Webportal
 - o Funktionen aus Absolventen-Sicht
 - o Funktionen aus Firmen-Sicht
 - o Funktionen aus Administrator-Sicht
 - o Benachrichtigungen
- Resümee
- Anhang
 - o Auszug aus der Projektdokumentation und Diplomarbeitsplakat

15.2 Christoph Penner

- Genereller Aufbau
- Formatvorlage, Corporate Design
- Eidesstattliche Erklärung
- Gender-Erklärung
- Danksagung
- Impressum
- Kurzbeschreibung
- Abstract
- Stakeholder
 - o Christoph Penner
 - o Betreuung
 - o Lehranstalt
 - o Auftraggeber
- Grundlagen
 - o Verwendete Technologien
 - o Verwendete Entwicklungssysteme
 - o Verwendete Architekturen
 - o Sonstige verwendete Software
- Planung und Realisierung
 - o Architektur
- Implementierung
 - o Backend
 - Datenmodell
 - Allgemeine Informationen
 - Tabellenbeschreibungen
 - Datenbankoperationen
 - REST-API
 - Statuscodes
 - Zugriff auf die Datenbank
 - Datenübertragung über REST
 - Authentifizierung
 - Passwort-Hashing

- Resümee
- Glossar
- Anhang
 - o Dokumentation der REST-API

15.3 Daniel Tüchler

- Stakeholder
 - o Daniel Tüchler
- Einleitung
 - o Ist-Zustand
 - o Motivation
 - o Ziel der Arbeit
- Grundlagen
 - o Verwendete Technologien
- Planung und Realisierung
 - o Meilensteine
 - o Projektstrukturplan
 - o Designs des Webportals – Seitenaufbau
 - o IMV-Matrix
- Implementierung
 - o Frontend
 - CSV-Import von neuen Benutzern
- Ergebnis
 - o Funktionen Absolventenansicht
 - o Funktionen aus Administratorsicht
- Resümee
- Anhang
 - o Auszug aus der Projektdokumentation
 - o Diplomarbeitsplakat

16 Glossar

API

API steht für Application Programming Interface und kann als Programmierschnittstelle übersetzt werden. Eine API dient der Kommunikation mit anderen Systemen und lässt sie auf diese Weise auf Softwareteile zugreifen.

Array

Ein Array ist eine Datenstruktur, die es erlaubt, viele gleichartige Datensätze in einer Variable zu speichern und zu verwalten. Der Zugriff auf einzelne Felder erfolgt über Indizes, die deren Position bezeichnen. (vgl. wikipedia.org, 2017)

Backend

Das Backend ist in einem nach der Schichtentrennung entwickelten Software-System typischerweise weiter vom Benutzer entfernt als das Frontend. Im Vordergrund steht nicht die direkte Interaktion mit dem Nutzer, sondern die Datenverarbeitung und Geschäftslogik im Hintergrund.

Compiler

Als Compiler bezeichnet man ein Computerprogramm, welches Quellcode einer Programmiersprache in eine maschinenlesbare Form übersetzt, damit der Computer die gewünschten Anweisungen ausführen kann. (vgl. wikipedia.org, 2017)

CMS

CMS steht für Content Management System und bezeichnet eine Software-Komponente zur gemeinschaftlichen Schaffung, Manipulation und Organisation von Website-Inhalten (Content). Vor allem für technisch unversierte Anwender ist der ausbleibende oder geringe Programmieraufwand ein großer Vorteil. Auf der anderen Seite wird man durch die Vorgaben des Systems eingeschränkt. (vgl. wikipedia.org, 2017)

CSV

CSV (Comma-separated values) bezeichnet ein Dateiformat zur Speicherung einfach strukturierter Daten in einem Textformat. Wie der Name andeutet werden die Daten durch Beistriche getrennt gespeichert. (vgl. wikipedia.org, 2017)

DBMS

DBMS steht für Datenbank-Management-System. Ein DBMS stellt die zentrale Komponente einer Datenbank dar und ermöglicht Benutzern den Zugriff und die Manipulation von Daten unabhängig von der physischen Speicherung ebendieser. (vgl. searchenterprisesoftware.de, 2017)

Framework

Ein Framework („Rahmenstruktur“) ist ein Programmiergerüst, welches einen Rahmen bietet, mithilfe dessen ein Entwickler eine Anwendung erstellen kann. Somit wird der Entwicklungsprozess unterstützt. (vgl. wikipedia.org, 2017)

Frontend

Das Frontend ist in einem nach der Schichtentrennung entwickelten Software-System typischerweise näher beim Benutzer angesiedelt. Im Vordergrund steht die direkte Interaktion mit dem Nutzer. Die benötigten Daten werden üblicherweise vom Backend bereitgestellt.

HTML-Editor

Ein HTML-Editor ist eine Software-Komponente, die es erlaubt, einfach HTML-Code zu erstellen und zu editieren. Häufig (und auch in dieser Diplomarbeit) werden sogenannte WYSIWYG (What you see is what you get) - Editoren eingesetzt, die dem Anwender während der Bearbeitung direkt den Inhalt so anzeigen, wie er später auf der Webseite aussehen wird.

Java EE

Java Platform, Enterprise Edition (kurz Java EE) bezeichnet eine Softwarearchitektur-Spezifikation für die transaktionsbasierte Ausführung von Anwendungen in Java – insbesondere sind darunter Webanwendungen zu verstehen. (vgl. wikipedia.org, 2017)

JDK

Das Java Development Kit (JDK) von Oracle ist das am weitesten verbreitete Software Development Kit (SDK) für Java und wird für die Entwicklung von Software mit Hilfe der Java-Technologie benötigt. Darin sind Entwicklungswerkzeuge wie der Java-Compiler und der Java-Debugger enthalten. (vgl. wikipedia.org, 2017)

JRE

Mit der Java-Laufzeitumgebung (JRE – Java Runtime Environment) werden Java-Programme unabhängig vom darunterliegenden Betriebssystem ausgeführt. Dazu muss die JRE lediglich auf dem jeweiligen System installiert sein, der Java-Quellcode ist auf sämtlichen Zielplattformen ident. (vgl. wikipedia.org, 2017)

JSON

JSON (JavaScript Object Notation) ist der Name eines kompakten Datenformats für menschenlesbaren Text zum Datenaustausch zwischen Anwendungen. JSON-Dokumente sollen gültiges JavaScript sein. Für JSON existieren Übersetzer für fast jede Programmiersprache, wodurch es in praktisch jeder Anwendung verwendet werden kann. (vgl. wikipedia.org, 2017)

JVM

Die Java Virtual Machine (JVM) gehört zur Java-Laufzeitumgebung und führt den Java-Bytecode auf der Zielplattform aus. Eine eigene virtuelle Maschine wird als Rahmen für jedes gestartete Java-Programm genutzt. (vgl. wikipedia.org, 2017)

Kryptologische Hashfunktion

Als Spezialform der Hashfunktion sollte eine kryptologische Hashfunktion kollisionsfrei sein. Das heißt aus unterschiedlichen Eingabewerten soll immer ein unterschiedlicher Hash errechnet werden. Durch die Definition als Einwegfunktion soll es nicht möglich sein, vom Ergebnis der Funktion auf die Eingabeparameter schließen zu können.

Eine Hashfunktion ist eine Abbildung von großen Eingabemengen auf kleinere Ziel-mengen (Hashwerte). Trotz unterschiedlicher Längen der Eingaben sind die Hash-werte einer Funktion normalerweise immer gleich lang. (vgl. wikipedia.org, 2017)

ORM

Object-relational-mapping beschreibt die Technik der Datenkonvertierung zwischen der objektorientierten Programmierwelt und der relationalen Welt der Datenspeicherung in Datenbanken. Da Objekte durch ihre Speicheradressen und relationale Datensätze durch ihre Werte selbst bestimmt werden, kann die Persistierung zu Problemen führen, wenn beispielsweise zwei wertgleiche, aber verschiedene Objekte gespeichert werden sollen (object-relational impedance mismatch). OR-Mapper sollen diese Probleme lösen und sehen sich für die Persistierung von Objekten verantwortlich, um dem Entwickler diese Mühen zu erleichtern.

Overhead

Als Overhead bezeichnet man in der Informatik Zusatzdaten, die für die Datenübertragung notwendig sind, aber nicht zu den eigentlichen Nutzdaten einer Übertragung beitragen. Ein großer Overhead verlangsamt die Übertragung und benötigt mehr Speicherplatz als zwingend nötig wäre.

PBKDF2

PBKDF2 (Password Based Key Derivation Function 2) bezeichnet eine genormte Funktion zur Ableitung eines Schlüssels aus einem Passwort zum Einsatz in symmetrischen Verschlüsselungsverfahren (Schlüssel für die Ver- und Entschlüsselung ident). (vgl. wikipedia.org, 2017)

Relationale Datenbank

Eine relationale Datenbank beziehungsweise ein relationales Datenbank-Management-System (RDBMS) ist eine tabellenbasierte Möglichkeit der Datenspeicherung. Zur Datenabfrage- oder manipulation wird üblicherweise die Abfragesprache SQL eingesetzt. Die Grundlage für die Technik bildet die relationale Algebra aus der Mathematik. Durch die Speicherform wird grundlegend eine gewisse Datenkonsistenz gewährleistet. (vgl. wikipedia.org, 2017)

SHA512

SHA512 ist ein Ableger von SHA-2, einer standardisierten kryptologischen Hashfunktion. 512 bezeichnet hierbei die Länge des Hashwerts in Bit.

SMTP

SMTP steht für Simple Mail Transfer Protocol und ist ein Protokoll zum Austausch von E-Mails in Computernetzen.

Responsives Layout

Eine responsiv gestaltete Website hat die Eigenschaft, dass sie auf die Eigenschaften eines Endgeräts reagieren können. Gemeint ist damit die angenehme Verwendbarkeit auf mobilen Geräten mit kleinen Bildschirmen wie Smartphones. Der Seiteninhalt passt sich dynamisch an die Bildschirmgröße an.

XML

XML (Extensible Markup Language) ist eine Auszeichnungssprache, die zur Darstellung von hierarchisch strukturierten Daten in Textform dient. Es gibt hierbei keine fix vorgegebenen Strukturen, die Datenhierarchie kann für eine spezielle Anwendung selbst definiert werden. XML wird neben JSON gerne zur Datenübertragung im Internet (vor allem für REST-APIs) verwendet. (vgl. wikipedia.org, 2017)

17 Anhang

17.1 Dokumentation der REST-API

Die Basis-URL zum Aufruf der REST-API lautet: <https://pabs.htl-perg.ac.at/pabsweb>

Im folgenden Abschnitt sind die Aufrufrouen der einzelnen Entitäten genauer beschrieben.

Um eine Schnittstelle verwenden zu können, ist eine Authentifizierung über den Auth-Endpoint notwendig. Nach erfolgreicher Übermittlung von E-Mail und Passwort im korrekten Format erhält man einen JSON Web Token, der im „Authorization“-Header bei anderen Anfragen angegeben werden kann, um sich zu authentifizieren.

17.1.1 Auth

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Login	POST	/auth/login	Form: email, password	Generiert und retourniert bei korrekter Email und Passwort-Kombination einen JSON Web Token, der bei nachfolgenden Requests anzugeben ist	200 + JWT im Header 401 Unauthorized: Email oder Passwort falsch

17.1.2 PCUser

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle User anzeigen	GET	/user	-	Gibt alle User zurück	200 + List<PCUser>
User für Id anzeigen	GET	/user/{id}	Path: id	Gibt einen einzelnen User aufgrund der id zurück	200 + PCUser 404: User mit id nicht gefunden
Menge an Usern anzeigen	POST	/user	Body: PCUser[]	Legt alle User des übergebenen Arrays an, setzt die createdAt-Zeitstempel	200 + „<Anzahl> User angelegt“ 400 + „Email <Email> bereits vergeben“ 400 + „Email <Email> ungueltig“ 400 + „Fehler bei User <Email>“
User updaten	PUT	/user/{id}	Path: id, Body: PCUser	Aktualisiert einen User mit der id aufgrund des übergebenen Bodies, setzt den updatedAt-Zeitstempel	200 404: User mit id nicht gefunden
Passwort ändern	PUT	/user/{id}/changePassword	Path: id, Form: oldPW, newPW	Aktualisiert Passwort auf Formparameter newPW, wenn Formparameter oldPW gültig für das aktuelle Passwort ist und setzt den korrekten neuen Hash bzw. Salt	200 400: Falsches oldPW
Passwort zurücksetzen (Administrator)	PUT	/user/{id}/resetPassword	Path: id, Form: newPW	Weist dem User mit der id das Passwort newPW zu (Hash+Salt-Aktualisierung)	200 400: User mit id nicht gefunden
User löschen	DELETE	/user/{id}	Path: id	Setzt den deletedAt-Zeitstempel	200 404: User mit id nicht gefunden
E-Mails aller Absolventen	GET	/user/mails/alumni	-	Gibt die E-Mail Adressen aller Absolventen als String[] (getrennt durch Beistriche) zurück	200 + String[] mails
E-Mails aller Firmen	GET	/user/mails/company	-	Gibt die E-Mail Adressen aller Firmen als String[] (getrennt durch Beistriche) zurück	200 + String[] mails

17.1.3 PCNews

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle News anzeigen	GET	/news	-	Gibt alle News zurück	200 + List<PCNews>
News-Übersicht anzeigen	GET	/news/newsList	-	Gibt eine Menge an Informationen von News für die Übersicht zurück	200 + List<PCNewsDto>
News für Id anzeigen	GET	/news/{id}	Path: id	Gibt eine einzelne News aufgrund der id zurück	200 + PCNews 404: News mit id nicht gefunden
News anlegen	POST	/news	Body: PCNews	Legt eine News aufgrund des übergebenen Bodies an; setzt den createdAt-Zeitstempel; schickt eine E-Mail an User, die dies wünschen	200 + PCNews 400 + „Fehler bei News <Title>“
News updaten	PUT	/news/{id}	Path: id, Body: PCNews	Aktualisiert eine News mit der id aufgrund des übergebenen Bodies, setzt den updatedAt-Zeitstempel	200 404: News mit id nicht gefunden
News löschen	DELETE	/news/{id}	Path: id	Setzt den deletedAt-Zeitstempel	200 404: News mit id nicht gefunden
Kommentar hinzufügen	POST	/news/{id}/comment	Path: id, Body: PCComment	Fügt einen Kommentar aufgrund des Bodies zur angegebenen News hinzu	200 404: News mit id nicht gefunden
Kommentar löschen	DELETE	/news/{id}/comment/{commentId}	Path: id, commentId	Löscht einen Kommentar aufgrund der commentId, sucht diesen mithilfe der id	200 404: News mit id oder Kommentar mit commentId nicht gefunden

17.1.4 PCEvent

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle Events anzeigen	GET	/event	-	Gibt alle Events zurück	200 + List<PCEvent>
Event-Übersicht anzeigen	GET	/news/eventList	-	Gibt eine Menge an Informationen von Events für die Übersicht zurück	200 + List<PCEventDto>
Event für Id anzeigen	GET	/event/{id}	Path: id	Gibt ein einzelnes Event aufgrund der id zurück	200 + PCEvent 404: Event mit id nicht gefunden
Event anlegen	POST	/event	Body: PCEvent	Legt ein Event aufgrund des übergebenen Bodies an; setzt den createdAt-Zeitstempel; schickt eine E-Mail an User, die dies wünschen	200 + PCEvent 400 + „Fehler bei Event <Name>“
Event updaten	PUT	/event/{id}	Path: id, Body: PCEvent	Aktualisiert ein Event mit der id aufgrund des übergebenen Bodies, setzt den updatedAt-Zeitstempel	200 404: Event mit id nicht gefunden
Event löschen	DELETE	/event/{id}	Path: id	Setzt den deletedAt-Zeitstempel	200 404: Event mit id nicht gefunden
Kommentar hinzufügen	POST	/event/{id}/comment	Path: id, Body: PCComment	Fügt einen Kommentar aufgrund des Bodies zum angegebenen Event hinzu	200 404: Event mit id nicht gefunden
Kommentar löschen	DELETE	/event/{id}/comment/{commentId}	Path: id, commentId	Löscht einen Kommentar aufgrund der commentId, sucht diesen mithilfe der id	200 404: Event mit id oder Kommentar mit commentId nicht gefunden
An Event teilnehmen	PUT	/event/{id}/participation/{pcuserId}	Path: id, pcuserId	Fügt einen User aufgrund der pcuserId zum Event mit der id hinzu	200 404: Event mit id oder User mit pcuserId nicht gefunden
Teilnahme zurückziehen	DELETE	/event/{id}/participation/{pcuserId}	Path: id, pcuserId	Revidiert die Teilnahme eines Users aufgrund der pcuserId vom Event mit der id	200 404: Event mit id oder User mit pcuserId nicht gefunden

17.1.5 PCFileUpload

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle Uploads anzeigen Upload für Id anzeigen	GET	/pcfileupload	-	Gibt alle Uploads zurück	200 + List<PCFileUpload>
	GET	/pcfileupload/{id}	-	Gibt einen einzelnen Upload aufgrund der id zurück	200 + PCFileUpload 404: Upload mit id nicht gefunden
Alle Uploads eines bestimmten Benutzers anzeigen	GET	/pcfileupload/user/{id}	-	Gibt alle Uploads des Users mit der id zurück	200 + List<PCFileUpload> 404: User mit der id nicht gefunden
	POST	/pcfileupload	file: hochzuladende Datei type: Typ der Datei uploadedBy: UserId des Uploaders	Im Header muss folgendes Feld gesetzt sein: Enctype: multipart/form-data Akzeptiert werden nur Dateien mit dem Dateiformat: pdf, jpg, png	201 + PCFileUpload 400: Bad Request 404: User mit id nicht gefunden
Upload Löschen	DELETE	/pcfileupload/{id}	-	Setzt den deletedAt-Zeitstempel	200 404: Upload mit der id nicht gefunden

17.1.6 PCJobAd

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle JobAds anzeigen	GET	/pcjobad	-	Gibt alle JobAds zurück	200 + List<PCJobAd>
JobAd für Id anzeigen	GET	/pcjobad/{id}	-	Gibt ein einzelnes JobAd aufgrund der id zurück	200 + PCJobAd 404: JobAd mit id nicht gefunden
Alle JobAds eines bestimmten Benutzers anzeigen	GET	/pcjobad/user/{id}	-	Gibt alle JobAds des Users mit der id zurück	200 + List<PCJobAd> 404: User mit id nicht gefunden
JobAd anlegen	POST	/pcjobad	Body: PCJobAd	Legt ein JobAd aufgrund des übergebenen Bodies an; setzt den createdAt-Zeitstempel	201 + PCJobAd
Viewcount erhöhen	POST	/pcjobad/viewcount/{id}	-	Viewcount des JobAds mit der id erhöhen	200
JobAd löschen	DELETE	/pcjobad/{id}	-	Setzt den deletedAt-Zeitstempel	200

17.1.7 PCNotificator

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Email bei neuen Stelleninseraten senden	GET	/pcnotificator/send-Email?password=emailadmin	-	Versendet eine Benachrichtigungs-E-Mail, wenn neue Stelleninserate verfügbar sind	200 403: Passwort URL-Aufruf nicht korrekt

17.1.8 PCStory

Funktion	http-Methode	Pfad	Parameter, Body	Erklärung / Bemerkung	Rückgabe
Alle Stories anzeigen	GET	/pcstory	-	Gibt alle Stories zurück	200 + List<PCStory>
Story anlegen	POST	/pcstory	Body: PCStory	Legt eine Story aufgrund des übergebenen Bo-dies an; setzt den createdAt-Zeitstempel	201 + PCStory

17.2 Auszug aus der Projektdokumentation

Im Folgenden ist auszugsweise die Projektdokumentation angeführt.

17.2.1 Risikomanagement

Beschreibung des Risikos und der möglichen Folgen	Eintrittswahrscheinlichkeit	Potenzielle Schadenshöhe	Gegenmaßnahmen (GM)	Eintrittsindikatoren (EI)	Verantwortlicher	Eintrittswahrscheinlichkeit nach Gegenmaßnahmen	Schadenshöhe nach Gegenmaßnahmen
Die Absolventen und Firmen, die das Produkt enthalten, finden keinen Anreiz, den Inhalt zu verfolgen und interessieren sich bald nicht mehr für	3	5	Anspruchenderes Design auf Feedback reagieren überarbeitete Benutzeroberfläche	Tester geben Desinteresse bekannt	Team	2	5
Das User Interface / GUI ist nicht ansprechend gestaltet, ev. unübersichtlich und nicht benutzerfreundlich; der User findet sich nicht zurecht oder findet das Design nicht ansprechend	2	2	Redesign des UI Erhöhung der Benutzerfreundlichkeit	Feedback von Tester, Team	Team	1	2
Die Apps laufen (vor allem auf älteren) Geräten nicht flüssig Lange Wartezeit auf Feedback der App	3	4	Effizienzsteigerung Reimplementierung	Techniktests auf Testgeräten bringen schlechte Ergebnisse	Penner, Huber	1	4
Es treten häufig Bugs auf, die Fehler der App offenlegen, oder das System sogar zum Absturz bringen	2	4	Bug-Fixing Reimplementieren der betroffenen Bereiche	Beim Testen fallen viele Fehler auf	Team	2	2
Einzelne Tasks/User Stories oder das Gesamtvorhaben werden nicht zu den festgelegten Terminen fertiggestellt	3	4	Genaue Zeitplanung Projektmanagement Scrum-Planung	Das Team kann für die Erfüllung bestimmter Aufgaben den vereinbarten Zeitplan nicht einhalten	Team	1	3
Ein Teammitglied fällt längerfristig aus und kann seine Projektarbeit aufgrund einer Krankheit, Verletzung, etc. nicht ausführen	2	3	Höherer Arbeitseinsatz der verbleibenden Mitarbeiter ev. Weglassen mancher Funktionen, um den Zeitplan einzuhalten	Krankheit/Verletzung/etc eines Teammitgliedes	Team	1	2

17.2.2 Auszug Change-Request

P@BS Connect	
Projektmitglieder	Lukas Huber, Christoph Penner, Daniel Tüchler
Datum	15. Februar 2017
Klasse	5AHIF
Bearbeiter	Daniel Tüchler

Change Request 15.02.2017

2 Aktuelle Situation

Laut Besprechung und Planung wurde die Realisierung des Projekts mithilfe von Joomla! angepeilt. Im Verlauf der Entwicklung hat sicher immer mehr herausgestellt, dass das Content-Management-System Joomla! nicht ausreichend bzw. nicht zweckmäßig ist, um die gewünschten bzw. geplanten Funktionen zu realisieren.

Aus diesem Grund wurden Überlegungen, welche die Entwicklung des Projekts mithilfe anderer Technologien vorsehen, getätigt. Dies wird im folgenden Punkt genauer erklärt.

3 Neues Konzept

In der neuen Variante bleibt der allgemein-zugängliche Teil der Homepage mittels Joomla! bestehen. Die Applikationsebene der Website wird mittels angular2 realisiert. Die geplanten mobilen Apps für Android und iOS werden aufgrund von Feedback der Absolventen und Firmen aus den Anforderungen genommen, um die Qualität des Webinterface zu erhöhen.

Folgende Funktionsbereiche sind bis zur Präsentation am 17.03.2017 fertigzustellen:

- Stammdatenverwaltung
- Stellenangebote

4 Auswirkungen

Die Projekttermine müssen angepasst werden und der geplante Lieferumfang ebenfalls.

4.1 Vorteile

- Keine Abhängigkeit von Joomla! und externen Plugins für die Applikationsebene
- Einbringen von eigenem Know-How
- Kundenwünsche besser realisierbar
- Konzentration auf die lt. Feedback wichtigen Teile des Systems

4.2 Nachteile

- Höherer Arbeitsaufwand
- Umstrukturierung der Arbeitsverteilung notwendig

17.2.3 Teilabnahmeformular

P@BS Connect	
Projektmitglieder	Lukas Huber, Christoph Penner, Daniel Tüchler
Datum	15. Februar 2017
Klasse	5AHIF
Bearbeiter	Lukas Huber, Christoph Penner
Teilabnahme informelle Homepage	

2 Allgemein

Das Projektteam bittet den Obmann des Absolventenverbandes Herrn Dipl.-Ing. Gerhard Zweimüller zur Abnahme folgender Funktionen der informellen Homepage (aktuell pabsneu.htl-perg.ac.at). Um die Ablösung des alten Systems zu beschleunigen bittet das Projektteam um eine ehest mögliche Abnahme.

Sobald die Abnahme erfolgt ist und die unter Bemerkung (weiter unten) angeführten Punkte erledigt sind, veranlasst der Projektleiter eine Abschaltung des alten Systems. Zusätzlich wird die Domain von pabsneu.htl-perg.ac.at auf pabs.htl-perg.ac.at umgeändert.

Diese Ablösung sollte aufgrund des P@BS Project Awards bis spätestens 05.03.2017 erfolgen.

3 abzunehmende Funktionen

3.1 informelle Homepage

Wir bitten den Auftraggeber, die Inhalte der unten genannten Tabs zu kontrollieren und bei Fehlern dem Projektteam Bescheid zu geben.

Tab	Beschreibung	Bemerkung
Home	Startseite der Homepage mit einem Bild	Dieses Bild soll sobald als möglich durch ein „Gruppenfoto“ der Mitglieder des Absolventenverbandes ersetzt werden. Ein solches Foto ist durch den Absolventenverband bereitzustellen.
Über Uns	Allgemeine Informationen zum P@BS Absolventenverband der HTBLA Perg. Zusätzlich gibt es einen Link auf die Website der HTL Perg, um allgemeine Informationen einholen zu können.	Diese Informationen wurden von der alten Homepage übernommen.
News	Aktuelle öffentliche News (Ankündigungen, etc. welche öffentlich sichtbar sein sollen)	Wie neue Informationen hinzugefügt werden können, wird weiter unten beschrieben.
Team	Aktuelles Team des P@BS Absolventenverbandes Name inkl. Portraitfoto und Nennung der Funktion	Bei den Portraitfotos handelt es sich um Bilder der HTL Homepage in höherer Auflösung. Alle externen Personen haben zurzeit ein Standardbild.

P@BS Connect	
Projektmitglieder	Lukas Huber, Christoph Penner, Daniel Tüchler
Datum	15. Februar 2017
Klasse	5AHIF
Bearbeiter	Lukas Huber, Christoph Penner

Teilabnahme informelle Homepage

		Das Projektteam empfiehlt einen Austausch der aktuellen Bilder durch aktuelle Portraitfotos, darum bittet das Projektteam den Auftraggeber, die betreffenden Absolventen zu kontaktieren und diese Bilder dem Projektteam zur Verfügung zu stellen, damit die Fotos ehestmöglich ausgetauscht werden können.
Kontakt	Kontaktseite mit Adressdaten, Karte, Bankdaten und Emailadresse Link zur Kooperationsvereinbarungsseite der HTL Perg Link zum Kontaktformular der HTL Perg	IBAN und BIC wurden aus den Informationen der alten Homepage generiert. Das Projektteam bittet den Auftraggeber um Prüfung der Bankdaten.
Impressum	Standardimpressum	Dieses Impressum hat das Projektteam vom der HTL Perg Homepage übernommen und adaptiert.

3.2 responsive Layout

Die informelle Homepage ist komplett responsive. Dadurch kann sie sowohl am Computer als auch auf mobilen Geräten (Smartphone, Tablet) ohne Probleme betrachtet werden.

17.3 Diplomarbeitsplakat

P@BSconnect

htuiperg

5AHIF

Projektbeschreibung

- Neues Webportal des Absolventenverbandes
- Verwaltung der Stammdaten
- Newsverwaltung
- Veranstaltungen erstellen und verwalten
- Erstellen von Jobangeboten
- Onlinebezahlung des Mitgliedbeitrags
- Erstellen von Erfolgsgeschichten

Projektteam

- Lukas Huber
- Christoph Penner
- Daniel Tüchler

Verwendete Technologien

Joomla! PayPal MySQL

NEWS JOBS

€

htuiperg

17.4 Literaturverzeichnis

centratrip.com. (03 2017). Von <http://centratrip.com/images/Teknologi.png> abgerufen

clker.com. (03 2017). Von

<http://www.clker.com/cliparts/6/9/a/c/13438417961910789838Linux%20Server%20Final.svg.hi.png> abgerufen

clker.com. (03 2017). Von

<http://www.clker.com/cliparts/6/9/a/c/13438417961910789838Linux%20Server%20Final.svg.hi.png> abgerufen

Copernic Forum. (03 2017). Von <http://forum.copernic.com/index.php/search-inside-dropbox-files/> abgerufen

Elon Musk

FCP. (03 2017). Von <http://www.fcp.at/de/projekte/htbla-perg> abgerufen

fedoraproject.org. (03 2017). Von

https://fedoraproject.org/w/uploads/a/a3/Wildfly_logo.png abgerufen

getpostman.com (03 2017). Von <https://www.getpostman.com/> abgerufen

git-for-windows.github.io. (03 2017). Von <https://git-for-windows.github.io/> abgerufen

Google Play. (03 2017). Von

<https://play.google.com/store/apps/details?id=org.telegram.messenger&hl=de> abgerufen

HTL Perg. (03 2017). Von <http://htl-perg.ac.at/hoehere-abteilung> abgerufen

Poblog.getpostman.com. (03 2017). Von http://blog.getpostman.com/wp-content/uploads/2015/04/logo-postman-512--551cff77v1_site_icon.png abgerufen

javacodegeeks.com. (03 2017). Von <https://www.javacodegeeks.com/wp-content/uploads/2016/01/javascript-shield.png> abgerufen

jomboom.com. (03 2017). Von <http://jomboom.com/template/1560/Akeeba-Backup/> abgerufen

JoomlaArt. (03 2017). Von <https://www.joomlaart.com/blog/introducing-ja-builder-build-sites-for-joomla-on-the-fly-part-ii> abgerufen

jwt.io. (03 2017). Von <https://jwt.io/> abgerufen

MBT Check. (03 2017). Von <http://mbtcheck.com/v2/wp-content/uploads/2016/08/angular.png> abgerufen

mbtcheck.com. (03 2017). Von <http://mbtcheck.com/v2/wp-content/uploads/2016/08/angular.png> abgerufen

memz.co. (03 2017). Von <https://memz.co/wp-content/uploads/2016/03/api.png> abgerufen

Microsoft. (03 2017). Von <https://products.office.com/de-at/try> abgerufen

outdatedbrowser.com. (03 2017). Von <http://outdatedbrowser.com/public/imgs/browsers-bg.png> abgerufen

P@BS. (03 2017). Von pabs.htl-perg.ac.at abgerufen

Pamela Zave

PDF association. (03 2017). Von <https://www.pdfa.org/wp-content/uploads/2016/08/ITSC-Logo-Horizontal-RGB-300dpi.png> abgerufen

qph.ec.quoracdn.net. (03 2017). Von <https://qph.ec.quoracdn.net/main-qimg-9c7723e302afb4d0063023eba3055e9e> abgerufen

Queen Elizabeth II.

searchenterprisesoftware.de. (03 2017). Von <http://www.searchenterprisesoftware.de/definition/Datenbank-Managementssystem-DBMS> abgerufen

selfhtml. (03 2017). Von <https://wiki.selfhtml.org/wiki/HTML> abgerufen

toptal.com. (03 2017). Von <https://uploads.toptal.io/blog/category/logo/364/CSS.png> abgerufen

Trish Parr

tutorialspoint. (03 2017). Von https://www.tutorialspoint.com/jpa/jpa_architecture.htm abgerufen

tutorialspoint. (03 2017). Von <https://www.tutorialspoint.com/jpa/index.htm> abgerufen

ubuntuusers. (03 2017). Von <https://wiki.ubuntuusers.de/PuTTY/> abgerufen

Visual Studio Marketplace. (03 2017). Von <https://marketplace.visualstudio.com/items?itemName=ms-vsts.jboss-wildfly-management-extension> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Ubuntu> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/SQL> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/MySQL> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Java-Technologie> abgerufen

wikipedia.org. (03 2017). Von [https://de.wikipedia.org/wiki/Java_\(Programmiersprache\)](https://de.wikipedia.org/wiki/Java_(Programmiersprache)) abgerufen

wikipedia.org. (03 2017). Von <https://en.wikipedia.org/wiki/HTML> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/JavaScript> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/AngularJS> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Web-Engineering> abgerufen

wikipedia.org. (03 2017). Von [https://de.wikipedia.org/wiki/Eclipse_\(IDE\)](https://de.wikipedia.org/wiki/Eclipse_(IDE)) abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Joomla#/media/File:Joomla!-Logo.svg> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Content-Management-System> abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/TypeScript#/media/File:TypeScript_Logo.png abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Data_Access_Object abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Adobe_Photoshop#/media/File:Photoshop_CC_icon.png abgerufen

wikipedia.org. (03 2017). Von

<https://upload.wikimedia.org/wikipedia/commons/thumb/e/e8/Joomla%21-Logo.svg/2000px-Joomla%21-Logo.svg.png> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/HTTP-Statuscode> abgerufen

wikipedia.org. (03 2017). Von [https://de.wikipedia.org/wiki/Salt_\(Kryptologie\)](https://de.wikipedia.org/wiki/Salt_(Kryptologie)) abgerufen

wikipedia.org. (03 2017). Von [https://de.wikipedia.org/wiki/Feld_\(Datentyp\)](https://de.wikipedia.org/wiki/Feld_(Datentyp)) abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Compiler> abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Content-Management-System> abgerufen

wikipedia.org. (03 2017). Von [https://de.wikipedia.org/wiki/CSV_\(Dateiformat\)](https://de.wikipedia.org/wiki/CSV_(Dateiformat)) abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Framework> abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Java_Development_Kit abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Java-Laufzeitumgebung> abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/JavaScript_Object_Notation abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Java_Virtual_Machine abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/Hashfunktion> abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Relationale_Datenbank abgerufen

wikipedia.org. (03 2017). Von <https://de.wikipedia.org/wiki/PBKDF2> abgerufen

wikipedia.org. (03 2017). Von https://de.wikipedia.org/wiki/Extensible_Markup_Language abgerufen

Wildfly. (03 2017). Von <http://wildfly.org/about/> abgerufen

Zazzle. (03 2017). Von https://www.zazzle.at/gitkraken_aufkleber-217656805059903623 abgerufen

17.5 Abbildungsverzeichnis

Abbildung 1: Lukas Huber (Privat, 2017).....	16
Abbildung 2: Christoph Penner (Privat, 2017)	17
Abbildung 3: Daniel Tüchler (Privat, 2017).....	18
Abbildung 4: Dipl.-Ing. Christian Reisinger (HTL Perg, 2017)	19
Abbildung 5: HTL Perg (FCP, 2017).....	19
Abbildung 6: Dipl.-Ing. Gerhard Zweimüller (HTL Perg, 2017)	21
Abbildung 7: Startseite der alten Homepage.....	22
Abbildung 8: Ubuntu-Logo (wikipedia.org, 2017)	27
Abbildung 9: MySQL-Logo (wikipedia.org, 2017)	28
Abbildung 10: Java-Logo (wikipedia.org, 2017)	28
Abbildung 11: WildFly-Logo (Visual Studio Marketplace, 2017).....	29
Abbildung 12: HTML-Logo (wikipedia.org, 2017)	31
Abbildung 13: CSS-Logo (toptal.com, 2017)	31
Abbildung 14: PHP-Logo (qph.ec.quoracdn.net, 2017).....	31
Abbildung 15: JavaScript-Logo (javacodegeeks.com, 2017).....	32
Abbildung 16: Angular 2-Logo (mbtcheck.com, 2017).....	32
Abbildung 17: TypeScript-Logo (wikipedia.org, 2017).....	32
Abbildung 18: Git-Logo (git-for-windows.github.io, 2017).....	33
Abbildung 19: Joomla-Logo (wikipedia.org, 2017)	33
Abbildung 20: Ausschnitt von MySQL Workbench	34
Abbildung 21: Ausschnitt der Eclipse-IDE	35
Abbildung 22: JetBrains WebStorm.....	36
Abbildung 23: iText-Logo (PDF association, 2017)	37
Abbildung 24: JA Builder (JoomlaArt, 2017).....	37
Abbildung 25: Logo Akeeba Backup (jomboom.com, 2017).....	37
Abbildung 26: kodierter JWT	39
Abbildung 27: dekodierter JWT	40
Abbildung 28: PuTTY-Terminal	42
Abbildung 29: Java-Datei in Notepad++	43
Abbildung 30: Übersicht der Diplomarbeiten-VM im VMware vSphere Client	44
Abbildung 31: SourceTree.....	45
Abbildung 32: GitKraken-Logo (Zazzle, 2017)	45

Abbildung 33: Microsoft Office-Logo (Microsoft, 2017).....	46
Abbildung 34: Dropbox-Logo (Copernic Forum, 2017).....	46
Abbildung 35: Leere GIMP-Anwendung nach dem Start.....	47
Abbildung 36: Logo Adobe Photoshop (wikipedia.org, 2017).....	47
Abbildung 37: FileZilla Client.....	48
Abbildung 38: Telegram-Logo (Google Play, 2017)	48
Abbildung 39: Postman-Logo (http://blog.getpostman.com, 2017).....	49
Abbildung 40: Funktionale Idee Absolventen & Firmen.....	55
Abbildung 41: Funktionale Idee Administrator	56
Abbildung 42: Use-Case-Diagramm Events.....	57
Abbildung 43: Use-Case-Diagramm Stellenangebote	57
Abbildung 44: Use-Case-Diagramm Mitgliedsbeitrag.....	58
Abbildung 45: Use-Case-Diagramm Erfolgsgeschichten	58
Abbildung 46: P@BS Connect Architektur (clker.com, 2017), (centratrip.com, 2017), (wikipedia.org, 2017), (clker.com, 2017), (wikipedia.org, 2017), (memz.co, 2017), (fedoraproject.org, 2017) (MBT Check, 2017) (outdatedbrowser.com, 2017)	59
Abbildung 47: Webportal Entwurf Login-Seite.....	61
Abbildung 48: Webportal Entwurf Side-Menu.....	62
Abbildung 49: Webportal Entwurf Startseite	63
Abbildung 50: Webportal Entwurf Profil.....	64
Abbildung 51: Webportal Entwurf Profil (Administratorfunktionen).....	65
Abbildung 52: Webportal Entwurf Jobs - Stelleninserat hochladen (Administrator & Firma).....	69
Abbildung 53: Webportal Entwurf Jobs - Administratorfunktionen.....	70
Abbildung 54: Webportal Entwurf Events- und News-Übersicht.....	71
Abbildung 55: Webportal Entwurf Events bzw. News - Einzelansicht	72
Abbildung 56: Webportal Entwurf Events bzw. News erstellen und bearbeiten	73
Abbildung 57: Webportal Entwurf Erfolgsgeschichten	74
Abbildung 58: Webportal Entwurf Erfolgsgeschichten - Administratorfunktionen.....	75
Abbildung 59: Webportal Entwurf Mitgliedsbeitrag.....	76
Abbildung 60: Datenmodell	78
Abbildung 61: Entwurf HTL-Erfolgsgeschichte	102
Abbildung 62: Einstieg in das Webportal.....	112
Abbildung 63: Startseite interner Bereich	113

Abbildung 64: Stammdatenverwaltung.....	114
Abbildung 65: News-Übersicht	115
Abbildung 66: News-Detailansicht.....	116
Abbildung 67: offene Jobinserte	116
Abbildung 68: Event-Übersicht.....	117
Abbildung 69: Event-Detailansicht	117
Abbildung 70: Hochladen einer Erfolgsstory	118
Abbildung 71: Stelleninserat hochladen	119
Abbildung 72: Verwalten der persönlichen Stelleninserate	120
Abbildung 73: Administrator-Benutzerübersicht	122
Abbildung 74: Anzeigen aller Benutzer-E-Mails	123
Abbildung 75: Erstellen von neuen News.....	124
Abbildung 76: Erstellen von neuen Events.....	125
Abbildung 77: Verwaltung der Erfolgsstories.....	126
Abbildung 78: E-Mail-Benachrichtigung HTL-Erfolgsstory.....	126
Abbildung 79: Auszug E-Mail-Benachrichtigung Jobinserte	127
Abbildung 80: E-Mail-Benachrichtigung neuer Newseintrag	127

17.6 Tabellenverzeichnis

Tabelle 1: JPA-Bestandteile (vgl. tutorialspoint, 2017)	30
Tabelle 2: Attribute PCUser	82
Tabelle 3: Attribute PCNews	83
Tabelle 4: Attribute PCEvent	84
Tabelle 5: Attribute PCComment	85
Tabelle 6: Attribute PCParticipation	85
Tabelle 7: Attribute PCStory	87
Tabelle 8: Attribute PCFileUpload	88
Tabelle 9: Attribute PCTransaction	89
Tabelle 10: Oftmals verwendete http-Statuscodes (vgl. wikipedia.org, 2017)	91