

Elternsprechtags- applikation



ESTA

Dokumentation

Ein Projekt von

Christian Ehringfeld
David Mock
Matthias Unterbusch

Inhaltsverzeichnis

1 Thematisierung und Projektumfeld	4
1.1 Thematisierung	4
1.2 Projektumfeld	4
1.2.1 Projektteam	4
1.2.2 Auftraggeber	4
1.2.3 Softwareumgebung	4
2 Planung	4
2.1 Zeitplanung	4
2.2 Ressourcen	5
2.3 Kosten	5
3 Lösungskonzept	6
3.1 Ziele	6
3.1.1 Fehlerfreie Software	6
3.1.2 Barrierefreiheit	6
3.1.3 Usability	7
3.1.4 Datenschutz	7
3.2 Funktionalitäten	7
3.3 Diagramme	7
3.3.1 GANTT Diagramm	7
3.3.2 UML-Anwendungsfalldiagramm	9
3.3.3 Klassendiagramme	9
3.4 Datenbankmodell	13
4 Realisierung	13
4.1 Gestaltung der grafischen Oberfläche	13
4.1.1 Anmeldeseite	14
4.1.2 Registrierung	15
4.1.3 Terminliste	16
4.1.4 Lehrersuche	17
4.1.5 Termineintragung	18
4.1.6 Benutzerverwaltung	19

4.2 Umsetzung	19
4.3 Probleme	20
5 Tests	20
5.1 Testkonzepte	20
5.2 Testfälle	20
5.2.1 Rolle Administrator	20
5.2.2 Rolle Eltern	21
5.2.3 Rolle Lehrer	21
5.2.4 Rolle Verwaltung	21
5.3 Testprotokolle	21
6 Projektergebnis	22
6.1 Soll/Ist Vergleich	22
6.1.1 Zeiten	22
6.1.2 Software	22
6.2 Verlauf	22
6.3 Ergebnis und Fazit	22
6.4 Ausblick	23
7 Eidesstattliche Erklärung	23
8 Anhänge	
8.1 Lastenheft	
8.2 Pflichtenheft	
8.3 Angebot	
8.4 Installationsanleitung	
8.5 Benutzerhandbuch	
8.6 User Manual	
8.7 Testprotokolle	
8.8 Teamprotokolle	

1 Thematisierung und Projektumfeld

1.1 Thematisierung

Im Rahmen eines Schulprojekts der Brühlwiesenschule Hofheim sollte eine Elternsprechtagssoftware für Eltern entwickelt werden, um den Ablauf des Elternsprechtags zu optimieren. Es wurde entschieden eine Webseite zu erstellen.

1.2 Projektumfeld

1.2.1 Projektteam

Name	Rolle	Firma
Christian Ehringfeld	Back-End	Bundeskriminalamt
David Mock	Front-End	Bundesamt für Wirtschaft und Ausfuhrkontrolle
Matthias Unterbusch	Projektmanager	Hilscher Gesellschaft für Systemautomation mbH

1.2.2 Auftraggeber

Der Auftraggeber ist die Brühlwiesenschule in Hofheim. Dabei ist der Ansprechpartner Herr Wild. Der Betreuer des Projekts ist Herr Grüning.

1.2.3 Softwareumgebung

Da die Software als Webapplikation realisiert werden sollte, fiel die Wahl der Sprache auf PHP (Back-End) und HTML (Front-End). Um einige zusätzliche Funktionen auf der Webseite bereitzustellen, kommt außerdem JavaScript zum Einsatz. Ferner wurde ein Apache Webserver benutzt.

2 Planung

2.1 Zeitplanung

Da das Projekt zu umfangreich war, um es in drei Wochen in der Schule zu entwickeln, wurde an dieser Stelle mit einem normalen Arbeitstag von acht Stunden geplant. Bei drei Personen ergab das eine Arbeitszeit von 120 Stunden in der Woche und eine Gesamtarbeitszeit von 360 Stunden. Das Gesamtergebnis der folgenden Tabelle ist etwas geringer, da in der Planung etwas Luft für unvorhergesehene Probleme gelassen wurde. Eine Visualisierung in einem GANTT Diagramm ist in Kapitel 3.3.1 zu finden.

Stunden	Aufgabe	Durchführung von
40	Erstellung des Pflichtenhefts	Matthias Unterbusch
8	Erstellung eines Angebots	Matthias Unterbusch
16	Benutzerhandbuch (mehrsprachig)	Matthias Unterbusch
8	Erstellen einer Installationsanweisung	Matthias Unterbusch
32	Erstellung einer Dokumentation	Matthias Unterbusch
8	Datenbankmodellierung	Christian Ehringfeld
16	Benutzerverwaltung	Christian Ehringfeld
32	Terminverwaltung	Christian Ehringfeld
16	Administrationsbereich	Christian Ehringfeld
8	Datenimport	Christian Ehringfeld
16	Rechteverwaltung	Christian Ehringfeld
24	Erstellen von Mock Ups	David Mock
24	Erstellen von Prototypen	David Mock
16	Lehrerfiltererstellung	David Mock
8	Formelementanpassung	David Mock
32	Integration von Prototypen	David Mock
48	Systemtest	Alle
352	Gesamt	

2.2 Ressourcen

Als Ressourcen für das Projekt werden nur die Sachmittel benötigt. Diese umfassen einen normalen Bildschirmarbeitsplatz. Dazu gehören Tische, Stühle und PCs, die komplett von der Schule zur Verfügung gestellt wurden. Zudem wurde eine Verbindung mit ins Internet mit Hilfe eines WLAN-Hotspots zur Verfügung gestellt. Da das Projekt auch einige Arbeitszeit zu Hause erforderte und die PCs der Schule durch eine starke Netzauslastung sehr langsam waren, haben sich die Mitarbeiter darauf verständigt, dass mit privaten Notebooks gearbeitet wird. Außerdem wurden für Tests ein Webserver und eine Versionsverwaltung verwendet. Dazu wurde das webbasierte Projektmanagementtool Redmine benutzt, welches schon von einem Mitglied des Projektteams konfiguriert wurde.

2.3 Kosten

Pos	Anzahl	Einheit	Beschreibung	Netto
1	40	Stunden	Erstellung des Pflichtenhefts	2400
2	8	Stunden	Erstellung eines Angebots	480
3	16	Stunden	Benutzerhandbuch (mehrsprachig)	960
4	8	Stunden	Erstellen einer Installationsanweisung	480
4	32	Stunden	Erstellung einer Dokumentation	1920
5	8	Stunden	Datenbankmodellierung	480
6	16	Stunden	Benutzerverwaltung	960
7	32	Stunden	Terminverwaltung	1920
8	16	Stunden	Administrationsbereich	960
9	8	Stunden	Datenimport	480
10	16	Stunden	Rechteverwaltung	960

11	24	Stunden	Erstellen von Mock Ups	1440
12	24	Stunden	Erstellen von Prototypen	1440
13	16	Stunden	Lehrerfiltererstellung	960
14	8	Stunden	Formelementanpassung	480
15	32	Stunden	Integration von Prototypen	1920
16	48	Stunden	Systemtest	2880
	352	Stunden		
2% Skonto bei Zahlung innerhalb von 14 Tagen				
Gesamt Netto				21120,00
19 % Umsatzsteuer				4012,80
Gesamt Brutto				25132,80

3 Lösungskonzept

3.1 Ziele

Ziel ist es eine Elternsprechtagssoftware zu entwickeln, um die Handhabung des Elternsprechtags zu vereinfachen. Die Elternsprechtagssoftware soll es ermöglichen online Termine für den Elternsprechtag zu vereinbaren. Dazu wird eine Webseite entworfen, für die lediglich ein Internetzugang nötig ist. Dabei wird berücksichtigt, dass die Zielgruppe in der Regel Berufsfremd ist.

3.1.1 Fehlerfreie Software

Ein Hauptziel ist eine fehlerfreie Software. Da der Elternsprechtag ein offizieller Anlass ist und die Schule nach außen hin repräsentiert, dürfen an diesem Tag natürlich keine Unstimmigkeiten auftauchen, die auf die Elternsprechtagssoftware zurückzuführen sind.

3.1.2 Barrierefreiheit

Ein weiteres Ziel ist die Barrierefreiheit. Dabei ist es sehr wichtig, dass jeder, auch Menschen mit Benachteiligungen, die Webseite in vollem Umfang nutzen können. Dabei kann vor allem mit einem einfachen Design, mit möglichst starken Kontrasten darauf geachtet werden, dass auch Menschen mit einer Sehschwäche die Website in vollem Umfang benutzen können.

Ein weiterer Punkt der Barrierefreiheit ist die Verwendung von JavaScript. JavaScript muss zusätzlich auf jedem System installiert werden und erhöht somit die Anforderungen an die Benutzer. Daher funktioniert die gesamte Webseite auch ohne JavaScript. JavaScript stellt dabei für die Benutzer lediglich einige zusätzliche Funktionen zur Verfügung. Ein Beispiel dafür ist das zusätzlich einblendbare Menü, welches ein besseres Design hat. Da es nicht immer eingeblendet ist, steht der zusätzliche Platz für den Rest der Website zur Verfügung.

Ein weiteres Feature, welches nur mit JavaScript genutzt werden kann, gibt es bei der Termineintragung. Mit aktivierten JavaScript können freie Termine angeklickt werden, worauf die Felder Datum und Uhrzeit automatisch ergänzt werden. Wenn kein JavaScript aktiviert ist müssen diese Felder mit Hilfe von Drop Menüs manuell ausgefüllt werden.

3.1.3 Usability

Die Benutzerfreundlichkeit soll auf jeden Fall gewährleistet sein. Dazu trägt vor allem ein schlichtes Design der Seite bei, wobei vor allem auf klare Kontraste geachtet wurde. Außerdem wird auf eine sehr einfache Menüstruktur geachtet. Beim ersten Login werden die Eltern direkt aufgefordert ein Kind einzugeben und es muss nicht erst im Profil mühselig nachgetragen werden. Des Weiteren werden, falls Termine gemacht wurden, diese direkt im Startbildschirm angezeigt und es muss nicht jeder Lehrer bei dem Termine gemacht wurden, einzeln überprüft werden.

3.1.4 Datenschutz

Ein wichtiger Punkt ist der Datenschutz der Schüler, Eltern und Lehrer. Deshalb werden die Zeitpläne der Lehrer den Eltern nicht komplett angezeigt, um nicht jedem alle Informationen zugänglich zu machen. Die vergebenen Termine werden nur als belegt gekennzeichnet und nicht genau angezeigt wer welchen Termin gebucht hat. Außerdem werden die Passwörter in der Datenbank verschlüsselt gespeichert. Dabei werden gesalzene sha512 Hashes verwendet.

Da die Daten maskiert werden, ist eine SQL-Injection nicht möglich. Da alle Ausgaben maskiert werden ist Cross Site Scripting ebenfalls nicht möglich.

3.2 Funktionalitäten

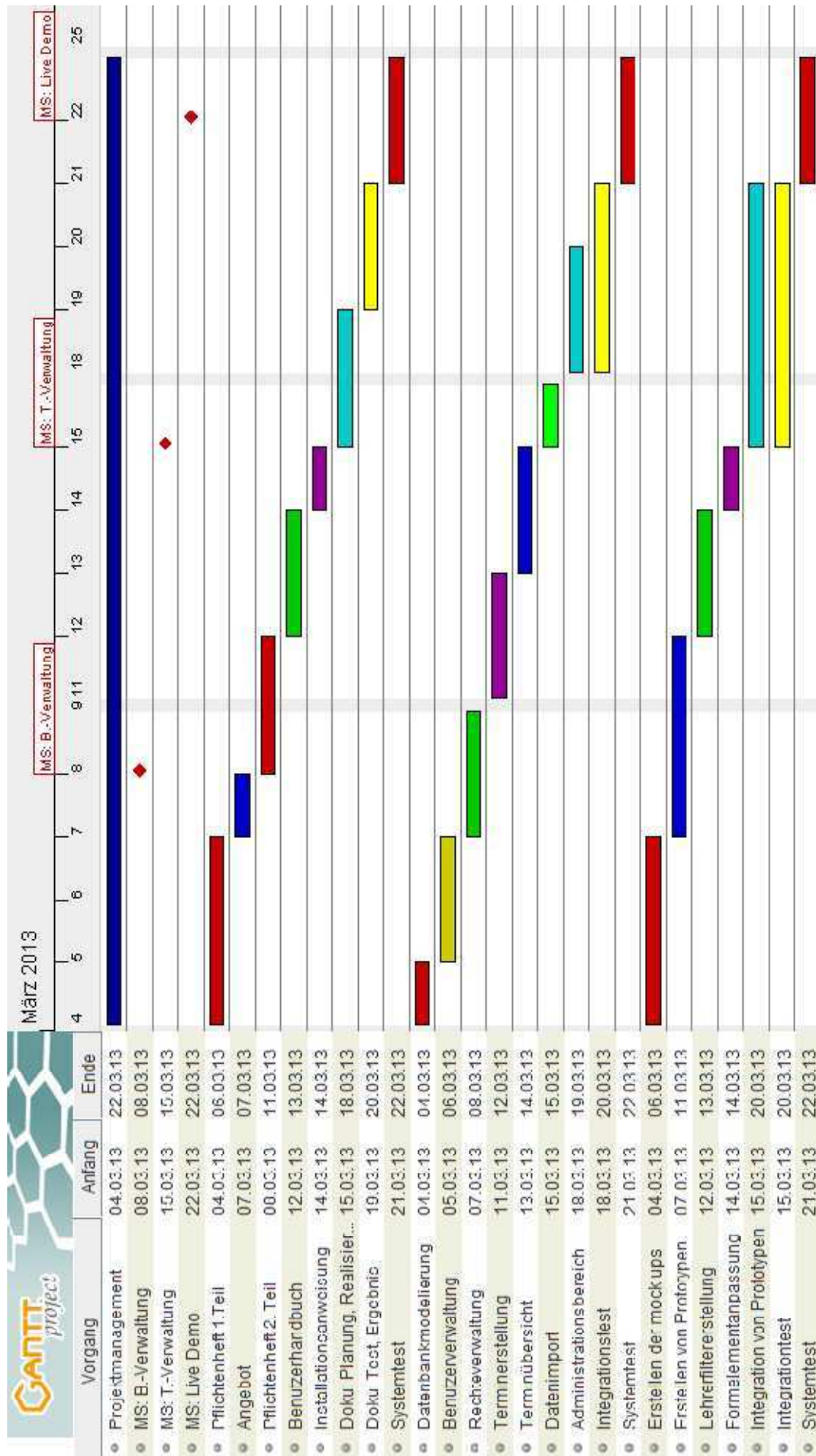
Die Umsetzung der Software wurde anhand der Muss-, Soll-, und Kannkriterien realisiert. Der erste Schritt war die Modellierung einer Datenbank, um alle anderen Funktionen zu ermöglichen. Danach wurde die Möglichkeit geschaffen sich zu registrieren, um einen internen Bereich zu schaffen. Damit wurde die Grundlage für den Datenschutz geschaffen. Zeitgleich wurde das Design für die Seite fertiggestellt und mit der Erstellung der Prototypen begonnen. Anschließend wurde mit der Rechteverwaltung begonnen, um mehrere Rollen zu realisieren.

Im Anschluss daran wurde das Eintragen von Terminen umgesetzt. Dazu wurde ein Lehrerfilter erstellt, um einen Lehrer auszuwählen. Nachfolgend wurden die Zeitpläne für Lehrer erstellt und das Termineintragen von der Rolle Eltern möglich gemacht. Währenddessen wurden die Prototypen integriert und Formelemente angepasst. Zum Schluss wurde der Datenimport in die Software integriert. Die Accounts der Lehrer können automatisch aus einer .csv Datei erstellt werden.

3.3 Diagramme

3.3.1 GANTT Diagramm (Abb. 1: GANTT Diagramm)

Das GANTT Diagramm wurde mit der Software GANTT-Projekt Version 2.6 erstellt.



3.3.2 UML-Anwendungsfalldiagramm

Das UML-Diagramm wurde mit der Software ArgoUML Version 0.32.2 erstellt.

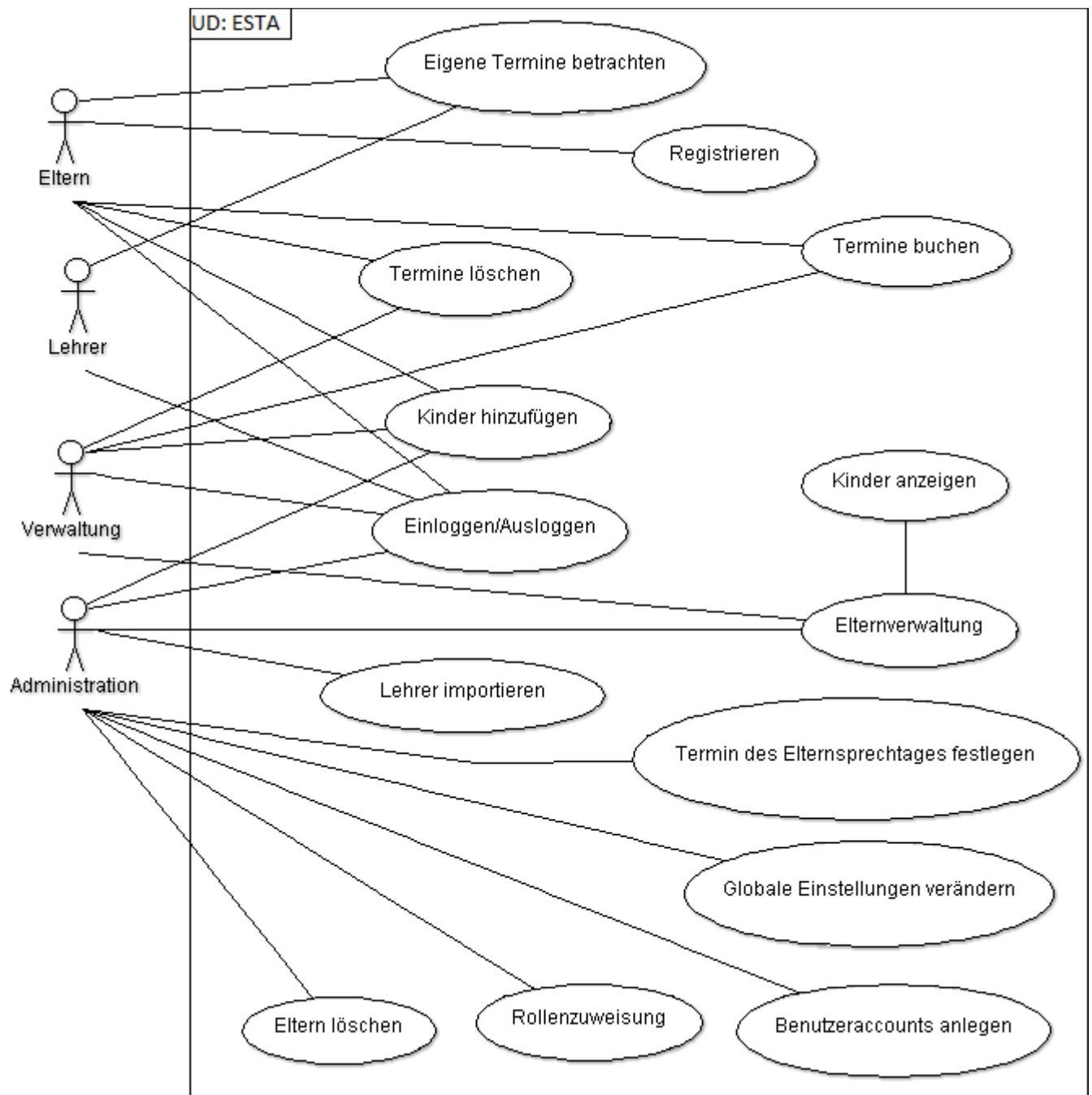
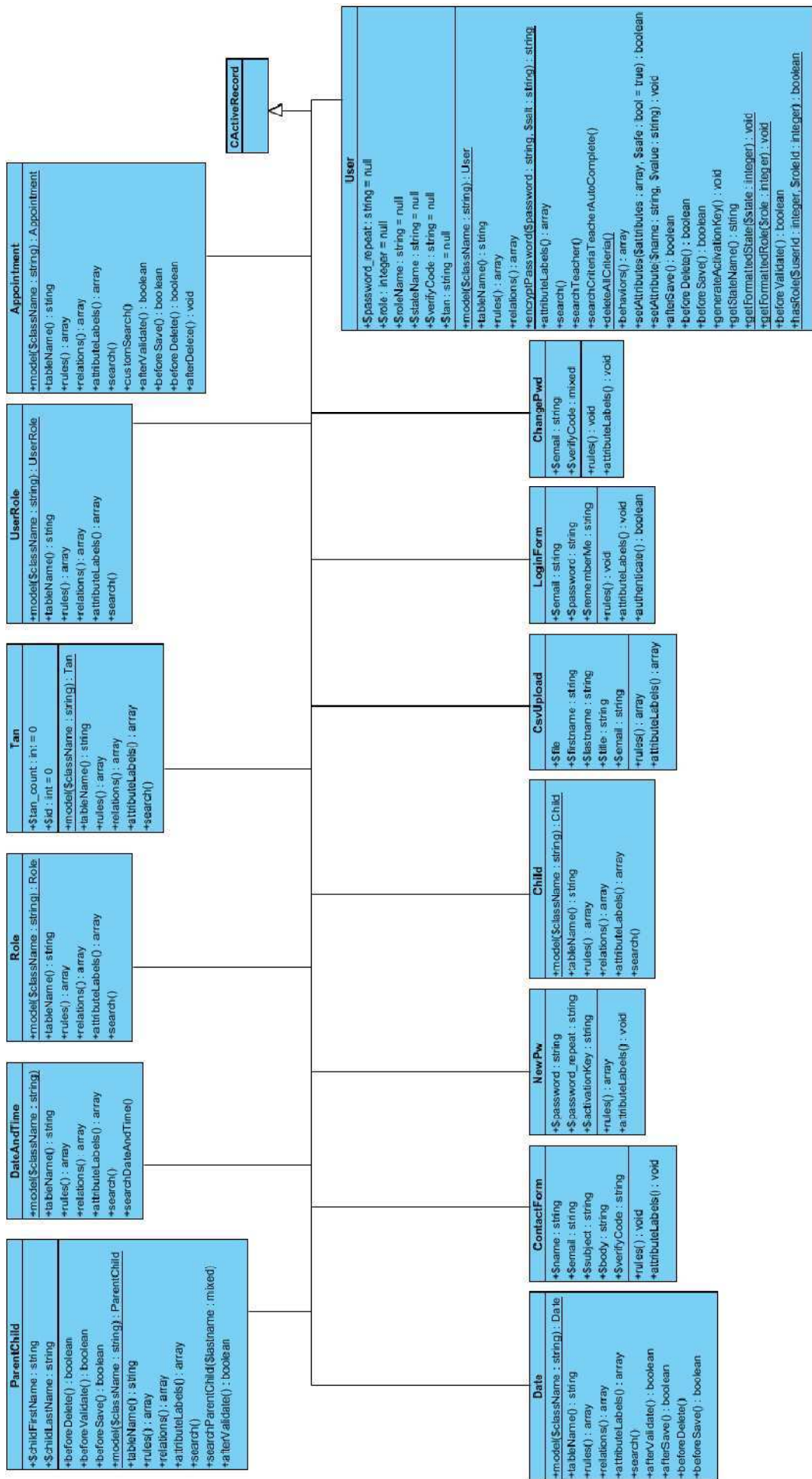


Abb. 2: UML-Anwendungsfalldiagramm

3.3.3 Klassendiagramme (Abb. 3-5: Klassendiagramme)

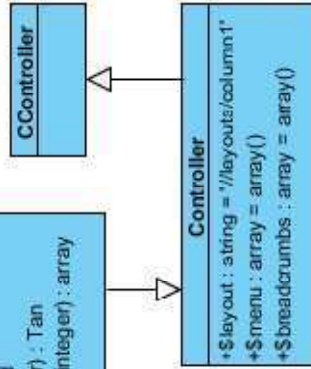
Die Klassendiagramme wurden mit der Software Visual Paradigm Version 10.1 erstellt.



AppointmentController
+\$layout : string = '//layouts/column2' +filters() : array +accessRules() : array +actionView(\$id : integer) : void +actionCreate() : void +actionGetTeacher() : void +actionMakeAppointment(\$teacher : integer) : void +actionUpdate(\$id : integer) : void +actionDelete(\$id : integer) : void +actionIndex() : void +actionAdmin() : void +loadModel(\$id : integer) : Appointment #performAjaxValidation(\$model : Appointment) void +getChildren(\$userId : integer) : array +getDatesWithTimes(\$dateMax : integer) : array +isAppointmentAvailable(\$teacher : integer, \$dateAndTimeId : integer) : array

DateController
+\$layout : string = '//layouts/column2' +filters() : array +accessRules() : array +actionSearch(\$term : string) : void +actionView(\$id : integer) : void +actionCreate() : void +actionUpdate(\$id : integer) : void +actionDelete(\$id : integer) : void +actionIndex() : void +actionAdmin() : void +loadModel(\$id : integer) : Date #performAjaxValidation(\$model : Date) : void

TanController
+\$layout : string = '//layouts/column2' +filters() : array +accessRules() : array +actionCreate() : void +actionGetTans() : void +loadModel(\$id : integer) : Tan +generateTan(\$count : integer) : array

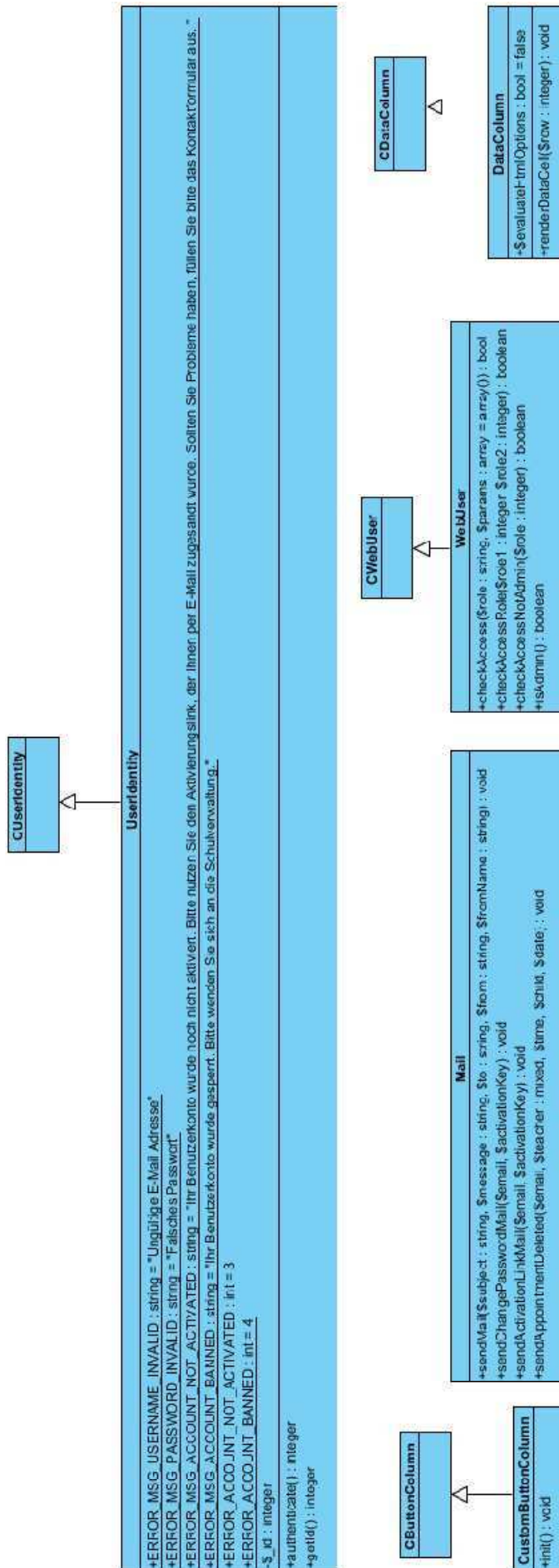


UserController
+\$layout : string = '//layouts/column2' +filters() : array +actions() : array +accessRules() : array +actionSearch(\$role : integer, \$term : string) : void +actionDeleteAll() : void +actionView(\$id : integer) : void +actionAccount() : void +actionActivate(\$activationKey : string) : void +actionImportTeachers() : void +encodingString(\$ioEncode : string) : string +actionNewPw() : void +actionChangePw(\$id : integer) : void +actionCreate() : void +actionUpdate(\$id : integer) : void +actionDelete(\$id : integer) : void +actionAdmin() : void +loadModel(\$id : integer) : User #performAjaxValidation(\$model : User) : void

UserRoleController
+\$layout : string = '//layouts/column2' +filters() : array +accessRules() : array +actionView(\$id : integer) : void +actionCreate() : void +actionUpdate(\$id : integer) : void +actionDelete(\$id : integer) : void +actionAdmin() : void +loadModel(\$id : integer) : UserRole #performAjaxValidation(\$model : UserRole) : void

ParentChildController
+\$layout : string = '//layouts/column2' +filters() : array +accessRules() : array +actionSearch(\$term : string) : void +actionView(\$id : integer) : void +actionIndex() : void +actionCreate() : void +actionDelete(\$id : integer) : void +actionAdmin() : void +checkUser(\$parentChildId : integer) : boolean +loadModel(\$id : integer) : ParentChild #performAjaxValidation(\$model : ParentChild) : void

SiteController
+actions() : void +actionIndex() : void +actionError() : void +actionContact() : void +actionLogin() : void +actionLogout() : void



3.4 Datenbankmodell

Das Datenbankmodell wurde mit MySQL Workbench Version 5.2.47 erstellt.

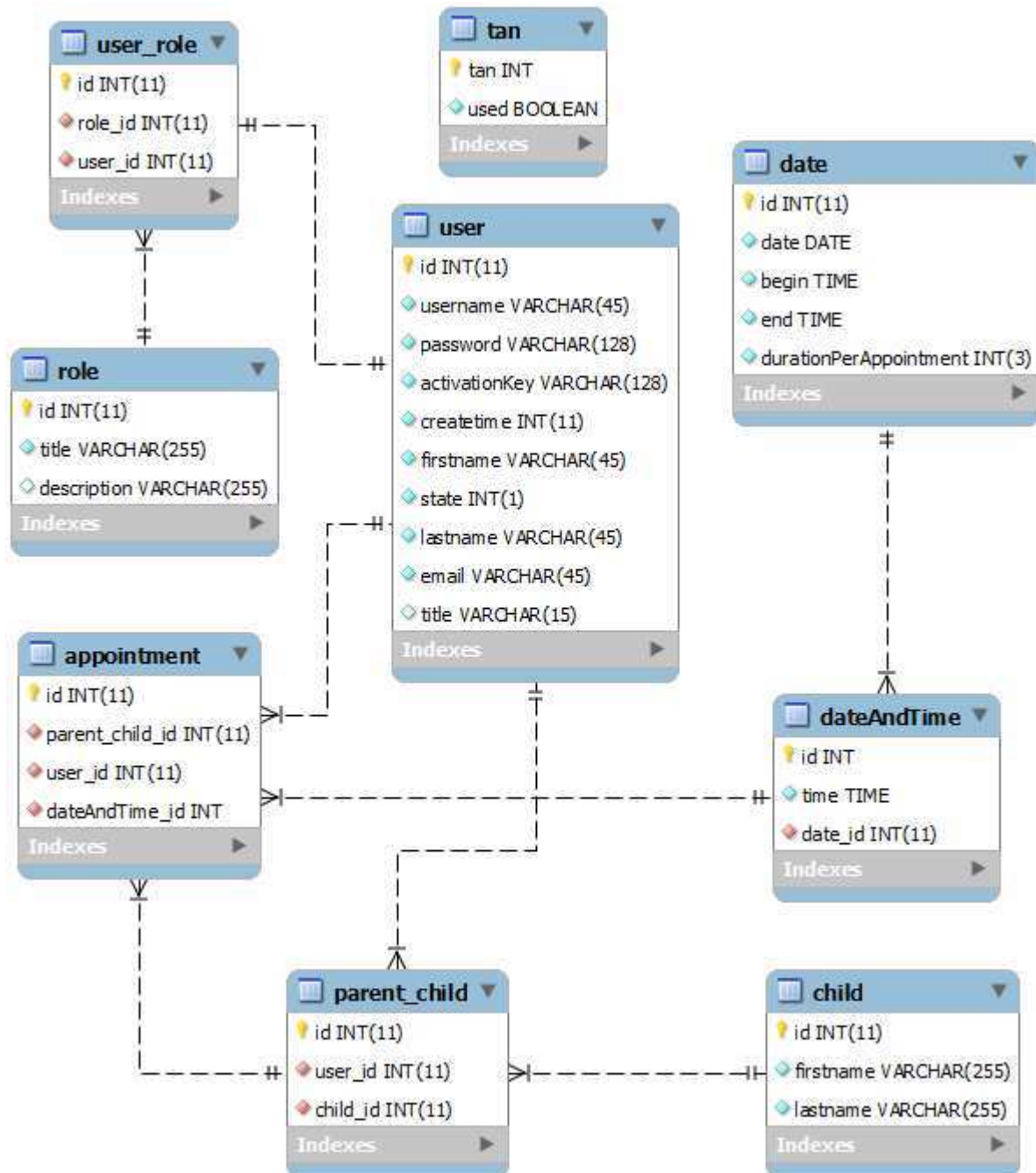


Abb. 6: Datenbankstruktur

4 Realisierung

4.1 Gestaltung der grafischen Oberfläche

Bei der Gestaltung der grafischen Oberfläche wurde vor allem auf ein einfaches und schlichtes Design geachtet. Die Eltern sollen nicht durch ein zu aufdringliches Design

verunsichert werden. Dabei wurde vor allem auf klare Kontraste viel Wert gelegt. Es wurde nur ein schlichtes Logo für die Software entworfen, welches sich eher unauffällig auf der Webseite präsentiert. Da es sich um eine offizielle Schulhomepage handelt, wurde ebenfalls das Logo der Brühlwiesenschule in die Webseite integriert, welches sich ähnlich wie das Logo der Elternsprechtssoftware im Hintergrund hält.

Um die Benutzerfreundlichkeit weiter zu verbessern, gibt es auf vielen Seiten eine Box, in welcher eine kurze Beschreibung vorhanden ist, welche Aktionen auf dieser Seite möglich sind und wie diese zu handhaben sind. Weiterhin wurde zu jedem Feld eine Beschreibung hinzugefügt, die dem Benutzer angibt, welche Information in dieses Feld hingeschrieben werden muss. Zudem wurden die Seiten nicht mit Funktionen überladen und relativ simpel gehalten.

4.1.1 Anmeldeseite



Liebe Eltern,

Willkommen auf der elektronischen Elternsprechtsplattform der Brühlwiesenschule Hofheim.

Melden Sie sich an oder registrieren Sie sich um ihre Termine einzusehen oder neue Termine zu vereinbaren.

Login

E-Mail

Passwort

☐ Anmeldedaten merken

Login

[Passwort vergessen?](#)

[Benötigen Sie einen neuen Zugang?](#)

[Klicken Sie hier.](#)

Abb. 7: Anmeldeseite

4.1.2 Registrierung


Die Registrierungsseite kann von der Anmeldeseite aus erreicht werden und ist ebenfalls für nicht registrierte Benutzer sichtbar.



Registrierung

Geben Sie ihre E-Mail-Adresse und ein Passwort ein um sich im System zu registrieren. Sie sollten innerhalb weniger Minuten eine E-Mail empfangen, die einen Link enthält mit dem Sie ihre Registrierung abschließen können.

Vorname	<input type="text"/>
Nachname	<input type="text"/>
E-Mail	<input type="text"/>
Passwort	<input type="password"/> <small>Mindestlänge 8 Zeichen</small>
Passwort wiederholen	<input type="password"/>
Tan	<input type="text"/>

 [Neuen Code erzeugen](#)

Sicherheitscode	<input type="text"/>
-----------------	----------------------

Bitte geben Sie den im Bild angezeigten Sicherheitscode ein.

[Registrieren](#)

[Zurück zur Startseite](#)

Abb. 8: Registrierung

4.1.3 Terminliste


Die Terminliste wird direkt nach dem Login angezeigt, falls bereits Termine vereinbart wurden.

Elternsprechtag

der

Holheim

bws

 Menü

Ihre Termine

#24	Am 20.10.2014 um 16:00 <i>Ihr Kind</i> David Mock <i>bei</i> Edwin Gram	Termin löschen
#25	Am 20.10.2014 um 17:00 <i>Ihr Kind</i> Christian Ehrlingfeld <i>bei</i> Peter Grüning	Termin löschen
#26	Am 20.10.2014 um 18:00 <i>Ihr Kind</i> Matthias Unterbusch <i>bei</i> Thomas Maul	Termin löschen

Abb. 9: Terminliste

4.1.4 Lehrersuche

Die Lehrersuche kann über das Menü aufgerufen werden und ist eine Filterfunktion für Lehrer, um die Termineintragung zu vereinfachen.

Elternsprechtag

der

Menü

Klicken Sie auf einen Buchstaben um sich alle Lehrer, deren Name mit diesem Buchstaben beginnt, anzeigen zu lassen.
Drücken Sie auf den Stern um eine Liste mit allen Lehrern zu erhalten.

Sie können alternativ auch den Namen in das Suchfeld eingeben um alle zutreffenden Möglichkeiten zu sehen.

Wenn Sie den richtigen Lehrer gefunden haben, klicken Sie einfach auf das Uhrensymbol neben seinem Namen um zu der Terminvereinbarung des Lehrers zu gelangen.

A B C D E F G H I J K L M N O

P Q R S T U V W X Y Z *

Lehrername

Zeige Ergebnisse 1-20 von 86.

Titel	Vorname	Nachname	
	Vjeka Maria	Adam	
	Gregor	Adler	
	Michael Günter	Albers	
	Elisabeth Maria	Artner-Peill	

Abb. 10: Lehrersuche

4.1.5 Termineintragung

Es wird der Zeitplan des Lehrers angezeigt und mit einem Klick auf einen verfügbaren Termin erscheint die untere Box, bei der Eltern noch die Möglichkeit haben, ein Kind auszuwählen falls zwei oder mehr ihrer Kinder auf die Schule gehen. Danach können sie einen Termin buchen, der danach als nicht verfügbar angezeigt wird.

 **Menü**

17:00	VERFÜGBAR	VERFÜGBAR
17:15	BELEGT	BELEGT
17:30	BELEGT	VERFÜGBAR
17:45	BELEGT	BELEGT
18:00	BELEGT	VERFÜGBAR
18:15	VERFÜGBAR	BELEGT
18:30	VERFÜGBAR	BELEGT
18:45	BELEGT	VERFÜGBAR
19:00	VERFÜGBAR	VERFÜGBAR

Termin

Mit

[LEHRER]

Am

[DATUM]

Um

[UHRZEIT]

Für

[KIND1]

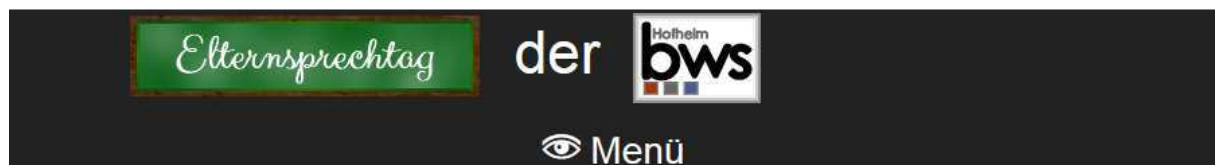
▼

Bestätigen

Abb. 11: Zeitplan

4.1.6 Benutzerverwaltung

Die Benutzerverwaltung ist nur für Administratoren sichtbar und kann zum Erstellen und Bearbeiten der Benutzer verwendet werden.



Benutzerverwaltung

Zeige Ergebnisse 1-20 von 90.

ID	E-Mail	Vorname	Nachname	Titel	Status	Rolle	
1	admin@admin.de	admin	admin		Aktiv	Administration	
94	verwaltung@verwaltung.de	Verwaltung	Verwaltung		Aktiv	Verwaltung	
110	lehrer@lehrer.de	Lehrer	Lehrer		Aktiv	Eltern	
111	eltern@eltern.de	Eltern	Eltern		Aktiv	Eltern	
113	v.adam@uni-mainz.de	Vjeka Maria	Adam		Aktiv	Lehrer	
114	g.adler@bws-hofheim.de	Gregor	Adler		Aktiv	Lehrer	
115	m.albers@bws-hofheim.de	Michael Günter	Albers		Aktiv	Lehrer	
116	artner@gmx.de	Elisabeth Maria	Artnr-Peill		Aktiv	Lehrer	
117	i.becker@bws-hofheim.de	Ingeborg	Becker		Aktiv	Lehrer	
118	m.becker@bws-hofheim.de	Manuela	Becker		Aktiv	Lehrer	
119	h.benz@bws-hofheim.de	Hubert	Benz	Dr.	Aktiv	Lehrer	
120	christophberg@gmx.de	Christoph	Berg		Aktiv	Lehrer	
121	w.bill@bws-hofheim.de	Wolfgang	Bill		Aktiv	Lehrer	
122	v.blam@bws-hofheim.de	Volker	Blam		Aktiv	Lehrer	

Abb. 12: Benutzerverwaltung

4.2 Umsetzung

Um das Projekt umzusetzen, wurde das Yii Framework benutzt. Das Framework hat die BSD Lizenz und ist Opensource. Daher kann dieses Framework ohne Bedenken verwendet werden.

Des Weiteren wird als Programmiersprache PHP verwendet, die mit der PHP Lizenz lizenziert und ebenfalls Opensource ist. Die weiteren verwendeten Sprachen wie HTML und JavaScript sind Lizenzfrei. Als Entwicklungsumgebung wurde Netbeans verwendet.

Das relationale Datenbankverwaltungssystem MySQL, welche die GPL Lizenz inne hat, ist ebenfalls Opensource. Wir verwenden als freie Storage-Engine für MySQL InnoDB.

Es werden auch mehrere Icons verwendet, die alle zu der CC-BY-SA 3.0 Lizenz gehören. Diese Icons sind also frei nutzbar.

4.3 Probleme

Ein Problem entstand mit der Kodierung in der Datenbank. Zunächst wurde UTF8-General-CI benutzt. Dieser Zeichensatz ist case-insensitive und daher gut für eine Datenbank geeignet. Außerdem ist diese Kodierung schneller, da nicht auf Bitebene verglichen wird. Das führte zu Problemen bei der Auswahl der Lehrer, die mit Umlauten beginnen, von denen zu unterscheiden, die mit dem entsprechenden Vokal beginnen, da UTF8-General-CI Umlaute als normale Vokale behandelt, z.B. Ä wie A. Aus diesem Grund wurde der Zeichensatz UTF8 binär gewählt, bei dem auf Bitebene verglichen wird, und somit Umlaute korrekt verarbeiten kann.

Ein weiteres Problem waren Elternsprechtage mit unterschiedlicher Termindauer. Dabei gab es Probleme mit der korrekten Anzeige. Bei dem ursprünglichen Design wurden beide Tage in einer Tabelle nebeneinander angezeigt, was nur möglich ist, wenn die Termindauer identisch ist. Dazu musste das Design der Termineintragung geändert werden, sodass bis zu drei unterschiedliche Tabellen angezeigt werden.

5 Tests

5.1 Testkonzepte

Während der Programmierung wurden einzelne Teile des Programms immer wieder einem Modultest unterzogen. Dieser wurde von den Entwicklern mit Hilfe des Whiteboxtestverfahrens durchgeführt. Außerdem wurde bei der Zusammenführung des Front-Ends und Back-Ends ein Integrationstest von den Programmierern durchgeführt.

Um die Software so fehlerfrei wie möglich zu gestalten, wurde zum Schluss noch ein Systemtest durchgeführt. Dieser wurde als Blackboxtest durchgeführt und vom Projektmanager durchgeführt.

5.2 Testfälle

5.2.1 Rolle Administrator

Als erstes wurde der Administrationsbereich getestet. Das geschieht deshalb, da zunächst ein Elternsprechtag erstellt und eine Lehrerliste eingelesen werden muss, damit die Software wie geplant funktionieren kann.

Als erstes wurde ein neuer Elternsprechtag erstellt. Beim Eintragen des Datums und der Zeiten ist ein sehr spezielles Format erforderlich. Aus diesem Grund werden mehrere falsche Eingaben getätigt, um zu überprüfen, ob ein falsches Datum festgelegt werden kann. Zum Schluss wird ein korrektes Datum eingegeben, um zu testen, ob ein Elternsprechtag korrekt angelegt wird.

Danach wird das Einlesen der Lehrerliste getestet. Dabei wird versucht eine falsche Datei auszuwählen. Anschließend wird eine korrekte Datei eingelesen und geprüft, ob die Lehrerdaten korrekt eingelesen werden.

Da jede höhere Rolle automatisch die Rechte der darunterliegenden hat, mit der Ausnahme, dass Lehrer keine Termine vereinbaren dürfen, werden alle folgenden Tests der Kapitel 5.2.2 – 5.2.4 auch bei der Rolle Administrator durchgeführt.

5.2.2 Rolle Eltern

Bei den Eltern wird zunächst das Eintragen von Kindern überprüft. Da an dieser Stelle keine Überprüfung stattfindet, ob die Kinder in der Datenbank vorhanden sind, können an dieser Stelle keine „falschen“ Eingaben gemacht werden. An dieser Stelle muss nur geprüft werden, wie viele Kinder maximal eingetragen werden können und ob das Löschen der Kinder korrekt funktioniert.

Anschließend wird die Verwaltung des eigenen Accounts getestet. Dazu wird das Menü „Ihr Account“ aufgerufen und getestet, welche Daten verändert werden können. Danach werden mehrere Daten geändert und gespeichert.

Danach wird die Termineintragung getestet. Dabei wird geprüft, ob der Lehrerfilter korrekt funktioniert oder Fehler auftreten. Danach wird ein Lehrer ausgewählt und versucht einen bereits belegten Termin einzutragen. Zum Schluss wird ein freier Termin reserviert und auf die Seite „Ihre Termine“ gewechselt, um zu überprüfen, ob der Termin korrekt eingetragen wurde und testweise ein Termin wieder gelöscht.

5.2.3 Rolle Lehrer

Bei den Lehrern wird ebenfalls die Accountverwaltung des eigenen Accounts. Dabei wird wieder getestet, welche Daten verändert werden können. Dann werden wieder mehrere Daten geändert, gespeichert und geprüft, ob alles korrekt übernommen wurde.

Des Weiteren wird überprüft, ob der Zeitplan des Lehrers korrekt angezeigt wird und ausgedruckt werden kann.

Zum Schluss wird die Tanverwaltung getestet. Als Testeingaben werden dabei 0, eine Zahl über 100 und letztendlich eine Zahl zwischen 1 und 100 eingegeben.

5.2.4 Rolle Verwaltung

Begonnen wird mit der Benutzerverwaltung. Dabei werden die detaillierte Benutzeransicht und das Bearbeiten eines Benutzers getestet. Dabei wird vor allem das Ändern der E-Mail Adresse geprüft, da die anderen Daten schon bei den Rollen Eltern und Lehrer genau geprüft wurden. Dann wird überprüft, ob ein Benutzer gelöscht werden kann.

Ferner wird die Termineintragung einem Test unterzogen. Dazu wird versucht ein Termin unter Angabe von inkorrekten Daten anzulegen, um prüfen, wie die Software reagiert. Am Ende werden mehrere Testtermine mit korrekten Eingaben erstellt und überprüft, ob die Termine korrekt übernommen wurden.

5.3 Testprotokolle

Der abschließende Systemtest wurde genutzt, um eventuelle Fehler aus der Software zu entfernen und um die Benutzerfreundlichkeit zu erhöhen. Die Testprotokolle sind im Anhang unter 8.7 vorhanden.

6 Projektergebnis

6.1 Soll/Ist Vergleich

6.1.1 Zeiten

Geplante Stunden	Benötigte Stunden	Aufgabe
40	40	Erstellung des Pflichtenhefts
8	10	Erstellung eines Angebots
16	18	Benutzerhandbuch (mehrsprachig)
8	4	Erstellen einer Installationsanweisung
32	30	Erstellung einer Dokumentation
8	8	Datenbankmodellierung
16	26	Benutzerverwaltung
32	16	Terminverwaltung
16	26	Administrationsbereich
8	4	Datenimport
16	20	Rechteverwaltung
24	30	Erstellen von Mock Ups
24	24	Erstellen von Prototypen
16	18	Lehrerfiltererstellung
8	6	Formelementanpassung
32	30	Integration von Prototypen
48	46	Systemtest
352	356	Gesamt

6.1.2 Software

Bei der Erstellung der Software wurden alle Musskriterien umgesetzt. Einige Kannkriterien wurden aus Zeitgründen nicht erfüllt. Dazu gehört die Erstellung eines Installationsskripts. Außerdem wurde kein Zähler eingebaut, der die fehlerhaften Logins zählt und eventuell eine kurze Sperre hervorruft.

6.2 Verlauf

Während des Projekts wurde relativ schneller klar, dass das größte Problem das Zeitmanagement ist. Das Einarbeiten in das Projekt gestaltete sich langwieriger als gedacht. Dabei kostete die Einarbeitung in das Yii Framework sehr viel Zeit. Aber auch das Erstellen der Mock Ups, die bereits im Pflichtenheft vorhanden sein sollten, was sehr zeitaufwändig. Obwohl der Anfang sich als relativ schwierig erwies, konnte der kleine Rückstand durch einige zusätzliche Arbeitsstunden zu Hause wieder aufgeholt werden. Bei dem weiteren Verlauf konnte die Planung eingehalten werden, sodass die Software wie geplant am 20.03.13 fertiggestellt und mit dem Systemtest begonnen wurde.

6.3 Ergebnis und Fazit

Die Elternsprechtagssoftware konnte innerhalb der geplanten drei Wochen fertiggestellt werden. In der Livedemonstration wurde die Software präsentiert, wobei

alle im Pflichtenheft festgelegten Musskriterien funktionierten. Die Software ist also wie geplant funktionsfähig. Allerdings konnte innerhalb der drei Wochen noch kein Abnahmetest durchgeführt werden. Der Abnahmetest muss noch von der Brühlwiesenschule zu einem späteren Zeitpunkt ausgeführt werden.

6.4 Ausblick

In die Elternsprechtagssoftware sollen noch einige Features eingebaut werden, bevor diese zum ersten Mal verwendet wird. Dazu gehört eine Suche in allen Verwaltungstabellen, vor allem in der Benutzerverwaltung.

Möglich wäre die Erstellung eines Installationsskripts, um die Software automatisch auf einem Server zu installieren. Das Skript soll die Konfiguration übernehmen.

Bei dem Einlesen der Lehrerliste wird im Moment ein Standardpasswort vergeben. Das Passwort kann eventuell zufällig generiert werden und per E-Mail versendet werden.

Das Projektsrepository soll auf Git Hub veröffentlicht werden. Außerdem steht noch ein Abnahmetest aus, bevor die Software live benutzt werden kann.

7 Eidesstattliche Erklärung

Hiermit erklären wir an Eides statt, dass das Projekt selbständig von uns verfasst worden ist und keine Anderen als die vereinbarten Hilfsmittel benutzt wurden. Alle benutzten Hilfsmittel sind in Kapitel 4.2 aufgeführt.

Datum, Unterschrift (Christian Ehringfeld)

Datum, Unterschrift (David Mock)

Datum, Unterschrift (Matthias Unterbusch)

8 Anhänge

8.1 Lastenheft

8.2 Pflichtenheft

8.3 Angebot

8.4 Installationsanleitung

8.5 Benutzerhandbuch

8.6 User Manual

8.7 Testprotokolle

8.8 Teamprotokolle