

# Bachelorthesis

## Analyse & Planung der Neustrukturierung einer bestehenden Software zur Verwaltung von Schulen

**Vorgelegt am:** 31.07.2015

**Von:** Johannes Kauschke  
OT Buscha Nr. 25  
04618 Langenleuba-Niederhain

**Studiengang:** Wirtschaftsinformatik  
**Studienrichtung:** Wirtschaftsinformatik

**Seminargruppe:** 4WI10-1

**Matrikelnummer:** 4000710

**Praxispartner:** K&W Informatik GmbH  
Am Bahnhof 4  
08056 Zwickau

**Gutachter:** Herr Dipl.-Ing. (FH) Marco Gärtner (K&W Informatik GmbH)  
Herr Prof. Dr. Rainer Penzel (Staatliche Studienakademie Glauchau)

## Themenblatt Bachelorthesis

Studiengang Wirtschaftsinformatik

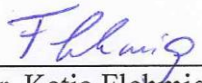
Student/in: **Johannes Kauschke**  
Matrikelnummer: **4000710**  
Seminargruppe: **4WI10-1**

### Thema der Bachelorthesis

**Analyse und Planung der Neustrukturierung einer bestehenden Software zur  
Verwaltung von Schulen**

Gutachter/ Betreuer: Herr Dipl.-Ing. (FH) Marco Gärtner  
Gutachter/ Betreuer: Herr Prof. Dr. Rainer Penzel

Ausgabe des Themas: **30.04.2015**  
Abgabe der Arbeit an den SG am: **31.07.2015, 14:00:00 Uhr**

  
\_\_\_\_\_  
Dr. Katja Flehmig  
Vorsitzende des Prüfungsausschusses  
Wirtschaft

**Berufsakademie Sachsen**  
Staatl. Studienakademie Glauchau  
Kopernikusstraße 51  
08371 Glauchau  
Tel. (0 37 63) 173 - 421  
Fax (0 37 63) 173 - 161



---

# Inhaltsverzeichnis

<b>Themenblatt</b> . . . . .	<b>II</b>
<b>Inhaltsverzeichnis</b> . . . . .	<b>III</b>
<b>Abbildungsverzeichnis</b> . . . . .	<b>VI</b>
<b>Tabellenverzeichnis</b> . . . . .	<b>VII</b>
<b>Abkürzungsverzeichnis</b> . . . . .	<b>VIII</b>
<b>1 Einleitung</b> . . . . .	<b>1</b>
1.1 Verbund der evangelischen Schulstiftung . . . . .	1
1.2 Ausgangssituation . . . . .	1
1.3 Zielsetzung . . . . .	2
<b>2 Analyse der bestehenden Software zur Verwaltung von Schulen</b> . . . . .	<b>3</b>
2.1 Systemübersicht . . . . .	3
2.2 Erfassung der vorhandenen technischen Struktur . . . . .	3
2.2.1 Serverstruktur . . . . .	3
2.2.2 Datenbankstruktur . . . . .	4
2.2.3 Quellcodestruktur/Backend (Programmtechnische Basis) . . . . .	5
2.2.4 Frontend (Präsentationstechnische Basis) . . . . .	6
2.3 Funktionen der Software . . . . .	7
2.3.1 Funktionsübersicht . . . . .	7
2.3.2 Allgemeine Funktionen . . . . .	7
2.3.3 Administration (Stammdatenverwaltung) . . . . .	8
2.3.4 Notenverwaltung . . . . .	15
2.3.5 Sitzpläne . . . . .	16
2.3.6 Stundenplan . . . . .	17

---

2.3.7	Klassen . . . . .	17
2.3.8	Elternansicht . . . . .	19
2.3.9	Lehrstoffplanung (Unterrichtsinhalte) . . . . .	20
2.3.10	Startseite . . . . .	21
2.4	Zusammenfassung . . . . .	22
<b>3</b>	<b>Betrachtung der Anforderungen an die zukünftige Software . . . . .</b>	<b>24</b>
3.1	Allgemeine Vorgaben . . . . .	24
3.2	Technische Anforderungen . . . . .	24
3.2.1	Skalierbarkeit und Schnittstellen. . . . .	24
3.2.2	Verfügbarkeit und Effizienz. . . . .	25
3.2.3	Sicherheit . . . . .	25
3.2.4	Wartung und Support. . . . .	25
3.2.5	RWD – Fähigkeit und Darstellung . . . . .	26
3.3	Funktionelle Vorgaben . . . . .	27
3.3.1	Login-System, Nutzerverwaltung . . . . .	27
3.3.2	Stammdatenverwaltung . . . . .	27
3.3.3	Zensuren (Notenverwaltung). . . . .	38
3.3.4	Fakturierung. . . . .	39
3.3.5	Stundenpläne und Zeitmanagement . . . . .	42
3.3.6	Sitzpläne. . . . .	42
3.3.7	Lehrstoffplanung . . . . .	43
3.4	Bewertung und Priorisierung der Anforderungen . . . . .	43
<b>4</b>	<b>Bewertung der vorhandenen Software anhand der Anforderungen . . . . .</b>	<b>45</b>
4.1	Erfüllung der allgemeinen Vorgaben . . . . .	45
4.2	Erfüllung der technischen Vorgaben . . . . .	45
4.3	Erfüllung der funktionellen Vorgaben . . . . .	46

---

4.4	Zusammenfassung der Anforderungen . . . . .	47
4.5	Umsetzungsmöglichkeiten . . . . .	47
4.5.1	Prüfung der Umsetzbarkeit. . . . .	47
4.5.2	Vorhandene Standardsoftware zur Verwaltung von Schulen. . . . .	48
4.5.3	Eigenentwicklung auf Basis der vorhandenen Software . . . . .	48
<b>5</b>	<b>Planung der Umsetzung. . . . .</b>	<b>49</b>
5.1	Systemübersicht. . . . .	49
5.2	Planung der technischen Struktur . . . . .	50
5.3	Planung der Systemarchitektur. . . . .	51
5.4	Entwicklungsumgebung. . . . .	52
5.5	Qualitätsmanagement – Systemtests . . . . .	53
5.6	Planung der Module . . . . .	54
5.6.1	Systemmodule . . . . .	54
5.6.2	Basismodule. . . . .	55
5.6.3	Anwendungsmodule . . . . .	55
5.7	Einführung und Parallelbetrieb . . . . .	56
<b>6</b>	<b>Zusammenfassung und Ausblick . . . . .</b>	<b>57</b>
6.1	Zusammenfassung . . . . .	57
6.2	Erreichung der Zielsetzung und Ausblick . . . . .	58
	<b>Quellenverzeichnis . . . . .</b>	<b>59</b>
	<b>Anhangverzeichnis . . . . .</b>	<b>62</b>

---

## Abbildungsverzeichnis

<b>Abbildung 1</b> - Systemübersicht Kreda . . . . .	3
<b>Abbildung 2</b> - Ausschnitt Schule der DB-Struktur Kreda . . . . .	5
<b>Abbildung 3</b> - UseCase-Diagramm Kreda . . . . .	7
<b>Abbildung 4</b> - MarktUp . . . . .	8
<b>Abbildung 5</b> - Schuldaten ändern und ergänzen . . . . .	10
<b>Abbildung 6</b> - Schuljahresdaten anlegen und bearbeiten. . . . .	11
<b>Abbildung 7</b> - Schülerdaten anlegen bzw. bearbeiten. . . . .	12
<b>Abbildung 8</b> - Kontakte anlegen bzw. bearbeiten . . . . .	13
<b>Abbildung 9</b> - Angestellte anlegen bzw. bearbeiten . . . . .	14
<b>Abbildung 10</b> - Benutzereinstellungen . . . . .	15
<b>Abbildung 11</b> - Notenübersicht . . . . .	16
<b>Abbildung 12</b> - Sitzplan erstellen . . . . .	17
<b>Abbildung 13</b> - Stundenplan . . . . .	17
<b>Abbildung 14</b> - Klassenübersicht . . . . .	18
<b>Abbildung 15</b> - Schüler-Fehltage . . . . .	19
<b>Abbildung 16</b> - Elternansicht . . . . .	20
<b>Abbildung 17</b> - Kreda Startseite . . . . .	22
<b>Abbildung 18</b> - Sammel-Fakturierung . . . . .	42
<b>Abbildung 19</b> - Systemübersicht des neuen Kreda . . . . .	49
<b>Abbildung 20</b> - Virtuelle Serverstruktur von Kreda . . . . .	51
<b>Abbildung 21</b> - Systemarchitektur . . . . .	52

## Tabellenverzeichnis

<b>Tabelle 1</b> - Person – Grunddaten . . . . .	29
<b>Tabelle 2</b> - Personengruppe – Interessenten. . . . .	30
<b>Tabelle 3</b> - Personengruppe - Schüler . . . . .	32
<b>Tabelle 4</b> - Personengruppe - Schüler – Förderschule . . . . .	33
<b>Tabelle 5</b> - Personengruppe - Schüler - Fotoerlaubnis . . . . .	34
<b>Tabelle 6</b> - Personengruppe – Sorgeberechtigte. . . . .	34
<b>Tabelle 7</b> - Personengruppe - Mitarbeiter. . . . .	35
<b>Tabelle 8</b> - Firmen . . . . .	35
<b>Tabelle 9</b> - Adressdaten . . . . .	36
<b>Tabelle 10</b> - Kontaktdaten . . . . .	36
<b>Tabelle 11</b> - Schuljahr . . . . .	37
<b>Tabelle 12</b> - Priorität der funktionalen Anforderungen . . . . .	44
<b>Tabelle 13</b> - Erfüllungsgrade der Anforderungen. . . . .	47

## Abkürzungsverzeichnis

ADHS	Aufmerksamkeitsdefizit-/Hyperaktivitätsstörung
ADS	Aufmerksamkeitsdefizitsyndrom oder –störung
AG	Arbeitsgruppe
AGB	Allgemeine Geschäftsbedingungen
BIC	Business Identifier Code (Geschäftskennzeichen)
BSI	Bundesamt für Sicherheit in der Informationstechnik
CPU	Central Processing Unit (Prozessor)
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
DB	Datenbank
EGE	Evangelische Schulgemeinschaft Erzgebirge
ERP	Enterprise-Resource-Planning
FIBU	Finanzbuchhaltung
GB	Gigabyte
HTML	HyperText Markup Language
HTTPS	HyperText Transfer Protocol Secure
IBAN	International Bank Account Number (Internationale Bankkontonummer)
ISO	Internationale Organisation für Normung
IT	Informationstechnologie
K&W	K&W Informatik GmbH
LRS	Lese- und Rechtschreibstörung (Legasthenie)
LTS	Long Term Support
MDK	Modul Development Kit
MOC	Modular Object Constructor



MS	Microsoft
NF	Normalform
OOP	Objektorientierte Programmierung
OP	Offene Posten
OS	Operating System (Betriebssystem)
OTP	One Time Password
PC	Personal Computer
PDF	Portable Document Format
PHP	Hypertext Preprocessor (ursprünglich Personal Home Page Tools)
PLZ	Postleitzahl
RAM	Random-Access Memory (Arbeitsspeicher)
RWD	Responsive Webdesign
SBA	Sächsische Bildungsagentur
SEPA	Single Euro Payments Area (Einheitlicher Euro-Zahlungsverkehrsraum)
TÜV	Technischer Überwachungsverein

# 1 Einleitung

## 1.1 Verbund der evangelischen Schulstiftung

Die Schulstiftung der Evangelisch-Lutherischen Landeskirche Sachsens berät und begleitet evangelische Schulen im Gebiet der Ev.- Luth. Landeskirche Sachsens. Aktuell sind 57 Evangelische Schulen bei 39 verschiedenen Schulträgern landeskirchlich anerkannt. Die Schulstiftung der Evangelisch-Lutherischen Landeskirche Sachsens bündelt die Interessen dieser Einzelschulen in rechtlichen und bildungspolitischen Fragen. Dadurch verleiht sie dem Evangelischen Schulwesen zusätzliches Gewicht und Bedeutung in der sächsischen Bildungslandschaft.

Darüber hinaus versteht sich die Schulstiftung als Beraterin und finanzielle Unterstützerin in allen Belangen, die die Führung und Entwicklung einer Schule betreffen. Das breite Spektrum erstreckt sich von konzeptionellen Themen, insbesondere zum evangelischen Profil, über die pädagogische Arbeit bis hin zur Betriebsführung. Im Mittelpunkt aller Bemühungen steht dabei das Leben und Lernen in christlicher Gemeinschaft.<sup>1</sup>

Weiterhin gehört zur Stiftungstätigkeit die Fort- und Weiterbildung der Mitarbeiter an Evangelischen Schulen und der Träger. Dazu veranstaltet die Schulstiftung eigene Fort- und Weiterbildungen, wobei den Themen entsprechend auch weitere kirchliche oder staatliche Bildungseinrichtungen mit einbezogen werden. Im Bereich der beratenden und begleitenden Tätigkeit der Schulstiftung stehen deren Mitarbeiter für Anfragen z.B. zum Schulalltag, Unterrichtsentwicklung oder Betriebsführung, persönlich oder über die Vermittlung von entsprechendem Fachpersonal bereit.<sup>2</sup>

## 1.2 Ausgangssituation

Im Rahmen der beratenden und begleitenden Tätigkeit der Schulstiftung im Bereich Betriebsführung der Schulen ist der Schulstiftung in den letzten Jahren aufgefallen, dass für die Verwaltung des Schulbetriebs in den verschiedenen Schulen eine Vielzahl von Programmen und System eingesetzt wird, um sich den individuellen Bedürfnissen der jeweiligen Schule zu stellen. Ungefähr die Hälfte der Schulen unter der Schulstiftung verwendet für die Verwaltung der Stammdaten (Schülerdaten, Lehrerdaten usw.) FuxSchool von FuxMedia. (FuxSchool ist eine Software für MS Windows als Einzelplatzanwendung oder als Mehrplatzlösung im Netzwerk für die tägliche Schulorganisation. Der Aufbau von FuxSchool besteht aus einzelnen, modularen Programmbausteinen mit einer zentralen Steuerungsleiste. Zu den

---

<sup>1</sup> vgl. online: Schulstiftung Profil, 2015 (27.07.15)

<sup>2</sup> vgl. online: Schulstiftung Stiftungstätigkeit, 2015 (27.07.15)

Modulen gehören: Verwaltung, Zeugnis –Schule, Zeugnis – Lehrer, Statistik, Etat, Inventar, Schulbuch, Dokumente, Oberstufe, Stunden- und Vertretungsplan.)<sup>3</sup> In anderen Bereichen z.B. für die Abrechnung von Schulgeld wird DATEV („Das DATEV-Buchhaltungssystem ist **der** Buchhaltungsstandard in Deutschland.“<sup>4</sup> Es sind etwa 40.000 der ca. 50.000 zugelassenen Steuerberater in Deutschland Mitglied der DATEV eG. Ebenso wird die Finanzbuchführung von rund 2,4 Millionen deutschen Unternehmen (meist mittelständisch) vom Steuerberater mit DATEV-Software erstellt.<sup>5</sup>), SFirm<sup>6</sup> und Excel-Listen für Steuerberatungsbüros (einzeln und in Kombination) zum Einsatz. Bei der Notenerfassung und dem Zensuredruck werden teilweise FuxSchool oder andere Desktop-Lösungen verwendet.

Momentan existiert kein Programm oder System für die evangelischen Schulen Sachsens, welches alle ihre Anforderungen in einem System erfüllen kann. Um dem etwas Abhilfe zu schaffen, entstand an der Evangelischen Schulgemeinschaft Erzgebirge (EGE) in Annaberg das Webprogramm Kreda<sup>7</sup> durch Initiative eines einzelnen Informatik-Lehrers. Als die Schulstiftung auf Kreda aufmerksam gemacht wurde, entstand der Gedanke, Kreda weiter auszubauen und auch den anderen evangelischen Schulen für die Schulverwaltung zur Verfügung zu stellen. Um Kreda dafür fit zu machen, wendete sich die Schulstiftung an meinen Praxispartner, die K&W Informatik GmbH in Zwickau. Daraus entstand das Thema meiner Bachelorthesis: „Analyse & Planung der Neustrukturierung einer bestehenden Software zur Verwaltung von Schulen“.

### 1.3 Zielsetzung

Mit dieser Bachelorthesis soll ein Plan zur Neustrukturierung für die bestehende Software (Kreda) zur Verwaltung von Schulen erarbeitet werden. Dazu soll zunächst die bestehende Software nach der technischen Struktur und nach dem Funktionsumfang analysiert werden. Anschließend sollen die technische Vorgaben und funktionellen Vorgaben für die Software erfasst werden, damit diese in weiteren Schulen einsetzbar ist und zukunftsfähig wird. Aus diesen Vorgaben lässt sich die bestehende Software bewerten; in welchem Grad diese Vorgaben bereits erfüllt werden. Im Anschluss sollen die Umsetzungsmöglichkeiten für die Neustrukturierung geprüft und die Umsetzung geplant werden.

---

<sup>3</sup> vgl. online: FuxSchool Produkte, 2015 (27.07.15)

<sup>4</sup> GOLDSTEIN, 2013, S. 8

<sup>5</sup> vgl. GOLDSTEIN, 2013, S.8

<sup>6</sup> Banking-Software der Sparkasse

<sup>7</sup> Kreda (polnisch) bedeutet Kreide

## 2 Analyse der bestehenden Software zur Verwaltung von Schulen

### 2.1 Systemübersicht

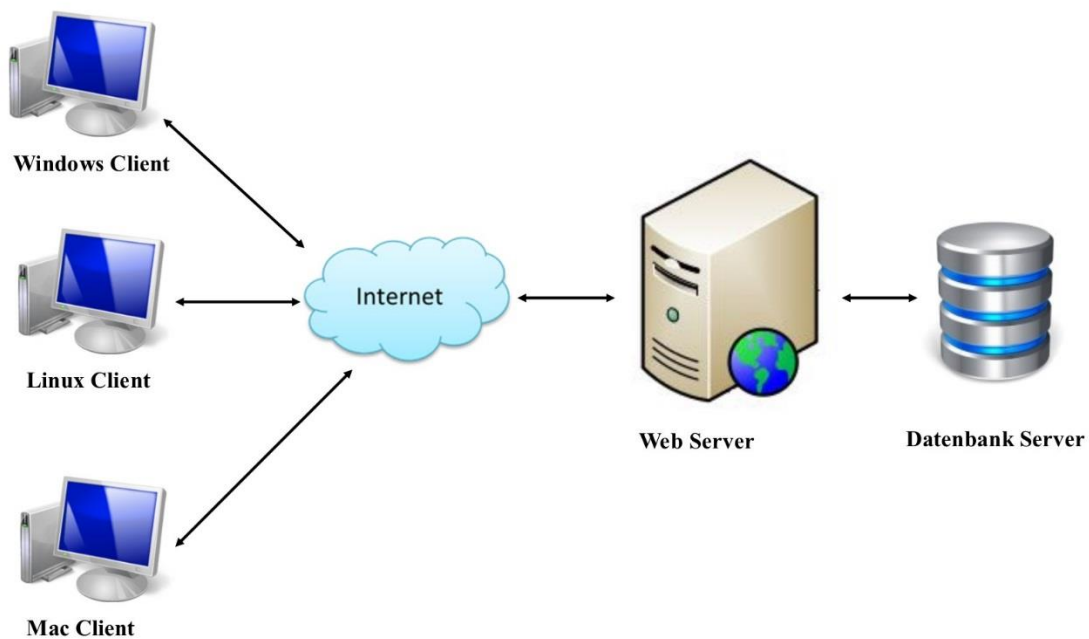


Abbildung 1 - Systemübersicht Kreda

Die bestehende Software zur Verwaltung von Schulen (Kreda) basiert entsprechend der Abbildung 1 auf einer Web-Client/Server-Architektur. Dabei greift der Nutzer (User) auf seinem Client (Windows Client, Linux Client, Mac Client, Tablet, Smartphone) via Browser über das Internet auf die Anwendung (Software) zu. Somit ist ein Zugriff auf die Anwendung von überall (Internet vorausgesetzt) möglich. Die eigentliche Anwendung läuft auf einem Web Server und die notwendigen Daten liegen auf einem Datenbank Server.

### 2.2 Erfassung der vorhandenen technischen Struktur

#### 2.2.1 Serverstruktur

Bei Kreda handelt es sich bei dem Web Server um einen einzelnen Apache HTTP Server 2.4.7<sup>8</sup> und bei den dem Datenbank Server um einen MySQL Community Server 5.5.34 (Laut DB-Engines ist MySQL eines der weltweit verbreitetsten relationale

<sup>8</sup> weitere Informationen online: Apache HTTP Server Project, 2015 (27.07.15)

Datenbankverwaltungssysteme. [Stand: Juli 2015 auf Platz 2 hinter Oracle)]<sup>9</sup>. Sie werden momentan beide bei Strato<sup>10</sup> gehostet und laufen beim Hoster auf Ubuntu (Linux) Servern.

### 2.2.2 Datenbankstruktur

Der Anhang 1 zeigt die Datenbankstruktur der bestehenden Software Kreda. Die DB-Tabellen befinden sich teilweise in keiner normalisierten Form. Als Normalisierung wird ein Prozess bezeichnet, mit dessen Hilfe man eine gute DB-Tabellenstruktur erreichen kann. Die Normalisierung kann die in der Datenbank vorhandenen Datenredundanzen verringern. Wobei die Normalisierung die Redundanz nicht eliminiert, sondern sie nur in kontrollierbare Bahnen lenkt. Für die erste Normalform (1NF) muss jedes Attribut der Relation einen atomaren Wertebereich und die Relation frei von Wiederholungsgruppen sein. Folglich sind keine zusammengesetzte, mengenwertige oder geschachtelte Wertebereiche zulässig.<sup>11</sup> Dies wird z.B. in der DB-Tabelle *schule* (Abbildung 2) nicht erfüllt, weil die Adresse der Schule nur in die Attribute *adresse* und *plz\_ort* aufgeteilt ist und nicht wie sinnvoller Weise in Straße, Hausnummer, Postleitzahl (PLZ) und Ort. Außerdem befindet sich in der DB-Tabelle *schule* jeweils ein Attribute *telefon* und *fax*. Somit sind pro Schule immer nur eine Telefonnummer und eine Faxnummer in der Datenbank speicherbar (keine dynamische Erweiterung möglich). Weiterhin lässt sich die n:m-Verknüpfungstabelle *schule\_schulart* (Abbildung 2), welche nur aus den beiden Fremdschlüsseln *schule* und *schulart* besteht, schlecht updaten, da sie keinen einzelnen Primärschlüssel (z.B. Id) besitzt.

Des Weiteren sind fast alle Tabellennamen auf Deutsch, außer die beiden DB-Tabellen *users* und *user\_pwd*, diese besitzen einen Englischen Tabellennamen. Dasselbe trifft auf die Spaltennamen innerhalb der DB-Tabellen zu, wobei in der Tabelle *user\_pwd* deutsche (z.B. *gilt\_seit* und *verbleibende\_versuche*) und englische (z.B. *user\_pwd\_hash* und *user*) Spaltennamen besitzen. Es empfiehlt sich, die DB-Tabellen-Bezeichnungen komplett in einer Sprache zu halten. Am besten in englischer Sprache, durch seine oft kürzeren und treffenderen Bezeichnungen im Vergleich zur deutschen Sprache.

---

<sup>9</sup> vgl. online: DB-Engines Ranking, 2015 (27.07.15)

<sup>10</sup> Web-Hoster

<sup>11</sup> vgl. GEISLER, 2014, S. 177 - 184

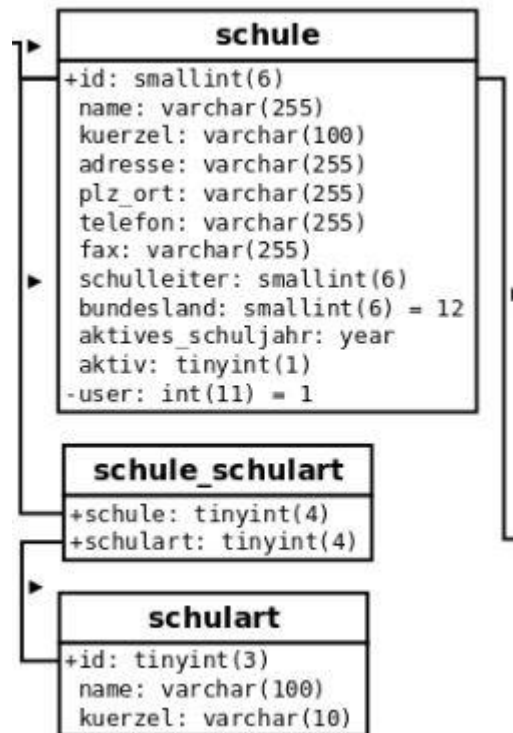


Abbildung 2 - Ausschnitt Schule der DB-Struktur Kreda

### 2.2.3 Quellcodestruktur/Backend (Programmtechnische Basis)

Kreda ist in der Skriptsprache PHP 5.3 (Hypertext Preprocessor) programmiert. Der Quellcode von Kreda ist seitenbasiert und prozedural.

Seitenbasiert bedeutet hier, dass für jede einzelne Web-Ansicht (innerhalb der Webanwendung) eine einzelne PHP-Datei vorliegt. Jede Web-Ansicht (Präsentation) interagiert unmittelbar mit der Datenquelle und auch der Programm-Logik zur Erstellung. Als Folge dessen ist die Präsentation nicht von den Daten und Logik entkoppelt, dementsprechend lässt sich der Quellcode an einer anderen Stelle nicht wiederverwenden und erschwert die Wartbarkeit der Webanwendung.

Bei der prozeduralen Programmierung erfolgt die Erstellung der Anwendung unter dem Ansatz, das Gesamtproblem in Teilprobleme zu untergliedern. Folglich schreitet die Anwendung von Teillösung (Prozedur) zu Teillösung zur eigentlichen Gesamtlösung.<sup>12</sup>

Weiterhin wurden teilweise ähnliche Funktionen durch das Kopieren von Quelltexten-Teilen zwischen den einzelnen PHP-Dateien umgesetzt. Wodurch bei Änderungen an diesen Teilen mehrere Stellen statt sinnvollerweise eine Stelle angepasst werden muss.

<sup>12</sup> vgl. NAHRSTEDT, 2009, S. 87

## **2.2.4 Frontend (Präsentationstechnische Basis)**

Die Präsentationstechnische Basis besteht bei Kreda aus HTML (HyperText Markup Language) und CSS (Cascading Style Sheets). Bei HTML handelt es sich nicht um eine Programmiersprache sondern um eine Seitenbeschreibungssprache, d.h. mit HTML wird nur beschrieben, wie die Webseite auszusehen hat, hingegen die Darstellung des exakten Layouts übernimmt der Webbrowser.<sup>13</sup> Mit CSS wird das Aussehen (Größe, Farbe, Schriftart usw.) der einzelnen HTML-Elemente beschrieben. Die Einbindung von CSS kann direkt im HTML-Header erfolgen oder per separate CSS-Datei. Im Falle von Kreda wird eine separate CSS-Datei für alle HTML-Seiten verwendet, wobei es keine separaten HTML-Dateien gibt, sondern der HTML-Quellcode direkt in den einzelnen PHP-Dateien integriert ist. Die Webseiten von Kreda sind für PC-Clients optimiert und eignen sich nur für Auflösung größer als 1024x768. Es erfolgt zwar eine automatische Anpassung auf die gesamte Breite des Bildschirms, allerdings verschwinden die Elemente der Webseite beim verkleinern des Browsers.

---

<sup>13</sup> vgl. WENZ; HAUSER, 2005, S.14

## 2.3 Funktionen der Software

### 2.3.1 Funktionsübersicht

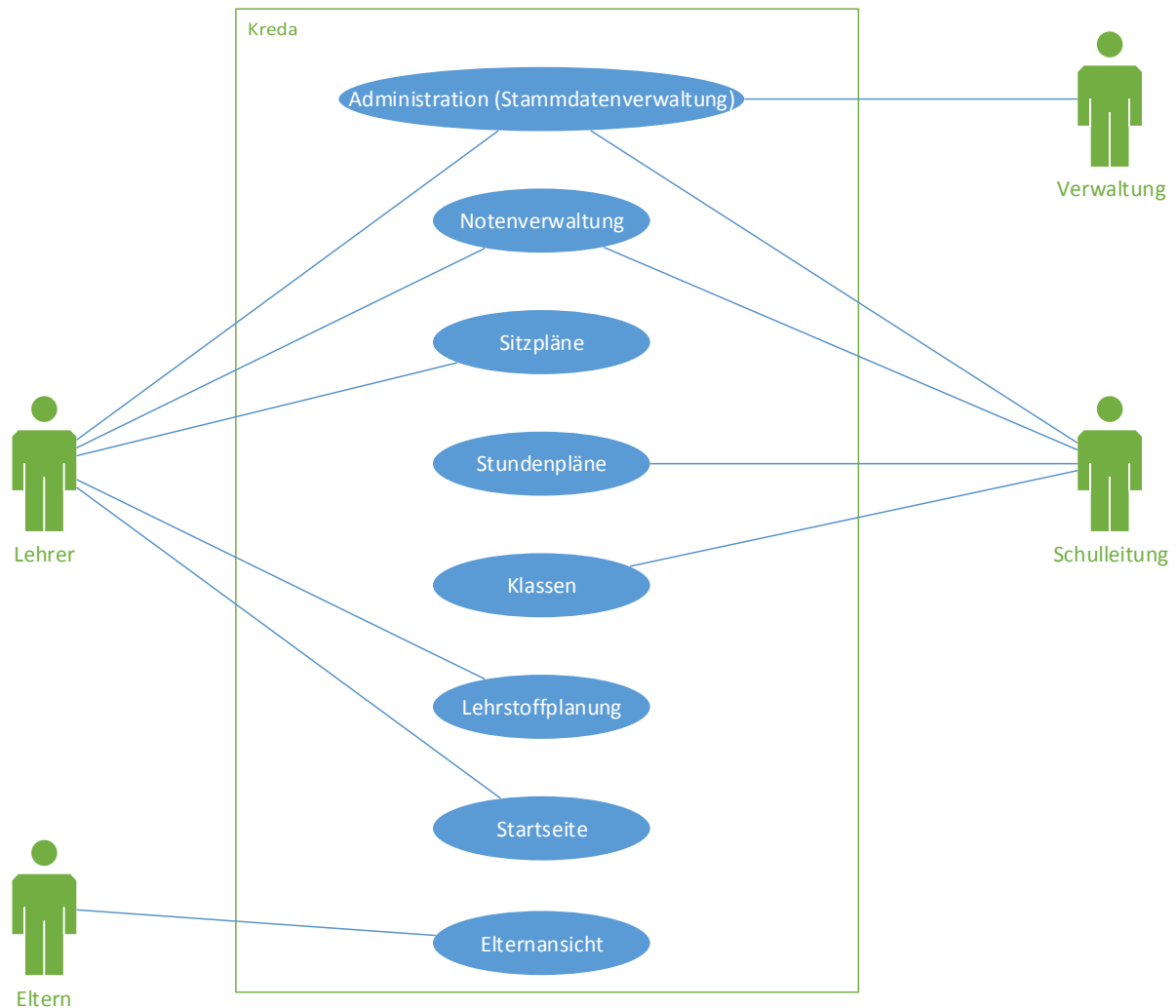


Abbildung 3 - UseCase-Diagramm Kreda

Die Abbildung 3 zeigt die einzelnen Funktionen von Kreda. Zu den Funktionen gehören die Administration (Stammdatenverwaltung), die Notenverwaltung, die Sitzpläne, die Stundenpläne, die Klassen, die Lehrstoffplanung, die Startseite und die Elternansicht. Diese Funktionen werden in den nächsten Abschnitten einzeln betrachtet.

### 2.3.2 Allgemeine Funktionen

Fast alle Textformulare (Bsp. siehe Abbildung 4) in Kreda enthalten einen Formatierungseditor (MarkItUp<sup>14</sup>). Mit diesem lassen sich die Texte formatieren, es lässt sich die Schriftart, die Schriftfarbe, eine Nummerierung usw. einstellen.

<sup>14</sup> Inhalts Editor Plugin auf Basis von jQuery



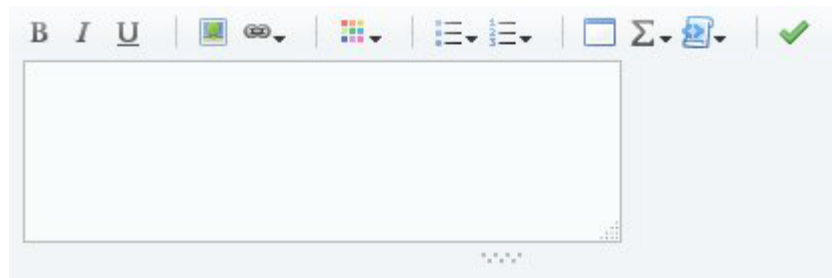


Abbildung 4 - MarkItUp

Zusätzlich zum normalen Login (Nutzername + Passwort) bei Kreda ist für die Authentifizierung der Lehrer, Verwaltungsangestellten und des Systemadministrators ein YubiKey erforderlich. Der YubiKey wird von dem amerikanisch-schwedischen Unternehmen yubico (bis 2011 Yubico) angeboten. Beim YubiKey handelt es sich um einen One Time Password (OTP) – Token, welcher mit der USB-Schnittstelle des Hostsystems kommuniziert, dazu ist keine zusätzliche Software auf dem Hostsystem erforderlich (funktioniert betriebssystemunabhängig). Der YubiKey erzeugt bei jedem auf Aufruf (durch Auflegen eines Fingers auf den Knopf) ein OTP, welches gegen ein YubiKey-Server validiert wird.<sup>15</sup>

### 2.3.3 Administration (Stammdatenverwaltung)

Die Stammdatenverwaltung von Kreda besteht aus den folgenden Punkten:

- Schuldaten
- Schuljahresdaten
- Kalender
- Schulfächer
- Schülerdaten
- Kontakte
- Angestellte
- Lehrauftragsverteilung
- Benutzereinstellungen

Zugriff auf die Stammdatenverwaltung haben das Sekretariat, die Schulleitung und der System-Administrator.

---

<sup>15</sup> vgl. SAMBLEBEN; SCHUMACHER, 2013, S.113

### **Schuldaten:**

In der Bearbeitungsmaske von einer Schule (Abbildung 5) können im oberen Bereich die Schul-Daten bearbeitet werden, wobei Pflichtfelder mit einem roten Stern markiert werden. Zu den Schuldaten gehören:

- Schulkürzel
- Schulname
- Schulart (mehrfach Auswahl ist möglich)
- Adresse
- Postleitzahl (PLZ) mit Ort
- Telefonnummer
- Telefaxnummer
- Schulleiter

Im mittleren Bereich der Abbildung 5 werden die bestehenden Räume der Schule dargestellt und es können dort neue Räume angelegt werden.

Der abschließende untere Bereich von Abbildung 5 zeigt die aktuellen Unterrichtszeiten der Schule, diese können hier bearbeitet werden. Wobei jeweils immer die Anfangszeit einer Stunde angegeben wird und die Unterrichtsstunde mit einer Länge von 45 Minuten festgelegt ist.

Die Räume und Unterrichtszeiten einer Schule sind wichtig für die Erstellung der Stundenpläne und Unterrichtsstunden der Lehrer.

← zurück

**Schul-Daten**

Schule anzeigen:

Kürzel\*:  Schulname\*:

Schulart(en)\*:  GY  OS  BGW  BVJ  BerGym  BFS  FO  FS  Fö  GS  HS  RS

Adresse:  PLZ Ort:

Telefon:  Fax:

Schulleiter:

**Räume**

! Hier können neue Räume eingetragen werden. Alte Räume bleiben aufgrund der Datenkonsistenz bestehen und können lediglich deaktiviert werden. Im jeweils zweiten Feld können außerdem Kommentare für einen Raum hinterlassen werden. Es reicht aus, wenn Sie nur die Räume eintragen, in denen Sie unterrichten.

Bestehende Räume:

- -1.02:  • -1.05:  • -1.06:  • -1.54:  • -1.56:  • -1.58:  • -1.59:  • 0.07:  • 0.08:  • 0.11:  • 0.12:
- 0.52:  • 0.53:  • 0.54:  • 0.55:  • 0.57:  • 0.58:  • 1.04:  • 1.05:  • 1.06:  • 1.07:  • 1.09:  • 1.10:
- 1.11:  • 1.51:  • 1.52:  • 1.53:  • 1.54:  • 1.55:  • 1.56:  • 1.57:  • 2.01:  • 2.02:  • 2.03:  • 2.04:
- 2.05:  • 2.06:  • 2.07:  • 2.08:  • 2.52:  • 2.53:  • 2.55:  • 2.56:  • 2.58:  • Halle 1:  • Halle 2:

Neu:

-

**Unterrichtszeiten**

! Um den Stundenplan nutzen zu können, müssen Sie hier den Beginn der Unterrichtszeiten der Schule im Format HH:MM eintragen. Eine Unterrichtsstunde ist prinzipiell 45 Minuten lang. Bei Doppelstunden tragen Sie eine anschließende zusätzliche Einzelstunde ein.

Bestehende Zeiten: 07:25 | 08:15 | 09:25 | 10:15 | 11:10 | 12:45 | 13:35 | 14:30 | 15:15 |

Neu:

- •  •  •  •  •  •  •
- •

Abbildung 5 - Schuldaten ändern und ergänzen

### Schuljahresdaten:

Die schulweiten unterrichtsfreien Tage sind für die Stoffverteilungspläne und Fehlzeitenerfassung wichtig. Zu diesen Tagen gehören z.B. gesetzliche Feiertage, Schulferien, Sportfest, Projektwoche usw. In der Abbildung 6 wird die Bearbeitungsmaske für die Schuljahresdaten dargestellt, in welcher die schulweiten unterrichtsfreien Tage einer Schule geändert werden können. Die Zuordnung der Schulferien erfolgt per Bundesland und der Halbjahreswechsel eines Schuljahrs kann angegeben werden. Durch die Festlegung des Halbjahreswechsels wird der Stichtag für die Zensuren des Halbjahres bestimmt. Die gesetzlichen Feiertage können aus einer festen Liste aller deutschen gesetzlichen Feiertage ausgewählt werden. Unter zusätzliche unterrichtsfreie Tage werden z.B. das Sportfest und die Projektwoche der Schule erfasst. Wird das Enddatum (Datum bis) nicht erfasst, gilt der unterrichtsfreie Tag nur für den angegebenen Tag.

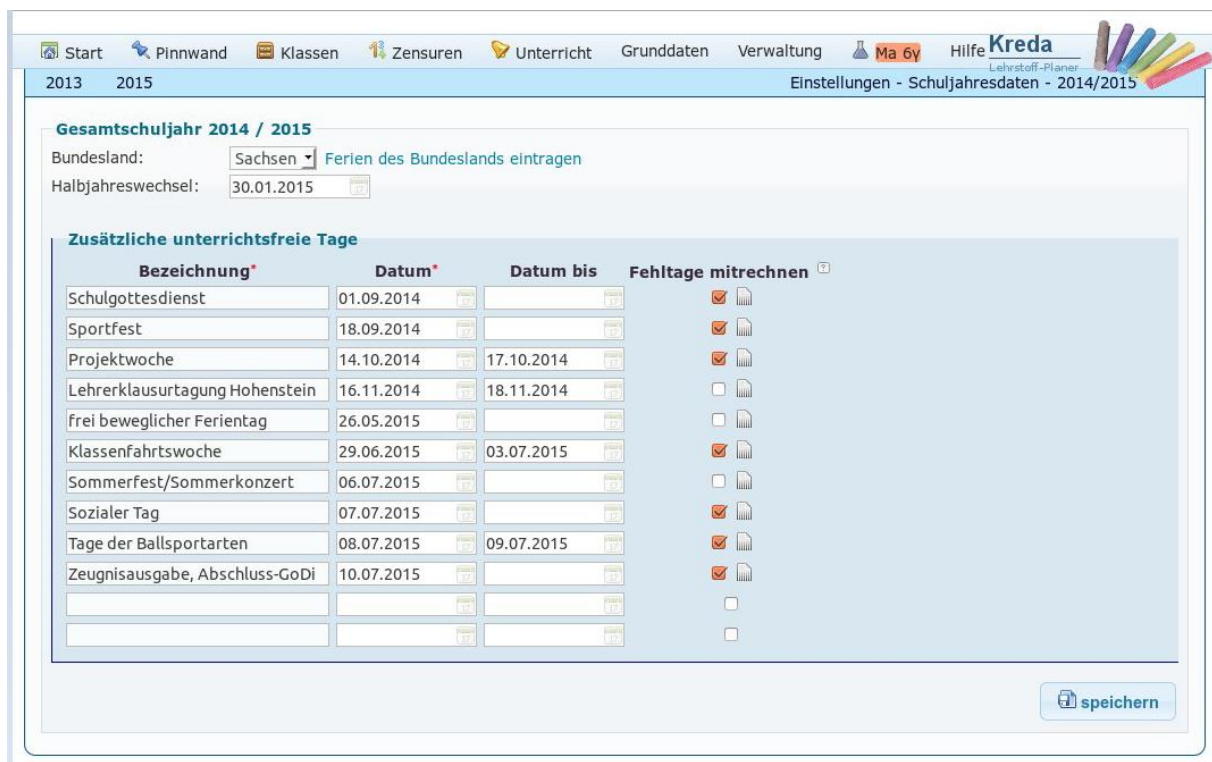


Abbildung 6 - Schuljahresdaten anlegen und bearbeiten

## Kalender:

Im Kalender werden Ferien und Feiertage (werden automatisch berechnet) dargestellt. Die Schulleitung kann einen A/B-Wochentausch vornehmen. Außerdem sind im Kalender Konferenzen und Dienstberatungen eingetragen. Die Erstellung und Bearbeitung der Konferenzen und Dienstberatungen erfolgt hier durch die Schulleitung). Diese Funktion sollte später in den Terminplaner eingepflegt werden.

## Schulfächer:

Unter Schulfächer werden von der Schulleitung zusätzlich zu den vordefinierten Standardfächern (z.B. Biologie, Deutsch, Musik, Mathematik) eigene Fächer erstellt. Für ein neues Fach werden die Schule ausgewählt, der Name und das Kürzel des Faches eingegeben. Es können nur eigene Schulfächer deaktiviert werden, nicht die vordefinierten.

## Schülerdaten:

Die Schülerdaten können aus FuxSchool mittels CSV-Datei importiert oder manuell eingegeben werden. Die folgende Abbildung 7 zeigt die Bearbeitungsmaske der Schülerdaten. Die Schülerdaten werden in die Bereiche: Schüler Grunddaten, Sorgeberechtigte und Schulintern gruppiert.

The screenshot displays a web-based form for entering or editing student data. It is divided into three main sections:

- Schüler Grunddaten:** Fields for Geschlecht (radio buttons for m, w), Name, Vorname, Geb.tag, -ort, Staat, Konf., Straße, PLZ, Ort, Telefon, E-Mail, and Bemerkungen.
- Sorgeberechtigte:** A table-like structure for adding guardians. Each entry includes Status, Sorgeb., Geschlecht, Titel, Name, Vorname, Straße, PLZ, Ort, E-Mail, and three phone numbers. There are also fields for KK (Krankenkasse) and im Notfall (Notfallsprechpartn).
- Schulintern:** Fields for Schülernummer, Zugang (Benutzername, Passw), Klasse, Stammg, abgebende Sch., Aufnahme- and aufnehmende S., Aufnahme- and aufnehmende S., Fotoerlaubnis, Schließfach, Religionsunt., Fremdspr. 1, 2, 3, Profil 1, 2, and Neigungskurs.

A 'speichern' button is located at the bottom left of the form.

Abbildung 7 - Schülerdaten anlegen bzw. bearbeiten

Zu den Schüler Grunddaten gehören:

- Geschlecht
- Name
- Vorname
- Geburtstag
- Geburtsort
- Staatsangehörigkeit
- Konfession
- Adresse (Straße, PLZ und Ort)
- Telefonnummer
- E-Mail-Adresse
- Bemerkungen

Im Bereich Sorgeberechtigte werden die folgenden Informationen zu den Sorgeberechtigten eines Schülers verwaltet:

- Status
- Geschlecht
- Titel
- Name
- Vorname
- Adresse (Straße, PLZ und Ort)
- E-Mail-Adresse
- Telefonnummern
- Krankenkasse (KK)
- Kontaktperson im Notfall
- Kommentar

Unter Schulintern befinden sich die folgenden schulspezifischen Informationen:

- Schülernummer
- Zugang (Benutzername, Passwort für Elternansicht)
- Klasse, Stammgruppe
- Bildungsgang
- abgebende Schule, Aufnahmedatum (Schüler ist von dieser Schule gekommen)
- aufnehmende Schule, Aufnahmedatum (Schüler hat die Schule verlassen und wurde von dieser Schule aufgenommen)
- Fotoerlaubnis
- Schließfach
- Religionsunterricht mit Angabe der Jahre
- Fremdsprache 1 – 3 mit Angabe der Jahre
- Profil 1 – 2 mit Angabe der Jahre
- Neigungskurs mit Angabe der Jahre

### Kontakte:

Die Abbildung 8 zeigt das Anlegen eines neuen Kontaktes (Person mit Kontaktdaten). Hierbei werden Personen-Daten (Titel, Geschlecht, Vorname, Name, Geburtsdatum, Personentyp), Adress-Daten (Straße mit Hausnummer, Wohnort) und Kontakt-Daten (Email-Adresse, Telefon 1, Telefon 2, Telefon 3) eingepflegt.

**neue Person**

Name	Adresse	Kontakt	Bemerkungen
Titel <input type="radio"/> m   <input type="radio"/> w Vorname <input type="text"/> Name <input type="text"/> Geburtsdatum <input type="text"/>	Straße HN <input type="text"/> PLZ <input type="text"/> Wohnort <input type="text"/> E-Mail <input type="text"/>	Telefon 1 <input type="text"/> Telefon 2 <input type="text"/> Telefon 3 <input type="text"/>	Bemerkungen <input type="text"/> Sorgeberechtigt <input type="checkbox"/>
<input type="button" value="hinzufügen"/>			

Name	Adresse	Kontakt	Bemerkungen	Aktionen
Bergsch, Hans	Waldstraße 123 12345 München	01234 5678	Sorgeberechtigt	
Bergsch, Maria		01234 5678-9101112 - 13141516   17181920	Lehrer	
Bergsch, Peter	Waldstraße 2 12345 München		Sorgeberechtigt	
Bergsch, Thomas	Kornstraße 3	01234 5678	Sorgeberechtigt	

Abbildung 8 - Kontakte anlegen bzw. bearbeiten

### Angestellte:

Im Rahmen der Angestellten (siehe Abbildung 9) stehen nur die folgenden Stammdaten zur Verfügung:

- Benutzername (Kürzel)
- Name, Vorname
- Emailadresse
- Rolle

neu / bearbeiten

Benutzername	Name, Vorname	E-Mail	Rolle	Passwort	
vba	Beck, Stefan	stefan.beck@gsa-erndorf.at	Lehrer	<input type="checkbox"/> Passwort neu setzen *****v	ändern

Kürzel	Name, Vorname	E-Mail	Rolle(n)	letzter Login	Aktionen
mad	Adler, Michael	michael.adler@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
vba	Beck, Stefan	stefan.beck@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄 <span>neues Passwort setzen</span>
rba	Bauer, Tobias	tobias.bauer@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
mbe	Beizer, Michael	michael.beizer@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
abe	Beizer, Andrea	andrea.beizer@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
tbe	Beizer, Tobias	tobias.beizer@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
kbi	Birkner, Emma	emma.birkner@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
mbr	Braun, Matthias	matthias.braun@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄
rbr	Brunner, Barbara	barbara.brunner@gsa-erndorf.at	Lehrer	Mo 28.12. 10:30	→ 🔑 📄

Abbildung 9 - Angestellte anlegen bzw. bearbeiten

### Lehrauftragsverteilung:

In der Lehrauftragsverteilung werden die Klassenlehrer und Fachlehrer per Lehrer-Kürzel den entsprechenden Fach-Klassen zugeordnet. Diese Zuordnung ist unter anderem dafür wichtig, dass nur Lehrer mit einem Lehrauftrag in einem Fach in einer Klasse Zensuren vergeben dürfen. Die Lehrauftragsverteilung kann aus einer Indivare-XML-Datei importiert werden. Indivare ist eine Software für die Stundenplanung und Abiturverwaltung.<sup>16</sup>

### Benutzereinstellungen:

Im Rahmen der Benutzereinstellungen (siehe Abbildung 10) kann der System-Administrator die Rechte für die jeweilige Person bearbeiten. Hierbei gibt es Optionen für Zensuren mit Punktabgaben, Kommentarmöglichkeit, Unterschriften/Berichtigungen, Dienstberatungen, Sitzordnungen, Schuljahresplanung, Statistiken (Hausaufgaben, Fehlzeiten usw.), Unterrichtsstunden-Planung und mögliche Druckansichten.

<sup>16</sup> weitere Informationen online: Indivare Homepage, 2015 (27.07.15)

Abbildung 10 - Benutzereinstellungen

### 2.3.4 Notenverwaltung

Die Abbildung 11 zeigt die Notenübersicht von Kreda. Bei den Noten (Zensuren) können die klassischen Noten von 1-6 sowie das Punktesystem (0 -15) der Sekundarstufe II erfasst werden. Zusätzlich zu den Noten können die erreichten Punkte von der Gesamtpunktzahl bei den Leistungen (Leistungskontrollen (LK), mündlichen Leistungen (MDL), Klausurleistungen (KL), Klassenarbeiten (KA) usw.) angegeben werden. Leistungen können dynamisch erweitert werden. Aus der Angabe von Punkten und Gesamtpunktzahl kann automatisch eine Zensur für die Leistung berechnet werden. Für eine Leistung wird die Durchschnitts-Note der Klasse berechnet und Noten von Schülern, welche stark von dieser abweichen, farblich markiert. Die Leistungen können auch, falls angegeben, Aufgabenweise ausgewertet werden. Unabhängig vom Datum können Noten dem Halbjahr (HJ) oder Gesamtjahr (GJ) zugeordnet werden. Weiterhin können Noten für die Berechnung der Halbjahresnote und Ganzjahresnote (Stichtagsnoten) nicht berücksichtigt werden. Diese Berechnungen erfolgen in Kreda automatisch, wobei den unterschiedlichen Leistungen eine Gewichtung der Zensuren mit einfließt. Die Gewichtung erfolgt anhand einer konfigurierbaren Bewertungstabelle. Es lässt sich z.B. für eine Klassenarbeit eine 50:50-Gewichtung zu den anderen Leistungen angeben. Die Vergabe der Noten erfolgt durch den Fachlehrer.

Zusätzlich zu den normalen Zensuren für Leistungen, kann die Schulleitung die Kopfnoten (Betragen, Fleiß, Mitarbeit, Ordnung) für die Zeugnisse der Schüler erfassen. Dafür gibt der Klassenlehrer des Schülers einen Vorschlag für die Kopfnoten



ab. Die Kopfnoten lassen sich dynamisch erweitern. Ein Zeugnis-Druck ist allerdings noch nicht im aktuellen Kreda integriert.

Klausuren 50:50 Rest Stand: 02.12.14	LK 25.09.	MDL 06.11.	LK 11.12.	KL 22.01.	HJ	LK 26.03.	HA 09.04.	LK 14.05.	KL 04.06.	GJ	↑
1. Klausur, Deutsch	11 <sup>33/41</sup>		13 <sup>45,5/49</sup>	11 <sup>29/36</sup>	<b>11,5</b>				9 <sup>27,5/37</sup>	<b>9,25</b>	
2. Klausur, Deutsch	9 <sup>30/41</sup>		9 <sup>35/49</sup>	10 <sup>28/36</sup>	<b>9,5</b>				6 <sup>22/37</sup>	<b>5,5</b>	
3. Klausur, Deutsch	8 <sup>27/41</sup>		7 <sup>31/49</sup>	9 <sup>25,5/36</sup>	<b>8,25</b>	2 <sup>13/38</sup>	15	7 <sup>16,5/27</sup>	3 <sup>14,5/37</sup>	<b>5,5</b>	
4. Klausur, Deutsch	11 <sup>33/41</sup>	15	12 <sup>43/49</sup>	11 <sup>30,5/36</sup>	<b>11,83</b>	4 <sup>17,5/38</sup>		10 <sup>19,5/26</sup>	6 <sup>21/37</sup>	<b>6,5</b>	
5. Klausur, Math	13 <sup>38/41</sup>		15 <sup>48/49</sup>	15 <sup>36/36</sup>	<b>14,5</b>	12 <sup>34/38</sup>		13 <sup>23,5/26</sup>	15 <sup>37/37</sup>	<b>13,75</b>	
6. Klausur, Deutsch	12 <sup>35/41</sup>		10 <sup>37,5/49</sup>	9 <sup>26,5/36</sup>	<b>10</b>	13 <sup>35,5/38</sup>	15	13 <sup>24/26</sup>	10 <sup>28/37</sup>	<b>11,83</b>	
7. Klausur, Deutsch	13 <sup>38/41</sup>		10 <sup>37/49</sup>	6 <sup>21,5/36</sup>	<b>8,75</b>	7 <sup>24/38</sup>		3 <sup>11,5/26</sup>	5 <sup>18,5/37</sup>	<b>5</b>	
8. Klausur, Math	13 <sup>37/41</sup>		11 <sup>41/49</sup>	14 <sup>34/36</sup>	<b>13</b>				15 <sup>0/</sup>	<b>15</b>	
9. Klausur, Biologie	15 <sup>40/41</sup>		13 <sup>45,5/49</sup>	13 <sup>33/36</sup>	<b>13,5</b>	15 <sup>37,5/38</sup>		13 <sup>24,5/27</sup>	15 <sup>36/37</sup>	<b>14,5</b>	
10. Klausur, Math	8 <sup>27/41</sup>				<b>8</b>					-	

Abbildung 11 - Notenübersicht

### 2.3.5 Sitzpläne

Im Sitzplan werden die Sitzordnungen (Tischverteilung im Raum) von bestimmten ausgewählten Lehrern erstellt. Diese Sitzordnungen können Lehrer benutzen, um Sitzpläne zu erstellen (siehe Abbildung 12). Es sind pro Lehrer mehrere Sitzpläne für eine Klasse möglich. Die Erstellung der Sitzpläne geschieht per „Ziehen und Loslassen“ (Drag-and-drop). Ist kein Foto eines Schülers verfügbar, wird neben dem Vornamen zusätzlich der Nachname des Schülers angezeigt. Der Klassenlehrer kann einen Sitzplan als Klassenlehrersitzplan definieren, dieser dient als Referenz für neue Sitzpläne einer Klasse.

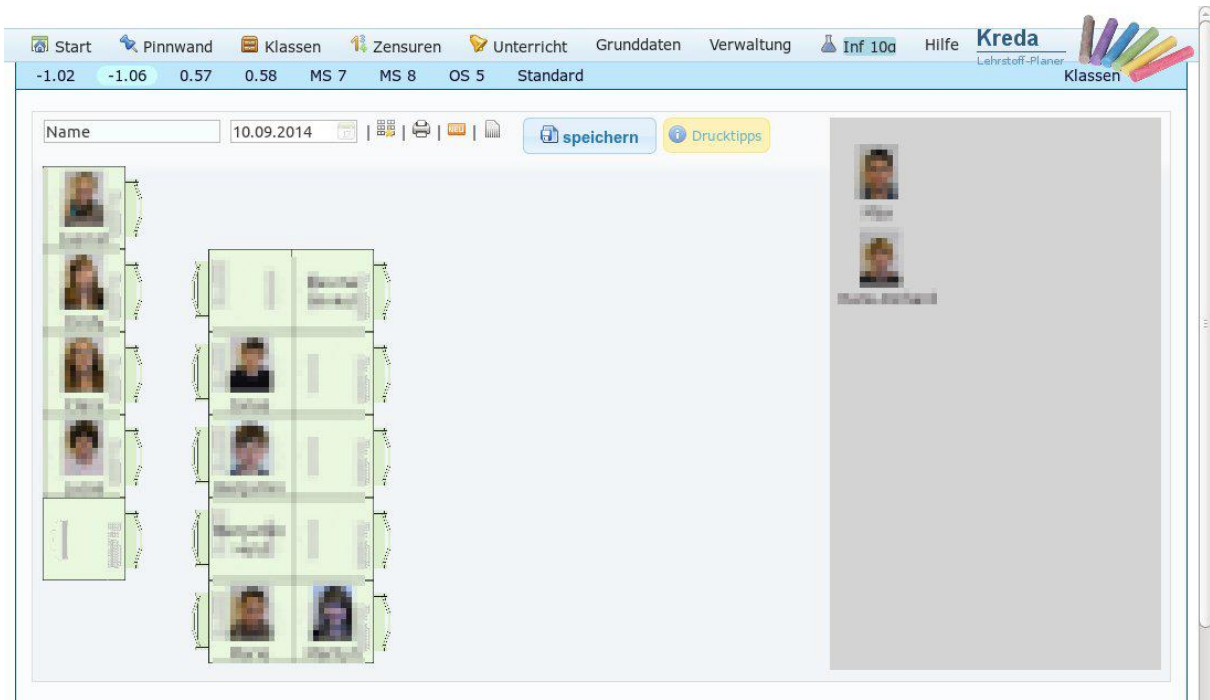


Abbildung 12 - Sitzplan erstellen

### 2.3.6 Stundenplan

Im Stundenplan gemäß Abbildung 13 können sich die Lehrer ihre Unterrichtsstunden per „Ziehen und Loslassen“ (Drag-and-drop) zusammenstellen. Weiterhin lassen sich hier Aufsichten eintragen und ändern. Die Veranschaulichung der A- bzw. B-Woche im Stundenplan erfolgt durch eine Kante (Bei einer A-Woche links, bei einer B-Woche rechts).

**Stundenplan - Schuljahr 14/15**

**Unterrichtsstunde hinzufügen**

Fach-Klasse:

Woche:  beide  
 A-Woche  
 B-Woche

Raum:

PC 5a A  
-1.02

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag	EGE (17 Ustd)
1. Std	Ma 6y 0.53(B)		Ma 7y 1.52(B)	Ma 6y 0.53	Ma 7y 1.52	07:25 08:10
2. Std	Ma 6y 0.53(B)		Ma 7y 1.52(B)	Ma 6y 0.53(A)	Ma 7y 1.52(A)	08:15 09:00
3. Std			Inf 10a -1.06(B)	1.06	Sportplatz Inf 10b -1.06(A)	09:25 10:10
4. Std			Inf 10a -1.06(B)		Inf 10b -1.06(A)	10:15 11:00

Abbildung 13 - Stundenplan

### 2.3.7 Klassen

Unter Klassen (siehe Abbildung 14) sieht jeder Fachlehrer seine Fach-Klassen-Kombinationen. Mit dem Häkchen vor einer Fach-Klasse wird eine aktive Fach-Klasse gekennzeichnet. Mit einem G nach der Fach-Klasse, wird diese als Gruppe markiert. Es handelt sich somit um eine Schülersauswahl aus mehreren Klassen (z.B. Sekundär-Stufe 2-Kurs). Die Schülerzuordnung zu einer Fach-Klasse ist per „Ziehen und

Loslassen“ (Drag-and-drop) aus verschiedenen Klassen möglich. Durch den Fach-Lehrer wird die Reihenfolge einer Fach-Klasse bestimmt oder die Reihenfolge wird automatisch berechnet. Das Löschen einer Fachklasse ist nur durch den System-Administrator möglich, wobei zuvor der zu löschende Inhalt nochmal als Vorschau angezeigt wird.

Klasse	Schüler	aktiv	Fach-Klasse-Kombinationen bearbeiten
5a	EGE 24	Ma, PC (G), PC (G)	
5b	EGE 27		
5d	EGE 28		
5y	EGE 28		
6a	EGE 24	TC (G), TC (G)	
6b	EGE 28	TC	
6d	EGE 29		
6y	EGE 29	Ma	Fach-Klassen
7a	EGE 28		
7b	EGE 28	Ma, OLe	
7d	EGE 26	Ma	
7y	EGE 27	Ma, Ma (G)	
8a	EGE 26		
8b	EGE 27	Inf, Ma	
8d	EGE 27		
8y	EGE 27		
9a	EGE 25	Inf	
9b	EGE 28	Inf	
9d	EGE 21		

Abbildung 14 - Klassenübersicht

Außerdem kann unter Klassen der Klassenlehrer eine Klassenbuchansicht der Klasse für das Klassenbuch ausdrucken. Weiterhin kann der Klassenlehrer in der Schülerübersicht der Klasse bzw. Fach-Klasse Schüler in eine andere Klasse verschieben, Schüler deaktivieren und Position und Bemerkung eines Schülers ändern. Andere Lehrer haben in der Schülerübersicht nur Leserechte.

In der Hausaufgabenstatistik von Schülern kann der Lehrer das Betragen und die Mitarbeit des Schülers ebenfalls mitnotieren.

Die Abbildung 15 zeigt die Fehltage von Schülern. Dazu werden die Fehltage automatisch berechnet (nur die tatsächlichen Fehltage werden einbezogen, keine Wochenenden, Ferien, unterrichtsfreie Tage mit Fehlzeiten, Feiertage usw.).

Weiterhin können unter Klassen Schülerlisten z.B. für das Bezahlen der Klassenfahrt angelegt werden.





Abbildung 16 - Elternansicht

### 2.3.9 Lehrstoffplanung (Unterrichtsinhalte)

Weiterhin bietet Kreda den Lehrern die Möglichkeit, Lehrstoffplanungen vorzubereiten und diese mit verschiedenen Themen zu taggen. Die Gliederung der Lehrstoffplanung erfolgt in Lehrpläne -> Unterrichtsabschnitte/Blöcke -> Abschnitts-Container -> Inhalte. Außerdem dient die Lehrstoffplanung der Vor- und Nachbereitung von

Unterrichtsstunden. Die Zuordnung der Blöcke zu dem Stoffverteilungsplan erfolgt per Drag-and-drop.

Im Übrigen enthält die Lehrstoffplanung folgende Funktionen:

- Aufgabensuche nach Themen
- Aufgabe bearbeiten
- Test erstellen und bearbeiten
- Test-Druckansicht und Lösungen
- Dateien (Arbeitsblätter, Folien, sonstige Dateien) anhängen
- Lehrbuchverwaltung
- Grafiken/Bilder hinzufügen (Thumbnails<sup>17</sup> werden automatisch erstellt)
- Material/Materialsammlung hinzufügen

### **2.3.10 Startseite**

Auf der Startseite von Kreda (Abbildung 17) für einen angemeldeten Lehrer werden die folgenden Informationen angezeigt:

- Nachbereitung
- Geburtstage
- Unterricht
- Tests
- Zensurenänderungen
- Notizfunktion

---

<sup>17</sup> Miniaturansicht

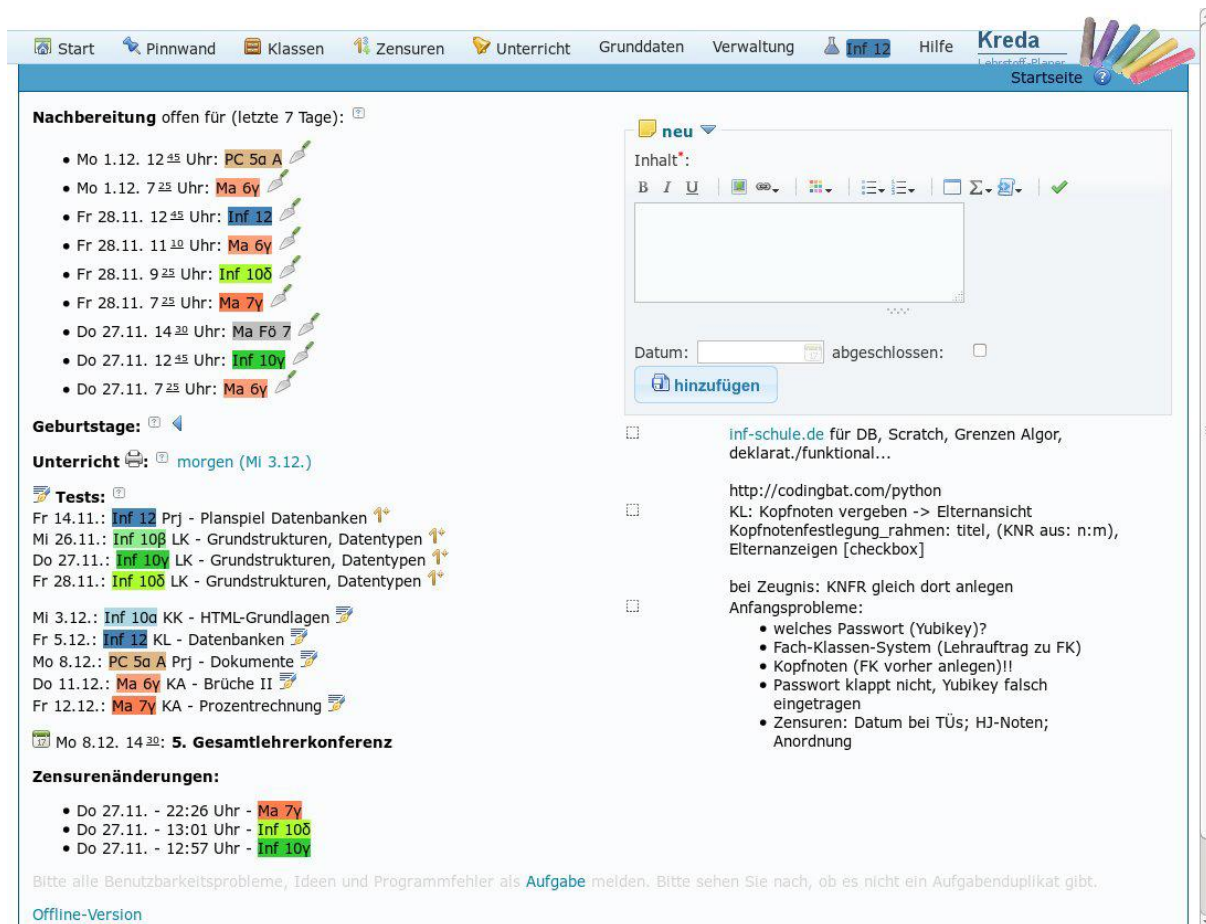


Abbildung 17 - Kreda Startseite

Oben links befindet sich die Nachbereitung. Diese zeigt dem Lehrer seine offenen Unterrichtsnachbereitungen der letzten 7 Tage. Damit erhält er einen schnellen Überblick seiner noch offenen Aufgaben. Unter der Nachbereitung werden die Geburtstage der nächsten Tage von eigenen Schülern und Lehrern angezeigt. Daraufhin folgt der nächste Unterricht des Lehrers.

Weiterhin werden dem angemeldeten Lehrer vergangene Tests und Leistungsüberprüfungen sowie vorzubereitende zukünftige Tests angezeigt. Abschließend auf der linken Seite werden die letzten drei Zensuren-Änderungen abgebildet.

Auf der rechten Seite der Startseite befindet sich eine Notizfunktion. Dort können auch direkt neue Notizen mittels eines Formatierungseditors angelegt werden. Die Notizfunktion sollte später in den Terminkalender als Aufgaben eingebettet werden.

## 2.4 Zusammenfassung

In diesem Kapitel wurde die bestehende Software (Kreda) zur Verwaltung von Schulen analysiert; beginnend mit einer Systemübersicht von Kreda, welche eine Web Client/Server Architektur aufweist, weiterführend mit der technischen Struktur, aus einem Web Server (Apache HTTP Server) und einem Datenbank Server (MySQL

Community Server) und der Datenbankstruktur. Der Quellcode (Backend) von Kreda ist seitenbasiert und prozedural. Für Präsentation wird bei der Webanwendung HTML und CSS eingesetzt. Abschließend wurden in diesem Kapitel die einzelnen Funktionen von Kreda betrachtet. Die Funktionen sind dabei gegliedert in allgemeine Funktionen (MarkItUp, YubiKey), Administration (Stammdatenverwaltung), Notenverwaltung, Sitzpläne, Stundenpläne, Klassen, Elternansicht, Lehrstoffplanung (Unterrichtsinhalte) und der Startseite von Kreda.



## **3 Betrachtung der Anforderungen an die zukünftige Software**

### **3.1 Allgemeine Vorgaben**

Im vorherigen Kapitel 2 wurde die bestehende Software zur Verwaltung von Schulen analysiert. In diesem Kapitel sollen nun die Anforderungen für die zukünftige Software betrachtet werden. Beginnend mit den allgemeinen Anforderungen, über die technischen Anforderungen, zu den funktionellen Anforderungen und abschließend mit der Bewertung und Priorisierung der Anforderungen.

Die Programmierung der zukünftigen Software muss nach den neuesten Standards erfolgen, um eine zukunftssichere und einfache Programmierung zu ermöglichen. Zudem soll die Software weiterhin als Webanwendung bestehen, damit der immer weiter vorantreibende Prozess der Nutzung von mobilen Endgeräten Betriebssystem unabhängig bleibt. Lehrer können z.B. auch von zuhause mit Kreda auf ihren eigenen Geräten arbeiten.

### **3.2 Technische Anforderungen**

#### **3.2.1 Skalierbarkeit und Schnittstellen**

Die zukünftige Software soll flexibel skalierbar sein um der steigenden Anzahl von Schulen, welche die Software einsetzen gerecht zu werden. Unter Skalierbarkeit versteht man die Fähigkeit, Ressourcen so zu erweitern, dass im Idealfall die Kapazitäten linear zunehmen. Das wesentliche Merkmal einer skalierbaren Anwendung ist der Umstand, dass bei einer Erhöhung der Auslastung lediglich zusätzliche Ressourcen und keine umfangreichen Anpassungen an der Anwendung notwendig sind.<sup>18</sup>

Aus dem Grund, weil die funktionellen Anforderungen an die zukünftige Software mit der Zeit wachsen, soll die Anwendung modular aufgebaut sein. Die Entwicklung neuer Module soll ohne Auswirkungen auf das Gesamtsystem erfolgen können (Erreichung durch festen Schnittstellen zwischen den einzelnen Modulen). Außerdem sollte die Möglichkeit bestehen, angepasste Modulversionen für verschiedene Schulen zu entwickeln. Dies wird erforderlich aufgrund unterschiedlicher Anforderungen einzelner Schulen.

---

<sup>18</sup> vgl. BEDNER, 2013, S.51

### **3.2.2 Verfügbarkeit und Effizienz**

Die Verfügbarkeit der Anwendung soll so sichergestellt werden, dass eine unerwartete Programmstörung nur auf Modul und Mandantenebene (jede Schule ist ein einzelner Mandant) auftreten, ohne das Gesamtsystem zu beeinträchtigen. Unter der Verfügbarkeit (availability) ist ein übergeordnetes IT-Schutzziel zu verstehen, welches den Schutz vor Informationsverlust, Informationsentzug oder Informationsblockade beinhaltet.<sup>19</sup> Die Anwendung soll jederzeit betriebsbereit sein, um den vorgesehenen Zugriff auf die notwendigen Daten zu gewährleisten. Der Zeitraum ohne Verfügbarkeit der Anwendung soll so gering wie möglich gehalten werden.

Ansonsten muss die Anwendung so gestaltet werden, dass diese effizient arbeitet. Unter Effizienz in der IT wird das Verhältnis zwischen der Leistung der IT-Funktion und dem dafür notwendigen Aufwand verstanden. Die Leistung wird dabei über Umfang und Qualität der Geschäftsprozessunterstützung bzw. –automatisierung in Form von Anwendung sowie der dazu notwendigen Hardware bestimmt. Hingegen der Aufwand entspricht den Gesamtkosten der IT-Funktion. Funktionale Redundanzen können z.B. zu erhöhten Wartungs- und Betriebskosten führen.<sup>20</sup> Zusätzlich soll ein Performanzgewinn durch Verteilung von Datenbankinhalten und Webanwendungen über mehrere Server (auch nachträglich) möglich sein.

### **3.2.3 Sicherheit**

Ein wichtiges Thema für die zukünftige Software ist das Thema Sicherheit, da sensible Daten wie z.B. Personendaten und Zensuren-Daten abgespeichert werden. Mit dem Begriff IT-Sicherheit werden Probleme und Gegenmaßnahmen umschrieben, welche sich bei der Nutzung moderner Informations- und Kommunikationssysteme durch die Einflussnahme Unbefugter ergeben. Diese Einflussnahme ist z.B. das Eindringen in Computersysteme oder die Manipulation von Daten. Es soll kein Unbefugter auf die Anwendung zugreifen, Daten lesen und Daten verändern können.<sup>21</sup> Auch der Zugriff auf die Daten einer Schule durch eine andere Schule muss verhindert werden, dies kann durch eine Kapselung der Datenbestände auf Basis von Mandant und Modul erreicht werden. Außerdem sollen die einzelnen Module untereinander nur über definierte Schnittstellen kommunizieren (Datenaustausch) können.

### **3.2.4 Wartung und Support**

Die Wartung von Software (software maintenance) beschäftigt sich mit Änderungen am System nach dessen Auslieferung. Bei den Änderungen handelt es sich z.B. um das Beheben von Fehlern (bugs), Verbesserung von Qualitätsattributen (wie

---

<sup>19</sup> vgl. BEDNER, 2013, S.184

<sup>20</sup> vgl. SCHMIDT, 2009, S.121

<sup>21</sup> vgl. POGUNTKE, 2007, S.3-4

Performanz und Evolution) und der Weiterentwicklung des Systems.<sup>22</sup> Gerade aus dem wichtigen Punkt - Weiterentwicklung des Systems - ergibt sich als Anforderung für die zukünftige Software, eine einfache und effiziente Wartung dieser zu gewährleisten. Durch die Modularisierung des Quellcodes soll eine effiziente Wartung ermöglicht werden. Zusätzlich sollte durch die Kapselung der Daten ein minimaler Anpassungsaufwand für die Weiterentwicklung der Software erreicht werden.

Neben der Wartung spielt für die Software auch der Support eine wichtige Rolle. Unter dem Begriff Support im Zusammenhang mit Software versteht man jede Form der Unterstützung, die Benutzer der Software bei auftretenden Problemen erhalten können. Diese Unterstützung kann durch die Mitarbeiter der eigenen Firma, einer Fremdfirma oder durch die Hersteller von Hardware und Software erfolgen.<sup>23</sup> Um den Support zu erleichtern, soll die Software benutzerfreundlich gestaltet werden. Zusätzlich soll eine Schulung zur Software für die neuen Benutzer erstellt werden, um diesen einen leichteren Einstieg zu ermöglichen und den Support zu entlasten.

### **3.2.5 RWD – Fähigkeit und Darstellung**

Für die Unterstützung der verschiedenen Endgeräte (Clients) soll die Software Responsive Web Design (RWD) enthalten. Responsive Web Design heißt übersetzt „reaktionsfähiges Webdesign“. Ohne RWD-Fähigkeit einer Webanwendung ergeben sich statische Websites, welche nicht in der Lage sind, auf die steigende Anzahl verschiedener Geräte und Display-Größen angemessen zu reagieren.<sup>24</sup> Durch RWD wird eine höhere Flexibilität erreicht, in dem für eine Website nur eine Oberfläche für die verschiedenen Display-Größen erstellt werden muss. Die Anpassung erfolgt automatisch an der verfügbaren Umgebung (PC, Tablet, Smartphone).

Neben der RWD-Fähigkeit der Software soll die Darstellung durch ein einheitliches Frontend in allen Modulen gestaltet werden. Dazu soll das Aussehen der Frontend-Elemente (z.B. Tabellen, Formulare, Buttons) innerhalb der gesamten Software gleich gestaltet werden. Ebenso soll die Software systemunterstützende Benutzereingaben beinhalten, um die Benutzerfreundlichkeit der Software zu erhöhen. Zu diesen systemunterstützenden Benutzereingaben soll eine Tabellenfilterung, Autovervollständigung und ein Kalender für Datumseingaben zählen.

---

<sup>22</sup> vgl. VOGEL; ARNOLD; CHUGHTAI, 2009, S. 189

<sup>23</sup> vgl. VILT, 2014, S.18

<sup>24</sup> vgl. ZILLGENS, 2012, S.7-8

### **3.3 Funktionelle Vorgaben**

#### **3.3.1 Login-System, Nutzerverwaltung**

Neben den technischen Anforderungen ergibt sich für die zukünftige Software auch eine Vielzahl von funktionellen Vorgaben. Im Bereich des Login-Systems soll auch weiterhin eine Hardware gestützte Authentifizierung (z.B. über den YubiKey<sup>25</sup>) die Sicherheit des Benutzerzugriffs gewährleisten.

Durch das Rechtesystem der Software soll sichergestellt werden, dass nur die entsprechenden berechtigten Nutzer Zugriff auf die einzelnen Module und Funktionen erhalten. Die Rechtestruktur der Software soll über ein Rollensystem erfolgen, in dem jeden Nutzer (Benutzer) eine Rolle zugewiesen wird, um seine Berechtigung auf die einzelnen Module und Funktionen zu ermöglichen. Das Zugriffslevel soll einmal auf Modul-Ebene erfolgen und darauf untergliedert innerhalb der einzelnen Funktionen und Services des Moduls.

#### **3.3.2 Stammdatenverwaltung**

Die Stammdatenverwaltung bezeichnet die zentrale Verwaltung von Stammdaten (Referenzdaten), um anwendungsübergreifende und systemübergreifende Konsistenz zu gewährleisten.<sup>26</sup>

Zu den Stammdaten in einer Schule gehören:

- Personendaten von Schülern, Mitarbeiter, Eltern, Spendern usw.
- Firmen (Unternehmen, Schulen, Vereine)
- Adressdaten, Kontaktdaten
- Klassen und Fächer
- Zeiten (Halbjahr, Schuljahr)
- Gebäude, Räumlichkeiten<sup>27</sup>
- Materialdaten<sup>28</sup>

Momentan ist FuxSchool das Stammdatenführende System bei über 20 von 55 Schulen der Schulstiftung der Evangelisch-Lutherischen Landeskirche Sachsens. Die neue Software soll FuxSchool in dieser Hinsicht ablösen und das führende System für die Stammdaten werden.

---

<sup>25</sup> beschrieben in 2.3.2 Allgemeine Funktionen

<sup>26</sup> vgl. HILDEBRAND, 2011, S.14

<sup>27</sup> zunächst in Kreda nicht erforderlich

<sup>28</sup> zunächst in Kreda nicht erforderlich

### **Personendaten:**

Die gemeinsamen Merkmale (Daten) der einzelnen Personentypen (Schüler, Mitarbeiter usw.) sollen durch die Grunddaten (siehe Tabelle 1 - Person – Grunddaten) zu einer jeden Person erfasst werden. Hingegen um die verschiedenen Merkmale (Meta-Daten) der einzelnen Personentypen (Schüler, Mitarbeiter usw.) in den Stammdaten erfassen zu können, soll diese über eine Gruppenverwaltung (Personengruppen) erfasst werden. Eine Person kann dabei zu mehreren Gruppen gehören, um die verschiedenen Merkmale der verschiedenen Gruppen zu erhalten. Für die Meta-Daten der einzelnen Gruppen soll es die folgenden vordefinierten (fest programmiert) Personengruppen geben:

- Interessenten
- Schüler
- Sorgeberechtigte
- Mitarbeiter (Pädagogisches Personal und Verwaltungspersonal)
- Sonstige

Zusätzlich zu den vordefinierten Gruppen soll es für die Schulen möglich sein, eigene freidefinierte Personengruppen (Pädagogisches Personal, Verwaltungspersonal, Geschäftspartner, Patenschaften, Vereinsmitglieder, Schulpostempfänger, Spender, Ehemalige Schüler, Lehrer, Erzieher, Technisches Personal usw.) anzulegen.

Name	Typ	Kommentar
Anrede	Auswahl	Herr, Frau, Schüler
Titel	Text	
Vorname	Text	
Zweitname	Text	
Nachname	Text	
Geschlecht	Auswahl	männlich, weiblich
Geburtstag	Datum	
Geburtsort	Text	
Staatsangehörigkeit	Auswahl	vordefiniert, auswählbar
Konfession	Text	
Mitarbeitsbereitschaft	Auswahl	ja, nein
Mitarbeitsbereitschaft – Tätigkeit	Text	
Bemerkungen	Text	für sonstige Informationen

**Tabelle 1 - Person – Grunddaten**

Neue Schüler, welche das Interesse haben, sich anzumelden, füllen momentan an den Schulen ein Aufnahme-Formular in Papierform aus. Manche Schulen geben diese Daten der Interessenten (siehe Tabelle 2 - Personengruppe – Interessenten) erst bei der Schüleraufnahme digital in ihr genutztes System (z.B. FuxSchool), andere Schulen geben die Grunddaten der interessierten Schüler und Eltern digital in einen Extra-Datenbestand von FuxSchool ein und übernehmen diese Daten bei der Aufnahme des Schülers. Im neuen System soll es die Möglichkeit geben, die Grunddaten von Interessenten zu erfassen und zu verwalten. Bei der Aufnahme eines Schülers sollen die Daten automatisch übernommen werden. Aus Datenschutzgründen muss eine Einwilligung für die Nutzung der Daten derjenigen vorliegen.

Name		Typ	Kommentar
Eingangsdatum		Datum	
Voranmeldung Klasse	für	Text	
Voranmeldung Schuljahr	für	Text	
Schnuppertag		Text	
Aufnahmegespräch		Text	

**Tabelle 2 - Personengruppe – Interessenten**

Die erforderlichen Daten für die Schüler sind in der Stammdatenverwaltung am umfangreichsten. Aus diesem Grund werden sie in die folgenden Bereiche aufgegliedert:

- Schüler (siehe Tabelle 3 - Personengruppe - Schüler)
- Förderschule (siehe Tabelle 4 - Personengruppe - Schüler – Förderschule)
- Fotoerlaubnis (siehe Tabelle 5 - Personengruppe - Schüler - Fotoerlaubnis)

Name	Typ	Kommentar
Schülernummer	Text	
Schulart	Auswahl	Grundschule, allgemein bildende Förderschule, Mittelschule, Gymnasium, Berufsschule, Berufsfachschule, Fachschule, Fachoberschule, Berufliche Gymnasium
Bildungsgang	Auswahl	Mittelschule mit Hauptschulabschluss, Mittelschule mit Realschulabschluss, Gymnasium mit Abitur
Ersteinschulung Datum	- Datum	
Ersteinschulung Schule	- Auswahl	Auswahl aus Firmen
Ersteinschulung Bemerkung	- Text	
Schülerbeförderung Einstiegshaltestelle	- Text	
Schülerbeförderung Ausstiegshaltestelle	- Text	
Schülerbeförderung Einstiegshaltestelle	- Text	
Schülerbeförderung Bemerkung	- Text	
Schuljahrwiederholung - Klassenstufe	- Text	
Schuljahrwiederholung - Wiederholungsart	- Auswahl	Freiwillig, Pflicht
Schuljahrwiederholung - Bemerkung	- Text	
Bisheriger Schulverlauf	Text	



Betrachtung der Anforderungen an die zukünftige Software

Aufnahme des Schülers - Datum	Datum	
Aufnahme des Schülers - Letzte Schulart	Auswahl	Auswahl siehe oben bei Schulart
Aufnahme des Schülers - abgebende Schule	Auswahl	Auswahl aus Firmen
Abgabe des Schülers - Datum		
Abgabe des Schülers - Zukünftige Schulart		Auswahl siehe oben bei Schulart
Abgabe des Schülers - aufnehmende Schule	Auswahl	Auswahl aus Firmen
Taufdatum	Datum	
Tauf-Ort	Text	
Krankenkasse	Text	
Krankenkassen - Versicherungsstatus	Auswahl	Freiwillig, Pflicht, Familien bei Mutter, Familien bei Vater, Privat
Eingabe des wievielten Kindes	Auswahl	1. Kind, 2. Kind, 3. Kind usw.
Krankheiten / Allergien	Text	
Medikamente	Text	
Behandelnder Arzt	Text	
Schließfach - Nummer	Text	
Schließfach - Schlüsselnummer	Text	
Hort	Auswahl	Auswahl aus Firmen
Hort - Wunsch	Auswahl	intern, extern
Hort - Öffnungszeiten	Text	

Tabelle 3 - Personengruppe - Schüler

Betrachtung der Anforderungen an die zukünftige Software

Name	Typ	Kommentar
Förderbedarf	Auswahl	ja, nein
Förderbedarfstest	Auswahl	ja, nein
Stundenbedarf pro Woche	Text	
Schulbegleitung	Auswahl	ja, nein
Teilstörungen	max. 3 mal Auswahl	LRS, Gehörschwierigkeiten, Augenleiden, Sprachfehler, Dyskalkulie, Autismus, ADS/ADHS, Rechenschwäche, Hochbegabung, Konzentrationsstörung, Körperliche Beeinträchtigung
Förderschwerpunkt	Auswahl	Sprache, Körperlich-motorische Entwicklung, Sozial-emotionale Entwicklung, Hören, Sehen, Geistige Entwicklung, Lernen
Datum für Antrag zur Beratung	Datum	
Datum für Antrag zur sonderpädagogischen Förderung	Datum	
Datum des Förderbescheides SBA	Datum	
Zuständige Förderschule	Auswahl	Auswahl aus Firmen
Bemerkung	Text	für sonstige Informationen

Tabelle 4 - Personengruppe - Schüler – Förderschule

Name	Typ	Kommentar
Foto	Datei	Bild des Schülers
Schuljahresveröffentlichungen	Auswahl	ja, nein (für Unterricht, Projekte und Schulveranstaltungen)
Schulschriften	Auswahl	ja, nein (wie Flyer und Newsletter)
Präsentationen zu Werbezwecken bei Partnern und anderen Schulen	Auswahl	ja, nein
Berichte in öffentlichen Zeitungen und Zeitschriften	Auswahl	ja, nein
Internetpräsenz der Schule	Auswahl	ja, nein
Facebook - Auftritt der Schule	Auswahl	ja, nein
Ton-, Video- und Filmaufnahmen	Auswahl	ja, nein

**Tabelle 5 - Personengruppe - Schüler - Fotoerlaubnis**

Weitere erforderliche Metadaten für die vordefinierten Personengruppen sind die Daten zu den Sorgeberechtigten (siehe Tabelle 6 - Personengruppe – Sorgeberechtigte) und die Daten für die Mitarbeiter (siehe Tabelle 7 - Personengruppe - Mitarbeiter). Momentan sind für die Sonstigen Personen keine eigenen Metadaten erforderlich, somit bestehen diese nur aus den Personen Grunddaten.

Name	Typ	Kommentar
Beruf	Text	
Arbeitsstelle	Text	

**Tabelle 6 - Personengruppe – Sorgeberechtigte**

Name	Typ	Kommentar
Angestellt für	Text	
Datum des Dienstvertrages	Datum	für Dokumentenerstellung - Änderungsvertrag/Vergütungsfestsetzung
befristet bis	Datum	
Grundvergütung	Text	
Mindeststundenzahl	Text	
Vergütungsgruppe	Text	
Vergütungsgruppe gültig bis	Datum	
Abminderungsstunden - Anzahl	Text	
Abminderungsstunden - Beschreibung	Text	

Tabelle 7 - Personengruppe - Mitarbeiter

### Firmen:

Neben den Personendaten sollen in Kreda auch Stammdaten zu Firmen (siehe Tabelle 8 - Firmen) erfasst werden. Unter Firmen werden Unternehmen, Verbände, Vereine, andere Schulen usw., welche Beziehungen mit der jeweiligen Schule pflegen, zusammengefasst. Es soll die Möglichkeit bestehen, den Firmen Personen mit ihrer jeweiligen Funktion in der Firma zuweisen zu können.

Name	Typ	Kommentar
Name	Text	

Tabelle 8 - Firmen

### Adressdaten und Kontaktdaten:

Zusätzlich zu den bereits erfassten Daten zu Personen und Firmen sollen Adressdaten (siehe Tabelle 9 - Adressdaten) und Kontaktdaten (siehe Tabelle 10 - Kontaktdaten) dynamisch zu den Personen und Firmen hinzufügar sein.

Name	Typ	Kommentar
Straße	Text	
Hausnummer	Text	
Postleitzahl	Text	
Ort	Text	
Ortsteil	Text	
Bundesland	Auswahl	Auswahl aus den deutschen Bundesländern.
Adresstyp	Auswahl	Hauptadresse, Rechnungsadresse, Lieferadresse, Standortadresse

Tabelle 9 - Adressdaten

Name	Typ	Kommentar
Typ	Auswahl	Telefonnummer, Emailadresse
Kategorie	Auswahl	Fax Geschäftlich, Fax Privat, Geschäftlich Festnetz, Geschäftlich Mobil, Notfall Festnetz, Notfall Mobil, Privat Festnetz, Privat Mobil für Telefonnummer  Privat, Geschäftlich für Emailadresse
Telefonnummer bzw. Emailadresse	Text	

Tabelle 10 - Kontaktdaten

### Zeiten:

Unter den Zeiten werden die einzelnen Schuljahre (siehe Tabelle 11 - Schuljahr) der Schulen definiert, die Schuljahre können je nach Bildungsgang abweichend sein.

Name	Typ	Kommentar
Name	Text	
Bildungsgang	Auswahl	
1. Halbjahr von	Datum	
1. Halbjahr bis	Datum	
2. Halbjahr von	Datum	
2. Halbjahr bis	Datum	

Tabelle 11 - Schuljahr

### Klassen und Fächer:

Neben der Möglichkeit Personen zu Personengruppen zusammen zu fassen, ist es für die Schüler erforderlich, diese zu Schulklassen zu gruppieren. Die Schulklassen bestehen aus Klassenstufe (z.B. 2. Klasse) und den Klassen (z.B. a, b). Außerdem werden die Schüler in Schulfächern (z.B. Mathematik, Deutsch) unterrichtet, welche von Schule zu Schule verschieden sein können. Ein konkretes Unterrichtsfach besteht dabei aus dem Zeitraum (Schuljahr), der Klassenstufe, der Klassengruppe (optional), Kategorie (Religionsunterricht, Fremdsprache, Profil, Neigungskurs, Vertiefungskurs, Fachrichtung, Wahlpflichtfach, Pflichtfach, Wahlfach und Arbeitsgruppe (AG)) und dem Schulfach. Alle Schüler eines Unterrichtsfaches sollen zu Unterrichtsgruppen zusammengefasst werden können. Außerdem soll es möglich sein, den Unterrichtsfächern Mitarbeiter (Lehrer) mit dem entsprechenden Lehrauftrag zu zuweisen.

### Import:

Für die Datenmigration (Datenübertragung) der Stammdaten aus dem von vielen Schulen verwendeten Alt-System FuxSchool soll es einen Daten-Import ins neue System geben. Dazu werden die Daten aus FuxSchool als CSV-Datei exportiert und anschließend ins neue System importiert. Für Schulen, welche z.B. Kreda oder andere Systeme nutzen, soll es einen allgemeinen CSV-Import ins neue System geben. Beim Datenimport ist auf Datenkonsistenz zu achten. Doppelte Einträge (Dubletten) sollen automatisch entfernt werden.

### Dokumente und Auswertung:

Die Dokumente und Auswertungen, welche aus den Stammdaten erstellt werden, sind sehr vielfältig. Diese variieren zwischen den einzelnen Schulen mitunter sehr stark.

Deshalb soll es im neuen System eine Möglichkeit für die Schulen geben, eigene spezifische Dokumente und Auswertungen zu generieren. Von Vorteil ist dabei ein visueller Designer für Dokumente und Auswertungen, damit die Schulen diese einfach und ohne großen Aufwand selbst erstellen können. Gegebenfalls ist zur Einführung des neuen Systems das Erstellen von Dokumenten und Auswertungen über eine ini-Datei (Text-Datei) ohne grafische Unterstützung möglich.

Beispiele für Dokumente:

- Finanzierungsanträge der SBA
- Serienbriefe an Eltern
- SEPA-Mandats Erteilung für SEPA-Lastschriften

Beispiele für Auswertungen:

- Auswertungen an die Schulstiftung
- Telefonlisten
- Klassenlisten

Aus Datenschutzgründen ist darauf zu achten, dass nur Personendaten von aktuellen Schülern (Angestellten usw.) zur Auswertung zur Verfügung stehen. Nach dem Verlassen der Schule von einer Person ist eine Auswertung ihrer Daten nicht mehr zulässig.

### **3.3.3 Zensuren (Notenverwaltung)**

Eine weitere Funktion der zukünftigen Software ist die Notenverwaltung. Bei den Noten sollen die klassischen Noten von 1-6 sowie das Punktesystem (0 - 15) der Sekundärstufe II erfasst werden können. Mit den Noten werden die Leistungen von Schülern bewertet. Die Leistungen (z.B. Leistungskontrollen, mündliche Leistungen, Klausurleistungen und Klassenarbeiten) sind an den verschiedenen Schulen unterschiedlich und sollen freidefinier- und erweiterbar sein. Neben den Noten für eine Leistung ist die Gesamtpunktzahl (mit den erreichten Punkten des jeweiligen Schülers) der Leistung sowie optional die Teilaufgabepunktzahl (mit den erreichten Punkten des jeweiligen Schülers) der Leistung angebar. Aus den erreichten Punkten und der Gesamtpunktzahl der Leistung wird automatisch ein Notenvorschlag für die Schüler erstellt. Dazu soll eine Notenverteilungstabelle (z.B. bis 96% Note 1, danach bis 81% Note 2 usw.) je Leistung ablegerbar sein. Über alle Schülernoten der Leistung soll eine Durchschnittsnote der Klasse gebildet werden und Noten von Schülern mit einer starken Abweichung von der Durchschnittsnote farblich markiert werden. Die Vergabe der Noten sollte nur durch Lehrer möglich sein, welche einen Lehrauftrag in dem entsprechenden Unterrichtsfach haben. Für die Berechnung der Stichtagsnoten (HJ und Gesamtjahr) sollen die Noten unabhängig vom Datum der Leistung dem HJ oder Gesamtjahr zuordenbar sein. Zur Berechnung der Stichtagsnoten werden die Noten

automatisch mit der freidefinierbaren Gewichtung der jeweiligen Leistung verrechnet. Der errechnete Wert der Stichtagsnote dient als Notenvorschlag und ist vom Fachlehrer anpassbar.

Neben den Noten für die Unterrichtsfächer vergibt die Schulleitung freidefinierbare Kopfnoten (z.B. Betragen, Fleiß, Mitarbeit, Ordnung) für die Zeugnisse der Schüler. Dafür darf der Klassenlehrer des jeweiligen Schülers einen Vorschlag für die Kopfnoten abgeben.

### **Zeugnisdruck:**

Die Layouts der Zeugnisse sind unter den Schulen der Schulstiftung verschieden und sollen über einen graphischen Designer (erforderliche Layout Elemente sind z.B. Bilder, freie Textfelder und Datenfelder) durch die Mitarbeiter der Schule erstellt werden; dabei entsteht eine Zeugnisvorlage (Layout). Es kann vorkommen, dass alte Zeugnisse erneut gedruckt werden müssen. Deshalb ist es erforderlich, die Zeugnisvorlagen mit ihrem Gültigkeitszeitraum (die Zeugnisvorlagen ändern sich über die Jahre, z.B. durch neue Vorgaben der Länder Bildungsministerien) digital aufzubewahren. Beim eigentlichen Druck der Zeugnisse für die Schüler werden die entsprechen Datenfelder im Layout gefüllt und das Zeugnis zusätzlich im PDF-Format abgespeichert. Der Druck der Zeugnisse soll durch Kreda auf unbedrucktem weisem Papier erfolgen. (Außer das Abschlusszeugnis, welches mit dem Wappen der Schule bereits vorgedruckt ist.).

### **3.3.4 Fakturierung**

Unter der Fakturierung (Rechnungslegung) wird der Vorgang verstanden, bei dem das Unternehmen (in diesem Fall die Schulen) Rechnungen an seine Kunden (in diesem Fall die Schüler mit ihren Eltern) generiert. Die so erstellten Faktura (Rechnungen) bilden die Grundlage für die Erfassung des Geschäftsvorfalles in der Buchhaltung (auf den entsprechenden Konten).<sup>29</sup>

Die Fakturierung in Kreda soll die folgenden Vorgänge des Schulträgers unterstützen:

- Fakturierung von Entgelten für Dienstleistungen, wie Beschulung (Schulgeld) und Betreuung (Hortgeld), diese sind in der Regel monatlich wiederkehrend während der gesamten Vertragslaufzeit.
- Fakturierung von Kosten für bereitgestellte Artikel bzw. Waren (Schulspeisung bzw. Essensgeld, Arbeitsmaterial, Kopiergeld, Schülerfotos usw.) mit einer Mengenangabe, inklusive das Verwalten von Warenkörben für Artikelsammlungen

---

<sup>29</sup> vgl. online: SIMON, 2015 (27.07.15)



- Fakturierung von anteiligen Projektkosten (Exkursionen, Klassenfahrten, Theatervorführungen usw.), dabei werden die gesamten Projektkosten anteilig auf die Teilnehmer umgelegt.

Die Herausforderung für die Fakturierung in Schulen im Vergleich zur Rechnungserstellung in einem klassischen Warenwirtschaftssystem (ERP (Enterprise-Resource-Planning) -System) ist die gleichzeitige Fakturierung von einer oder mehreren Leistung/en über viele Kunden. In einem Warenwirtschaftssystem werden eine oder mehrere Leistungen immer nur einem Kunden einzeln fakturiert.

Für die Fakturierung in Kreda sollen zunächst Debitorenkonten und falls erforderlich Finanzbuchhaltungskonten durch die Verwaltungsangestellten der Schule angelegt werden. Unter den Debitorenkonten werden der Debitor (kann der Schüler selbst oder ein Sorgeberechtigter des Schülers ein), die Debitorennummer, die Bankdaten des Debtors (Kontoinhaber, IBAN, BIC) und die Zahlungsart (vorgesehene Zahlungsarten sind SEPA-Lastschrift, SEPA-Überweisung, Bar) hinterlegt. Die Standard-Zahlungsart soll die SEPA-Lastschrift werden. Mit Verordnung zu SEPA (Single Euro Payment Area) hat das europäische Parlament und der Rat der europäischen Union einen integrierten Markt (Binnenmarkt) für die elektronische Zahlung in Euro, ohne die Unterscheidung zwischen einer Inlandszahlung und einer Zahlung ins Ausland (grenzüberschreitende Zahlung), geschaffen.<sup>30</sup> Im Rahmen der der SEPA-Lastschrift sind außerdem für ein Bankkonto eines Debtors SEPA-Mandatsreferenzen, welche der Debitor für jede Leistung erteilen muss, erforderlich. Für die Bankdaten des Debtors sollen in Kreda Mandatsreferenzen mit dem Signaturdatum hinterlegbar sein. Auf der anderen Seite lassen sich für die Finanzbuchhaltungskonten ein Kontenrahmen und Kontengruppen hinterlegen.

Anschließend können durch die Verwaltungsangestellten der Schule Leistungen (z.B. Schulgeld, Klassenfahrt) angelegt werden. Die Leistung kann als Einzel- oder Sammelleistung definiert werden. Bei einer Einzelleistung (z.B. Schulgeld) wird die Leistung einer jeden Person komplett berechnet, hingegen bei einer Sammelleistung (z.B. Klassenfahrt) wird die Leistung auf alle Personen gleichmäßig verteilt. Eine Leistung setzt sich wiederum aus einer oder mehreren Leistungsposition/en (Artikel) mit der jeweiligen Mengenangabe zusammen. Der Artikel besitzt einen Preis, eine Kostenstelle (optional), Bedingungen (optional) und ein Finanzbuchhaltungskonto (optional). Für die Bedingungen des Artikels in Kreda sind die Schulart und die Kind-Reihenfolge (welches Kind ist der Schüler) vorgesehen, dies ist erforderlich, weil z.B. bei vielen Schulen das Schulgeld abhängig von der Schulart und der Kind-Reihenfolge (weitere Kinder einer Familie zahlen weniger) ist. Ein Artikel mit Bedingungen soll für einen Schüler nur fakturiert werden, wenn diese Bedingungen erfüllt werden. Nachdem

---

<sup>30</sup> vgl. europäische Parlament; Rat der europäischen Union, 2012, S. 1

die Leistungen definiert wurden, können bei den Debitorenkonten zusätzlich noch Leistungen zugeordnet werden, welche für die entsprechende Leistung herangezogen werden sollen (z.B. zahlt die Mutter das Schulgeld für ihr Kind, und der Vater des Kindes zahlt das Hortgeld).

Die eigentliche Fakturierung für einen Vorgang erfolgt dann über die Zusammenstellung eines Warenkorbs (siehe Abbildung 18). Für den Warenkorb lassen sich einerseits Leistungen und andererseits Personen (Schüler) für eine Fakturierung auswählen. Die Leistungspositionen der ausgewählten Leistungen sollen nach Menge und Preis anpassbar sein. Durch die Auswahl eines Fälligkeitsdatums wird der Warenkorb abgeschlossen und für die Verwendung von wiederkehrenden Leistungen (z.B. bei Schulgeld) abgespeichert. Anschließend soll vom System automatisch geprüft werden, ob sich für jede Rechnung ein eindeutiger Debitor ermitteln lässt, ist dies nicht der Fall, kann der Nutzer den entsprechen Debitor auswählen. Nun sollen die einzelnen vom System erstellten Rechnungen anpassbar sein. Nach der Prüfung und Anpassung der Rechnungen durch den Verwaltungsangestellten kann dieser die Rechnungen freigeben. Durch die Freigabe einer Rechnung werden durch Kreda die Rechnungsdaten für den Export an DATEV und SFirm erzeugt (nur bei der Zahlungsart SEPA-Lastschrift) und in Kreda ein OP (Offener Posten) erstellt.

Für die Verwaltung der OP in Kreda sollen alle offenen Rechnungen (sind noch nicht vollständig bezahlt) in einer Übersicht angezeigt werden. Der Ausgleich der OP wird automatisch über einen Import des Zahlungseingangs von SFirm oder manuell über das Setzen der Rechnung auf Bezahlt erfolgen. Beim automatischen Abgleich müssen auch Teilzahlungen berücksichtigt werden.

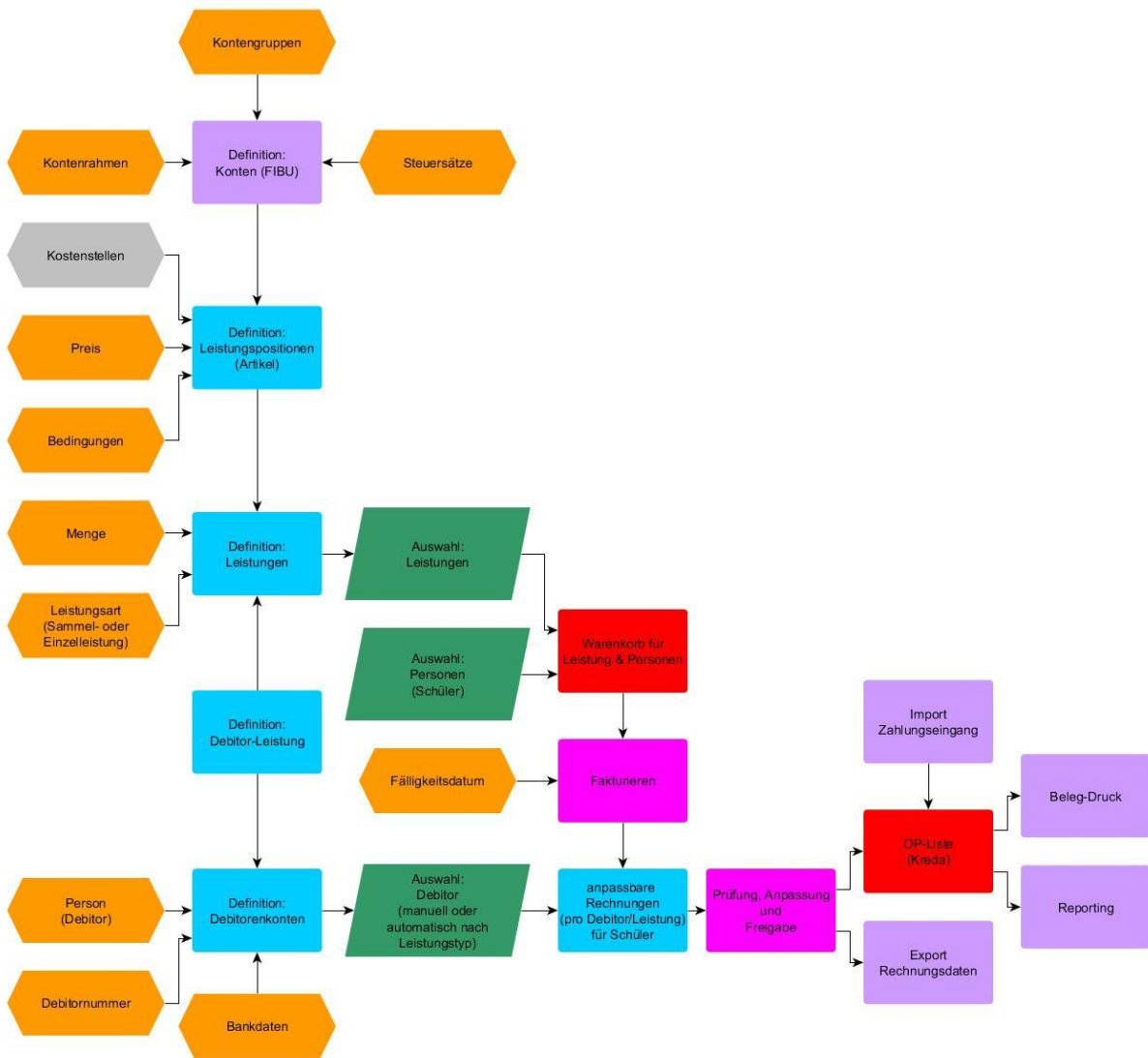


Abbildung 18 - Sammel-Fakturierung

### 3.3.5 Stundenpläne und Zeitmanagement

In Kreda ist die Erstellung von Stundenplänen möglich. Im Stundenplan werden den Unterrichtsstunden der Woche die Unterrichtsgruppen mit einem Raum zugewiesen. Die Unterrichtsstunden sind in ihrer Start- und Endzeit freidefinierbar. Weiterhin soll der Stundenplan eine A- und B-Woche berücksichtigen können, da der Stundenplan für manche Klassen nicht in jeder Woche gleich ist, sondern sich oft alternierend abwechselt. Zu den Stundenplänen gibt es auch eine Zeitplanplanung für Dienstberatungen, Dienstreisen, Veranstaltungen usw. Für die Zeitplanung sollen der Name, der Zeitraum (von, bis), der Ort, mögliche Teilnehmer und eine Beschreibung der Veranstaltung erfasst werden können. Auf mögliche Kollisionen von Veranstaltungen im Kalender soll hingewiesen werden.

### 3.3.6 Sitzpläne

Mit den Sitzplänen in Kreda werden Sitzordnungen (Tischverteilung im Raum mit Schülern einer Klasse) durch Lehrer realisierbar sein. Es sollen pro Lehrer mehrere

Sitzpläne für eine Klasse erstellbar sein. Anhand des Schülerfotos wird dieser im Sitzplan (neben seinem Namen) visuell dargestellt.

### **3.3.7 Lehrstoffplanung**

Zum momentanen Zeitpunkt liegen noch keine Anforderungen an die Lehrstoffplanung vor. Die Lehrstoffplanung soll entsprechend der Lehrstoffplanung des bestehenden Kreda realisiert werden. (siehe 2.3.9 Lehrstoffplanung (Unterrichtsinhalte))

## **3.4 Bewertung und Priorisierung der Anforderungen**

In diesem Kapitel wurden die Anforderungen an die zukünftige Software betrachtet. Die allgemeinen Vorgaben in der Form von Programmierung nach neuesten Standards und die Weiterentwicklung als Webanwendung bilden die Grundlage (Basis) für die zukünftige Software. Im Weiteren sind die technischen Vorgaben mit der Skalierbarkeit, den Schnittstellen, die Verfügbarkeit, die Effizienz, der Sicherheit, der Wartung, des Supports, der RWD-Fähigkeit und der Darstellung bei der Umsetzung der funktionalen Anforderungen zu berücksichtigen.

Bei den funktionalen Anforderungen bilden das Login-System/Nutzerverwaltung und die Stammdatenverwaltung die Grundlage. Somit besitzen diese die höchste Priorität (siehe Tabelle 12 - Priorität der funktionalen Anforderungen). Gefolgt von der Notenverwaltung als erstes eigenständiges Modul. Danach kommt von der Wichtigung gesehen die Fakturierung (vor allem im Zusammenhang mit Schulgeld). Weitere geplante Module für Stundenpläne und Zeitmanagement, Sitzpläne und Lehrstoffplanung besitzen die geringste Priorität und müssen zum Start von Kreda für weitere Schulen noch nicht verfügbar sein. Für die Verwendung von Stundenplänen kann zunächst die bereits erwähnte Software „Indiware“ genutzt werden, dafür ist lediglich ein Unterrichtsfächer-Export von Kreda an Indiware erforderlich.

Anforderung	Priorität
Login-System, Nutzerverwaltung	1
Stammdatenverwaltung	1
Notenverwaltung	2
Fakturierung	3
Stundenpläne und Zeitmanagement	4
Sitzpläne	4
Lehrstoffplanung	4

Tabelle 12 - Priorität der funktionalen Anforderungen

## **4 Bewertung der vorhandenen Software anhand der Anforderungen**

### **4.1 Erfüllung der allgemeinen Vorgaben**

Nachdem die im 2. Kapitel die bestehende Software zur Verwaltung analysiert wurde und im 3. Kapitel die Anforderungen an die zukünftige Software erfasst wurden, soll in diesem Kapitel Kreda mit den neuen Anforderungen verglichen werden. Dazu werden die allgemeinen, technischen und funktionellen Anforderungen betrachtet. Die allgemeine Anforderung, dass die Software eine Webanwendung sein soll, wird erfüllt. Hingegen kann das bisherige Kreda die Entwicklung nach neuesten Standards, aufgrund der vorhandenen Datenbankstruktur und Quellcode-Struktur, nicht erfüllen. Die Datenbankstruktur sollte mindestens nach der 3. Normalform gestaltet werden, um Redundanzen und Inkonsistenzen der Datenbank zu vermeiden. Ebenso eignet sich die Quellcode-Struktur mit seiner seitenbasierenden und prozeduralen Programmierung nicht für einen modernen Ansatz. Hierfür eignet sich eine objektorientierte Programmierung (OOP) besser. Die OOP wurde entwickelt, um den Software-Entwicklern ein eleganteres Modellieren ihrer Software auf Basis von realen Szenarien zu ermöglichen. Mit der OOP können Entwickler Dinge als Objekte mit grundlegenden Eigenschaften (properties) und ausführbaren Aktionen (methods) definieren.<sup>31</sup>

### **4.2 Erfüllung der technischen Vorgaben**

Im Bereich der technischen Vorgaben wird die geforderte Skalierbarkeit und Modularisierung vom bestehenden Kreda nicht unterstützt, da es einerseits keine Mandantenfähigkeit für die Nutzung bei mehreren Schulen unterstützt. Somit ist keine Aufspaltung von Ressourcen auf mehrere Web Server und Datenbank Server möglich. Aus diesem Grund ist auch keine Effizienz in Form eines Performanzgewinnes durch Verteilung auf mehrere unabhängige Ressourcen möglich. Andererseits besitzt Kreda keine abgetrennten Module, um die Entwicklung neuer Module ohne Auswirkungen auf das Gesamtsystem zu ermöglichen. Aus diesem Grund kann auch die Verfügbarkeit des Gesamtsystems, bei einer Programmstörung nur auf Modul- und Mandantenebene vom bestehenden Kreda nicht gewährleistet werden.

Ebenfalls kann die Sicherheit vom bestehenden Kreda nicht im vollen geforderten Umfang erfüllt werden, da momentan keine Möglichkeit für getrennte Datenbestände bei mehreren Schulen existiert. Allerdings eignet sich die Hardware gestützte

---

<sup>31</sup> vgl. HUDSON, 2006, S.133

Authentifikation mittels YubiKey des bestehenden Kredas gut, um das Eindringen ins System und die Manipulation durch unbefugte Personen zu verhindern.

Durch die Verwendung einer seitenbasierten und prozeduralen Programmierung beim bestehenden Kreda liegen teilweise bestimmte, zu einem Problem gehörende Anweisungen verstreut im Quelltext. Somit gestaltet sich die Wartung und der Support in den Bereichen Fehlersuche, Erweiterungen und Softwarepflege unübersichtlich. Weiterhin ergeben sich, wie bereits erwähnt, ähnliche Funktionen im bestehenden Kreda durch das Kopieren von Quelltexten-Teilen zwischen den einzelnen PHP-Dateien. Wodurch bei Änderungen an diesen Teilen mehrere Stellen statt sinnvollerweise eine Stelle angepasst werden muss. Falls die Anpassung nicht an allen notwendigen Stellen erfolgt, kann dies mitunter bei einer Fehlerbehebung zu einer weiteren Fehlerquelle führen.

Das bestehende Kreda besitzt keine RWD-Fähigkeit. Dadurch können verschiedene Endgeräte nicht durch die Verwendung einer Ansicht unterstützt werden. Die geforderte einheitliche Darstellung im gesamten System ist durch das bestehende Kreda gegeben. Allerdings existieren keine systemunterstützenden Benutzereingaben in Form von Tabellenfilterung und Autovervollständigung, sondern nur ein Kalender bei der Datumseingabe.

### **4.3 Erfüllung der funktionellen Vorgaben**

Nach den technischen Vorgaben werden nun die funktionellen Vorgaben im Vergleich zum bestehenden Kreda betrachtet. Beginnend mit dem Login-System. Das Login-System unterstützt durch die Verwendung des YubiKeys die geforderte Hardware-Gestützte-Authentifizierung. Allerdings eignet sich die momentane Nutzerverwaltung nicht, um ein Rollensystem abzubilden, weil im bestehenden Kreda die Rechte direkt einzeln am Nutzer vergeben werden.

Der Bereich der Stammdaten wird durch das bestehende Kreda momentan nur in einem sehr geringen Maß abgedeckt. Es fehlen z.B. im Bereich der Personen viele Merkmale der erforderlichen Personengruppen (es werden nur wenige Merkmale im Vergleich z.B. zu FuxSchool gepflegt). Im momentanen Kreda sind nur die Personen in Form von Schüler und Angestellten vorhanden, ein Gruppensystem für die Personen existiert nicht und Stammdaten für Firmen sind nicht enthalten. Weiterhin sind die Adressdaten und Kontaktdaten durch die feste Vorgabe der Anzahl pro Personentyp nicht dynamisch erweiterbar. Dokumente und Auswertungen sind im momentanen Kreda nicht vorhanden.

Im Rahmen der Notenverwaltung erfüllt das bestehende Kreda die Anforderungen zur Erfassung und Berechnung der Noten. Allerdings fehlt der geforderte Zeugnisdruck in Kreda noch komplett. Die auch geforderte Fakturierung fehlt ebenfalls noch komplett.

Die kleineren Bereiche wie Stundenpläne, Sitzpläne und Lehrstoffplanung werden durch das bestehende Kreda im geforderten Umfang abgedeckt; allerdings ist eine Zeitplanung für Dienstberatungen, Dienstreisen und Veranstaltungen noch nicht vorhanden.

#### 4.4 Zusammenfassung der Anforderungen

Die folgende Tabelle 13 - Erfüllungsgrade der Anforderungen - fasst die Erfüllung der einzelnen Vorgaben an das bestehende Kreda zusammen.

Anforderung	Erfüllungsgrade
Allgemeine Vorgaben	teilweise erfüllt
Skalierbarkeit	nicht erfüllt
Modularisierung	nicht erfüllt
Verfügbarkeit	nicht erfüllt
Effizienz	nicht erfüllt
Sicherheit	teilweise erfüllt
Wartung, Support	nicht erfüllt
RWD-Fähigkeit	nicht erfüllt
Darstellung	teilweise erfüllt
Login-System, Nutzerverwaltung	teilweise erfüllt
Stammdatenverwaltung	nur in einem sehr geringen Maße erfüllt
Notenverwaltung	erfüllt
Zeugnisdruck	nicht erfüllt
Fakturierung	nicht vorhanden

Tabelle 13 - Erfüllungsgrade der Anforderungen

#### 4.5 Umsetzungsmöglichkeiten

##### 4.5.1 Prüfung der Umsetzbarkeit

Nach dem die Erfüllung der Anforderungen am bestehenden Kreda geprüft wurden, wird nun untersucht, in welcher Form die bestehende Software zur Verwaltung von Schulen neustrukturiert werden kann. Aufgrund der veralteten Systembasis, welche



keine Trennung zwischen Benutzeroberfläche, Anwendungslogik und Datenbanknutzung hat, ist eine Neustrukturierung auf Basis des bestehenden Kreda nur sehr schwer möglich und nicht rentabel. Gerade die wichtigen technischen Anforderungen werden meist nicht erfüllt. Eine Modularisierung ist nicht möglich und die Modernisierung wäre extrem zeitaufwendig. Allerdings können Funktionen von Kreda in Form von Programm-Logik (Notenverwaltung, Stundenplan und Lehrstoffplanung) übernommen werden.

#### **4.5.2 Vorhandene Standardsoftware zur Verwaltung von Schulen**

Als eine Alternative zu Kreda kann die Verwendung einer Standardsoftware zur Verwaltung von Schulen in Betracht gezogen werden. Im Bereich der Windowsprogramme gibt es z.B. das bereits teilweise verwendete FuxSchool oder die Softwarelösung Atlantis<sup>32</sup>. Diese scheiden allerdings als Ersatz für Kreda aus, da sie keine geforderte Webanwendung sind (Atlantis teilweise über das erhältliche abgespeckte Web-Modul) und sie z.B. kein Modul zur Fakturierung besitzen. Im Webbereich ist als Standardsoftware nur die webbasierende Software All4Teachers<sup>33</sup> am Markt verfügbar. All4Teachers unterstützt zwar die meisten geforderten Anforderungen, allerdings fehlt ein Modul für die Fakturierung komplett und die Stammdaten von All4Teachers sind nicht im benötigten Umfang ausgebaut. Ein Vorteil von All4Teachers ergibt sich durch die sofortige Einsatzbarkeit für die Schulen.

#### **4.5.3 Eigenentwicklung auf Basis der vorhandenen Software**

Eine weitere Alternative zu den bereits erwähnten Realisierungsvarianten ist die Eigenentwicklung einer Webanwendung auf der Basis der vorhandenen Software. Bei dieser Variante wird eine neue Systembasis entwickelt und es wird teilweise die Programm-Logik (Notenverwaltung, Stundenplan und Lehrstoffplanung) von Kreda übernommen. Im Vergleich zu einer kompletten Neuentwicklung ist somit eine schnellere und kostengünstigere Umsetzung möglich. Von der Schulstiftung wurde die Variante der Eigenentwicklung auf Basis der vorhandenen Software für die Neustrukturierung von Kreda gewählt. Die Planung zu dieser wird im Weiteren betrachtet.

---

<sup>32</sup> weitere Informationen online: Atlantis Schulverwaltung, 2015 (27.07.15)

<sup>33</sup> weitere Informationen online: All4Teachers, 2015 (27.07.15)

## 5 Planung der Umsetzung

### 5.1 Systemübersicht

Nach der Entscheidung im vorangegangenen Kapitel 4 - die Neustrukturierung von Kreda in Form einer Eigenentwicklung auf Basis der vorhandenen Software durchzuführen - wird jenes in diesem Kapitel geplant. Dabei werden die technische Struktur, die Systemarchitektur, das Qualitätsmanagement, die Module und die Einführung des neuen Kreda geplant.

Die Nutzer des neuen Kreda-Systems können entsprechend der Abbildung 19 getrennt nach Mandanten (Schulen) über den Browser ihrer Endgeräte (PC, Tablet, Smartphone usw.) via einer Internetverbindung auf Kreda zugreifen. Das Hosten des neuen Systems erfolgt in einem Rechenzentrum, wobei die Daten der einzelnen Mandanten zur Sicherheit auf getrennten Datenbanken abgespeichert werden.

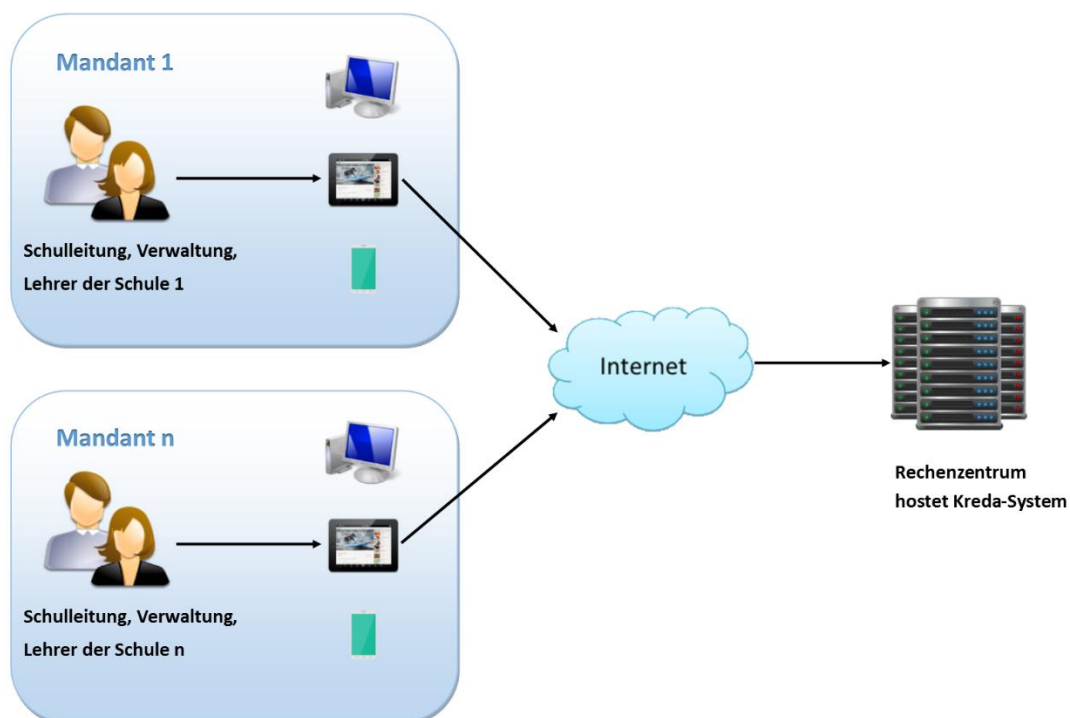


Abbildung 19 - Systemübersicht des neuen Kreda

## 5.2 Planung der technischen Struktur

Kreda soll in einem Rechenzentrum mit TÜV-Zertifizierung nach ISO 27001 gehostet werden. „Durch eine Zertifizierung wird nachgewiesen, dass in einen IT-Verbund die Standardsicherheitsmaßnahmen nach IT-Grundschutz umgesetzt wurden.“<sup>34</sup> Es werden somit die Vorgaben vom BSI (Bundesamt für Sicherheit in der Informationstechnik) zum IT-Grundschutz sichergestellt.

Um die geforderte Skalierbarkeit und Effizienz der Anwendung zu gewährleisten, soll die Anwendung auf mehreren virtuellen Servern mit dem Betriebssystem Ubuntu 64 Bit (ab Version 14.04.1 LTS) verteilt werden. Ein virtueller Server verhält sich dem Nutzer gegenüber wie ein realer physikalischer Computer. In Wirklichkeit ist ein virtueller Server allerdings kein eigenständiger Server, sondern nur eine Installation auf einem übergeordneten Server. Somit können sich mehrere virtuelle Server die Hardware-Ressourcen eines (physikalischen) Servers teilen.<sup>35</sup>

Die Abbildung 20 zeigt die geplante Serverstruktur für Kreda. Für die Startphase von Kreda sind zunächst bis zu 15 Mandanten (Schulen) geplant. Dafür sind die in der Abbildung 20 grün dargestellten Server vorgesehen. Beim späteren regulären Betrieb von Kreda ist eine Erweiterung um einen *Worker* und eine *Database* (gelb in der Abbildung 20) für jeweils 15 weitere Mandanten angedacht, um die gewünschte Performanz zu gewährleisten. Die beiden *Balancer* (OS: Ubuntu 64 Bit ab 14.04.1 LTS, CPU: Quad-Core, RAM: 4GB, HDD: 50GB) sind für die Lastverteilung der Anfragen (von den Nutzern) auf die dahinter parallel arbeitenden *Worker* (OS: Ubuntu 64 Bit ab 14.04.1 LTS, CPU: Quad-Core, RAM: 8GB, HDD: 50GB) zuständig. Auf diesen *Workers* läuft die eigentliche Anwendung von Kreda und die Daten befinden sich entkoppelt auf den *Databases* (OS: Ubuntu 64 Bit ab 14.04.1 LTS, CPU: Quad-Core, RAM: 16GB, HDD: 200GB). Für einen weiteren Performanzgewinn steht ein *Cache Server* (OS: Ubuntu 64 Bit ab 14.04.1 LTS, CPU: Quad-Core, RAM: 16GB, HDD: 50GB), welcher als schneller Puffer-Speicher für wiederholende Anfragen (Zugriffe) dient. Weiterhin ist für die beiden *Balancer* eine öffentliche IP-Adresse mit einer Internetanbindung von mindestens 20Mbit/s geplant.

---

<sup>34</sup> online: ISO 27001 Zertifizierung auf Basis von IT-Grundschutz, 2015 (27.07.15)

<sup>35</sup> vgl. LUNTOVSKYY; GÜTTER; MELNYK, 2011, S.240

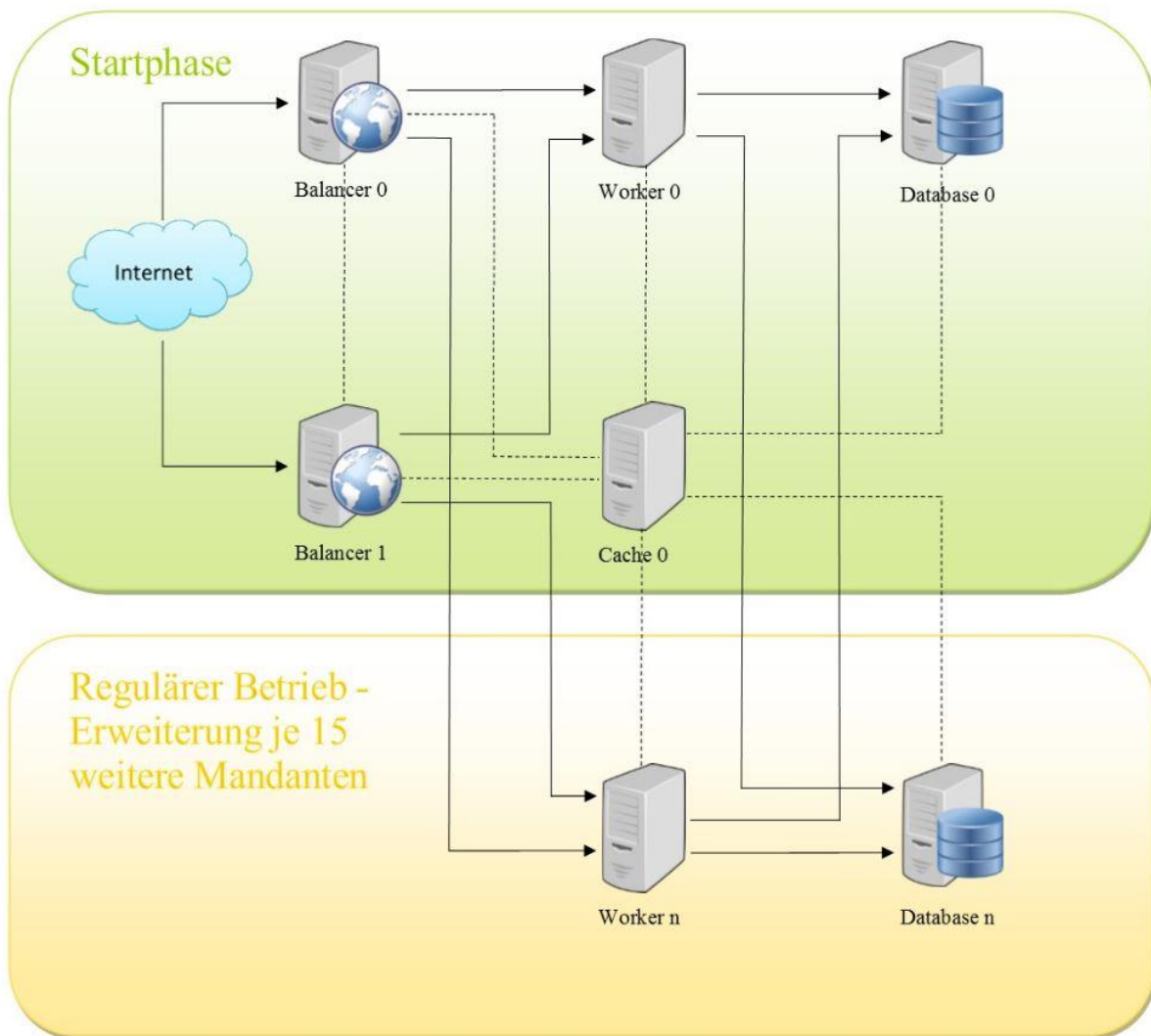


Abbildung 20 - Virtuelle Serverstruktur von Kreda

### 5.3 Planung der Systemarchitektur

Die Programm-Logik des bestehenden Kreda soll als Basis für die Modularisierung dienen. Allerdings ist eine Neustrukturierung der Systemarchitektur (refactoring) in Hinblick auf die Sicherheit, die Skalierbarkeit, die Schnittstellen, die Verfügbarkeit, die Effizienz, die Wartung und den Support notwendig. Durch die Kopplungen mit weiterführenden externen Systemen (z.B. DATEV und SFirm) sollen Synergien entstehen. Es soll eine Splittung (Modularisierung) der Datenbankstruktur unter Beachtung der bestehenden Datenbank-Logik zu Gunsten von Sicherheit, Skalierung, Verfügbarkeit und Wartung erfolgen. Für den weiteren Parallelbetrieb des bestehenden Kreda in Annaberg (für noch nicht übertragene Funktionen) ist zu diesem ein Datenbank – Adapter mit einem Unidirektionalen Datenfluss vorgesehen. Es sollen nur Daten aus dem bestehenden Kreda ins zukünftige Kreda übertragbar sein, aber nicht in umgekehrter Richtung.

Die Abbildung 21 stellt die geplante neue Systemarchitektur für Kreda dar. Dabei erfolgt eine Untergliederung in Systemmodule, Basismodule und Anwendungsmodule.

Die Systemmodule, wie *System* (zuständig für z.B. Datenbank-Protokollierung und die Update-Verarbeitung der Anwendung), *Assistance* (z.B. Hilfe-Funktionalität, Unterstützung für K&W-Online-Supportsystem) und der *Gatekeeper* (z.B. Zugangskontrolle, Mandanten- und Rechte-Verwaltung) bilden das neue Grundgerüst der Anwendung. Des Weiteren sind für die eigentlichen Anwendungsmodule (*Zensuren*, *Fakturierung*) zusätzlich die Basismodule, wie *Verwaltung* (Stammdaten) und *Daten-Import* bzw. *-Export*, erforderlich.

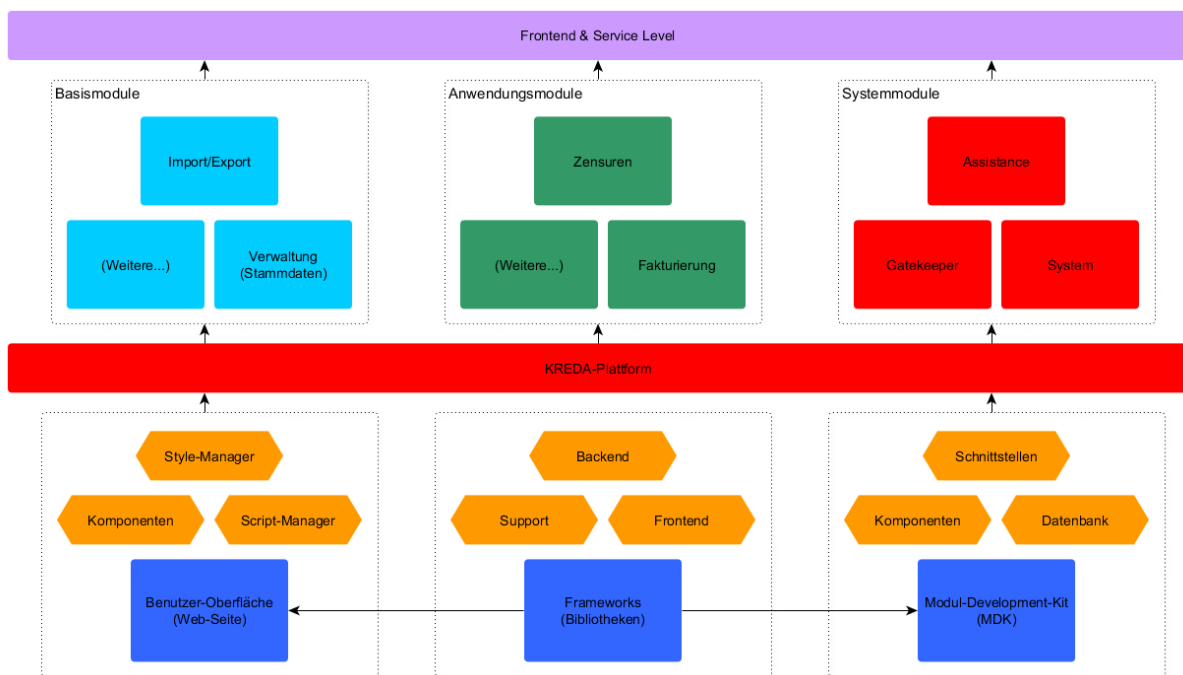


Abbildung 21 - Systemarchitektur

## 5.4 Entwicklungsumgebung

Für die Programmierung des zukünftigen Kreda ist die Skriptsprache PHP vorgesehen, da die K&W Informatik eine langjährige Erfahrung mit dieser besitzt und auch große Websysteme, wie z.B. Magento<sup>36</sup> oder Typo3<sup>37</sup>, ebenfalls auf PHP basieren.

Im Bereich des Frameworks für das Backend stehen die folgenden Frameworks zur Auswahl:

- Symfony<sup>38</sup> (Web Application Framework)
- MOC<sup>39</sup> bis Mark IV (Custom-Web-Development Framework)
- MOC Mark V

Symfony und MOC bis Mark IV besitzen eine hohe Komplexität, was eine zeitaufwändige Einarbeitung für die Software-Entwickler zur Folge hat. Hingegen ist MOC Mark V einfach strukturiert und beinhaltet updatebare Symfony

<sup>36</sup> Online-Shopsoftware

<sup>37</sup> Content-Management-System

<sup>38</sup> weitere Informationen online: Symfony Homepage, 2015 (27.07.15)

<sup>39</sup> Modular Object Constructor, Eigenentwicklung der K&W Informatik

Grundkomponenten. Somit fiel die Wahl auf MOC Mark V für das Backend-Framework, da es durch seine geringe Einarbeitungszeit für die Entwickler schneller einsetzbar ist.

Im Rahmen des Frameworks für das Frontend sollen Bootstrap (CSS-Framework) und Bootflat (Komponenten-Framework) genutzt werden, weil diese vorgefertigte Gestaltungsvorlagen beinhalten. Somit ergibt sich durch die Verwendung dieser eine Zeitersparnis im Vergleich zur Erstellung eigener Gestaltungsvorlagen. Ebenfalls von Vorteil ist die automatisch unterstützte RWD-Fähigkeit von Bootstrap und Bootflat. Bootstrap enthält HTML und CSS basierende Gestaltungsvorlagen für Typografie, Formulare, Buttons, Tabellen, Grid-Systeme, Navigations- und andere Oberflächengestaltungselemente. Diese werden durch Bootflat erweitert, welches basierend auf Bootstrap zusätzliche Style-Komponenten enthält.

## 5.5 Qualitätsmanagement – Systemtests

Auch im Bereich der Software-Entwicklung spielt das Qualitätsmanagement eine immer wichtigere Rolle. „Unter der Software-Qualität versteht man die Gesamtheit der Merkmale und Merkmalswerte eines Softwareprodukts, die sich auf dessen Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse zu erfüllen.“<sup>40</sup> Die Software-Qualität wird durch die funktionelle Eignung, Performanz und Effizienz, Kompatibilität, Benutzerfreundlichkeit, Zuverlässigkeit, Sicherheit, Wartbarkeit und Portabilität bestimmt. Die Entwicklung des zukünftigen Kreda soll in den Bereichen Performanz und Effizienz, Zuverlässigkeit, Sicherheit und Wartbarkeit durch die folgenden Tests unterstützt werden:

- SensioLabsInsight:
  - Analyse der Anwendung auf Sicherheit, Performanz, Architekturschwächen
  - Bericht über mögliche Fehlerquellen der Anwendung
- Travis CI:
  - Ausführung von automatisierten Systemtests auf virtuellen Servern mit unterschiedlichen Software-Konfigurationen wie z.B. PHP Versionen
- Code Climate:
  - Analyse des Programmcodes auf intuitive Verständlichkeit und strukturelle Einheit
  - Bericht über mögliche Optimierungen des Quellcodes
- Blackfire
  - Analyse des Programmablauf auf Ressourcen-Nutzung (Last)
- Coveralls
  - Bericht über die Vollständigkeit der Systemtests

---

<sup>40</sup> PFEIFER; SCHMITT, 2014, S.837

---

## 5.6 Planung der Module

### 5.6.1 Systemmodule

Angesichts des Umfangs der gewünschten Module von Kreda sollen hier die Planung der Systemmodule, Basismodule und Anwendungsmodule nur angerissen werden. Im Weiteren sind für die Umsetzung dieser die Erstellung von ausführlichen Pflichtenheften erforderlich (nicht Inhalt dieser Bachelorthesis).

Die Systembasis von Kreda soll durch die Systemmodule System, Gatekeeper und Assistance sichergestellt werden. Im Modul System sollen der Datenbankzugriff der Anwendung, die Update-Verarbeitung der Anwendung und die Datenbank-Protokollierung realisiert werden. Um die Sicherheit zu gewährleisten werden die Daten eines jeden Mandanten auf getrennten eigenen Datenbanken abgelegt und jeder Mandant erhält nur Zugriff auf seine eigenen Datenbanken. Für die Weiterentwicklung von Kreda wird eine Update-Verarbeitung integriert, mit welcher die notwendigen Änderungen an den Datenbanken automatisch vorgenommen werden und der Quellcode upgedatet wird. Mit der Datenbank-Protokollierung können Änderungen der Nutzer an den Daten genau nachvollzogen und diese bei Bedarf rückgängig gemacht werden.

Über den Gatekeeper soll die Zugangskontrolle, die Mandanten- und Rechteverwaltung abgedeckt werden. Der Gatekeeper ist ein globales System, d.h. die Daten für alle Mandanten befinden sich auf einer Datenbank. Für den Zugang von Mitarbeitern der Schulen soll zusätzlich zum Login und Passwort eine Hardware gestützte Authentifizierung per YubiKey erfolgen. Innerhalb der Mandantenverwaltung ist ein Wechsel zwischen den einzelnen Mandanten möglich, neue Mandanten können angelegt werden und Mandanten bearbeitet werden. Beim Wechsel des Mandanten wird die komplette Datenbasis der Anwendung gegen den neuen Mandanten ausgetauscht. Der Wechsel ist z.B. für den Support der Anwendung erforderlich. Außerdem soll die Rechteverwaltung per Nutzer-Rollen über das Zugriffslevel auf Modul-Ebene und darunter auf Funktions- bzw. Services-Ebene des Moduls erfolgen.

Mit dem Assistance-Modul soll in Kreda eine Hilfe-Funktion für den Nutzer integriert werden und der Support von Kreda, über eine automatische Ticket-Erstellung bei einem auftretenden Software-Fehler, unterstützt werden. Dieses Ticket kann vom Nutzer ans Ticketsystem der K&W-Informatik versendet werden. Es enthält die automatisch generierte Fehlermeldung sowie die Quellcodestelle, an der der Fehler aufgetreten ist und kann zusätzlich vom Nutzer kommentiert werden.

## 5.6.2 Basismodule

Neben den Systemmodulen sind für die Einführung von Kreda die Basismodule Verwaltung (Stammdaten) sowie der Daten-Import bzw. –Export Voraussetzung. Die Hauptdaten der Verwaltung bestehen aus den Personendaten. Innerhalb der Personendaten erfolgt eine Gliederung der Metadaten für die folgenden Personengruppen: Interessenten, Schüler, Sorgeberechtigte, Mitarbeiter (Pädagogisches Personal und Verwaltungspersonal) und Sonstige. Ebenfalls soll es möglich sein, dass eine Person zu mehreren Personengruppen gehört (z.B. ist ein Mitarbeiter der Schule ebenfalls Sorgeberechtigter eines Schülers). Die Personen können untereinander über Personenbeziehungen verknüpft werden. Sowohl den Personen als auch den Firmen können dynamisch erweiterbar Adress- und Kontaktdaten zugeordnet werden. Innerhalb der Zeiten können die Halb- und Schuljahre abhängig vom Bildungsgang erfasst werden. Neben der Möglichkeit, Personen zu Personengruppen zusammen zufassen ist eine Gruppierung von Schülern zu Klassen vorgesehen. Weiterhin können den Schülern Unterrichtsfächer, welche sie belegen, zugeordnet werden. Dabei sind die Fächer in die Kategorien Religionsunterricht, Fremdsprache, Profil, Neigungskurs, Vertiefungskurs, Fachrichtung, Wahlpflichtfach, Pflichtfach, Wahlfach und Arbeitsgruppe (AG) einteilbar und ihnen können Mitarbeiter (Lehrer) entsprechend ihrem Lehrauftrag zugewiesen werden.

Damit die Schulen nach der Fertigstellung des neuen Kreda nicht alle Daten (aus ihren Altsystemen) mit der Hand einpflegen müssen, soll die Datenmigration durch das Basismodul Daten-Import/Export gewährleistet werden. Dazu sind für die Stammdaten ein Import aus FuxSchool und ein allgemeiner Import im CSV-Format vorgesehen. Hingegen für die Datenmigration der Daten des alten Kreda in Annaberg kann der bereits erwähnte Datenbank – Adapter genutzt werden. Im Zusammenhang mit der Fakturierung ist ebenfalls ein Export an DATEV und SFirm geplant.

## 5.6.3 Anwendungsmodule

Die beiden wichtigsten Anwendungsmodule für die Schulen sind das Zensuren-Modul inklusive Zeugnisdruck und das Fakturierungsmodul. Nachrangig sollen später die weiteren Anwendungsmodule Stundenpläne und Zeitmanagement, Sitzpläne und die Lehrstoffplanung in Kreda umgesetzt werden.

Im Zensuren-Modul können durch die Lehrer die Leistungen (Leistungskontrollen, mündliche Leistungen, Klausuren, Klassenarbeiten usw.) der Schüler bewertet werden. Dafür ist eine Unterstützung der klassischen Noten (1-6) für die Sekundärstufe I und das Punktesystem (0-15) der Sekundärstufe II Voraussetzung. Die Auswertung der Leistung soll ebenfalls über die Punktevergabe von Teilaufgaben erfolgen können. Die einzelnen Leistungen können ihrer Kategorie entsprechend für



die automatische Berechnung der Stichtagnoten gewichtet werden. Für den Zeugnisdruck sind neben diesen Stichtagsnoten für die belegenden Fächer des Schülers noch die Kopfnoten für Betragen, Fleiß, Mitarbeit und Ordnung erforderlich. Diese werden vom Klassenlehrer des Schülers vorgeschlagen und anschließend von der Schulleitung festgesetzt. Wegen der verschiedenen Zeugnisvorlagen der unterschiedlichen Schulen soll für die Erstellung dieser ein graphischer Designer in Kreda integriert werden.

Mit dem Fakturierungsmodul in Kreda können die Schulen den Schüler mit ihren Sorgeberechtigten (Eltern) Rechnungen für erbrachte Dienstleistungen (Schulgeld, Hortgeld usw.), Kosten für bereitgestellte Artikel (Schulspeisung, Arbeitsmaterial, Kopiergeld usw.) und anteilige Projektkosten (Exkursionen, Klassenfahrten, Theatervorführungen usw.) stellen. Diese Rechnungsstellung soll über eine Sammel-Fakturierung über eine oder mehrere Leistungen und mehrere Personen (Schüler) erfolgen. Nach der Fakturierung können die Rechnungen an DATEV und SFirm exportiert und die eingehenden Zahlungseingänge für die Verwaltung der offenen Posten in Kreda importiert werden.

## **5.7 Einführung und Parallelbetrieb**

Die Einführung von Kreda an den Schulen soll schrittweise im Funktionsumfang und in der Anzahl der teilnehmenden Schulen erfolgen. Kreda soll für die ersten Schulen bereits nur mit dem Basismodul Verwaltung (Stammdaten) und Daten-Import/Export starten. Somit ist zur Einführung von Kreda nicht der voll benötigte Funktionsumfang gegeben und die ersten teilnehmenden Schulen sollen ihre vorhandenen Altsysteme zur Verwaltung ihrer Schule vorerst parallel weiterbetreiben. Durch den Parallelbetrieb ist es möglich, Kreda im produktiven Einsatz ausgiebig zu testen, ohne die Prozesse der Verwaltung zu beeinflussen. Allerdings ist der Parallelbetrieb mit einem zusätzlichen Aufwand der Mitarbeiter der Schule verbunden. Weiterhin ermöglicht der Parallelbetrieb, noch nicht erfasste Anforderungen an Kreda, welche sich erst durch die Nutzung der Software ergeben, umzusetzen. Nach der Verwaltung (Stammdaten) soll das Anwendungsmodul Zensuren (Notenverwaltung) eingeführt werden und anschließend das Anwendungsmodul Fakturierung. Außerdem sollen zur Einführung von Kreda für die Mitarbeiter Schulungen für den Umgang mit dem neuen Kreda angeboten werden.

## **6 Zusammenfassung und Ausblick**

### **6.1 Zusammenfassung**

Zum Abschluss meiner Bachelorthesis erfolgt eine kurze Zusammenfassung der Arbeit; die Prüfung, ob die Zielsetzung erreicht wurde und ein Ausblick auf die möglichen weiteren Schritte.

Im 2. Kapitel wurde die bestehende Software (Kreda) zur Verwaltung von Schulen analysiert. Dabei wurde auf vorhandene Web-Client/Server-Architektur eingegangen, um eine Übersicht über das System hinter Kreda zu geben. Anschließend wurde die technische Struktur aus einem Web Server und einem Datenbank Server analysiert. Die Programmierung des bisherigen Kreda erfolgte seitenbasiert und prozedural. Im Bereich der Funktionen von Kreda wurden die allgemeinen Funktionen (MarktUp, YubiKey), die Administration (Stammdatenverwaltung), die Notenverwaltung, die Sitzpläne, die Stundenpläne, die Klassen, die Elternansicht, die Lehrstoffplanung (Unterrichtsinhalte) und die Startseite betrachtet.

Danach wurden im 3. Kapitel die Anforderungen an die zukünftige Software betrachtet, damit diese auch für weitere Schulen der Schulstiftung zur Verwaltung der Schule eingesetzt werden kann und die Weiterentwicklung anhand der kommenden Herausforderungen erfolgen kann. Die Vorgaben wurden in allgemeine, technische und funktionelle Anforderungen gegliedert. Unter den technischen Anforderungen wurde auf die Skalierbarkeit und Schnittstellen, die Verfügbarkeit und Effizienz, die Sicherheit, die Wartung und Support und die RWD-Fähigkeit im Zusammenhang mit der Darstellung eingegangen. Im Anschluss wurden die funktionellen Vorgaben über das Login-System und die Nutzerverwaltung, die Stammdatenverwaltung, die Zensuren (Notenverwaltung), die Fakturierung, die Stundenpläne und das Zeitmanagement, die Sitzpläne und die Lehrstoffplanung abgehandelt. Abschließende wurde eine Bewertung und Priorisierung der Anforderung durchgeführt, um eine schnellere Einführung des neuen Kreda zu ermöglichen. Dabei ergaben für die funktionellen Anforderungen das Login-System, die Nutzerverwaltung und die Stammdatenverwaltung die höchste Priorität.

Um die Planung der Neustruktur durchführen zu können, wurde im 4. Kapitel die vorhandene Software anhand der allgemeinen, technischen und funktionellen Vorgaben bewertet. Die Anforderungen im Bereich Notenverwaltung, Sitzpläne, Lehrstoffplanung werden vom bestehenden Kreda erfüllt. Hingegen werden im Rahmen der allgemeinen Vorgaben die Sicherheit, Darstellung, Login-System, Nutzerverwaltung, Stammdatenverwaltung, Stundenpläne und Zeitmanagement von Kreda momentan nur teilweise erfüllt. Weiterhin werden die Anforderungen

Skalierbarkeit, Modularisierung, Verfügbarkeit, Effizienz, Wartung, Support, RWD-Fähigkeit, Zeugnisdruck, Fakturierung nicht erfüllt. Für das weitere Vorgehen der Neustrukturierung ergab sich als beste Möglichkeit die Eigenentwicklung auf Basis der vorhandenen Software.

Nachfolgend wurde im 5. Kapitel die Planung für die Umsetzung der Neustrukturierung durchgeführt. Dabei wurde auf eine neue technische Struktur für Kreda in Form der Verteilung der zukünftigen Last auf mehrere virtuelle Server hingewiesen. Die Systemarchitektur wurde modularisiert in die Systemmodule (System, Gatekeeper, Assistance), die Basismodule (Verwaltung, Import/Export) und die Anwendungsmodule (Zensuren, Fakturierung usw.). Die Module sind unabhängig voneinander und der Datenaustausch untereinander erfolgt nur über definierte Schnittstellen. Für die Entwicklung soll die Skriptsprache PHP mit dem Backend-Framework MOC Mark V und den Frontend-Frameworks Bootstrap und Bootflat zum Einsatz kommen. Das Qualitätsmanagement wird durch automatische Tests unterstützt. Abschließend soll die Einführung des neuen Kreda schrittweise im Funktionsumfang (beginnend mit der Verwaltung) und in der Anzahl der teilnehmenden Schulen erfolgen. Zur Einführung werden die Daten aus den Altsystemen der Schulen per Import migriert.

## **6.2 Erreichung der Zielsetzung und Ausblick**

Das Ziel dieser Bachelorthesis, die Erarbeitung eines Plans zur Neustrukturierung für die bestehende Software (Kreda) zur Verwaltung von Schulen wurde mit Erfolg erreicht. Als nächste Schritte für die Umsetzung der Neustrukturierung sind die Erstellung der Pflichtenhefte anhand der erfassten Anforderungen für die Basismodule Verwaltung und Import/Export und die Anwendungsmodule Zensuren und Fakturierung erforderlich. Nach der Einführung der Basismodule Verwaltung und Import/Export kann das neue Kreda aufgrund seiner neuen Struktur, anhand von kommenden Anforderungen der teilnehmenden Schulen, weiterentwickelt werden. Weiterhin kann Kreda weiteren Schulen, welche sich nicht in der Trägerschaft der Schulstiftung der Evangelisch-Lutherischen Landeskirche Sachsens befinden, angeboten werden. In dieser Hinsicht besteht momentan Interesse der Evangelischen Schulstiftung in Mitteldeutschland (Thüringen, Sachsen-Anhalt).

## Quellenverzeichnis

All4Teachers. Leipzig, 2015, In:

<http://all4teachers.de/>

Apache HTTP Server Project. Los Angeles, 2015, In:

<http://httpd.apache.org/>

Atlantis Schulverwaltung. Bad Abbach, 2015, In:

<https://www.swh-heider.de/index.php/atlantis.html>

BEDNER, Mark: Cloud Computing: Technik, Sicherheit und rechtliche Gestaltung. Kassel, 2013

DB-Engines Ranking. Wien, 2015, In:

<http://db-engines.com/de/ranking>

Europäische Parlament; Rat der europäischen Union: VERORDNUNG (EU) Nr. 260/2012 DES EUROPÄISCHEN PARLAMENTS UND DES RATES. Brüssel, 2012

FuxSchool Produkte. Dresden, 2015, In:

[http://www.fuxschool.de/index.php?ILNK=fuxschool\\_produkte&iL=1&PHPSESSID=8017715bfed2db085536458780a2ad7f](http://www.fuxschool.de/index.php?ILNK=fuxschool_produkte&iL=1&PHPSESSID=8017715bfed2db085536458780a2ad7f)

GEISLER, Frank: Datenbanken: Grundlagen und Design. 5. Aufl. Wachtendonk, 2014

GOLDSTEIN, Elmar: Schnelleinstieg DATEV, Freiburg, 2013

HILDEBRAND, Knut: Praxis der Wirtschaftsinformatik.

Bd. 279 : Stammdatenmanagement. Heidelberg, 2011

HUDSON, Paul: PHP in a nutshell. Köln, 2006

Indiware Homepage. Mühlhausen, 2015, In:  
<http://www.indiware.de/index.php?page=home>

ISO 27001 Zertifizierung auf Basis von IT-Grundschutz. Bonn, 2015, In:  
[https://www.bsi.bund.de/DE/Themen/ZertifizierungundAnerkennung/Zertifizierung27001/GS\\_Zertifizierung\\_node.html](https://www.bsi.bund.de/DE/Themen/ZertifizierungundAnerkennung/Zertifizierung27001/GS_Zertifizierung_node.html)

LUNTOVSKYY, Andriy; GÜTTER, Dietbert; MELNYK, Igor: Planung und Optimierung von Rechnernetzen: Methoden, Modelle, Tools Für Entwurf, Diagnose und Management im Lebenszyklus von Drahtgebundenen und Drahtlosen Rechnernetzen. Berlin, 2011

NAHRSTEDT Harald: C++ für Ingenieure: Effizient Programmieren erlernen. Heidelberg, 2009

SAMBLEBEN, Jörg; SCHUMACHER, Stefan: Informationstechnologie und Sicherheitspolitik: Wird der dritte Weltkrieg im Internet ausgetragen? Magdeburg, 2013

SCHMIDT, Christian: Management Komplexer IT-Architekturen. Berlin, 2009

Schulstiftung Profil. Radebeul, 2015, In:  
<http://www.evangelische-schulen-sachsen.de/profil-2/>

Schulstiftung Stiftungstätigkeit. Radebeul, 2015, In:  
<http://www.evangelische-schulen-sachsen.de/stiftungstaetigkeit/>

SIMON, Fabian: Fakturierung. Berlin, 2015, In:  
<http://www.rechnungswesen-verstehen.de/grundlagen-buchfuehrung/fakturierung.php>

Symfony Homepage. Köln, 2015, In:  
<https://symfony.com/>

PFEIFER, Tilo [Hrsg.]; SCHMITT, Robert [Hrsg.]: Masing Handbuch Qualitätsmanagement. 6. Auf. München, 2014

POGUNTKE, Werner: Basiswissen IT-Sicherheit: Das Wichtigste für den Schutz von Systemen und Daten. Fernwald, 2007

VILT, Lajos: ITIL konformes Incident Management im Bereich der Software-Entwicklung: Chancen im Einsatz von Open Source Software. Hamburg, 2014

VOGEL, Oliver; ARNOLD, Ingo; CHUGHTAI, Arif u.a.: Software-Architektur: Grundlagen – Konzepte – Praxis. Heidelberg, 2009

WENZ, Christian; HAUSER, Tobias: HTML: Webseiten einfach selber machen. Hallbergmoos, 2005

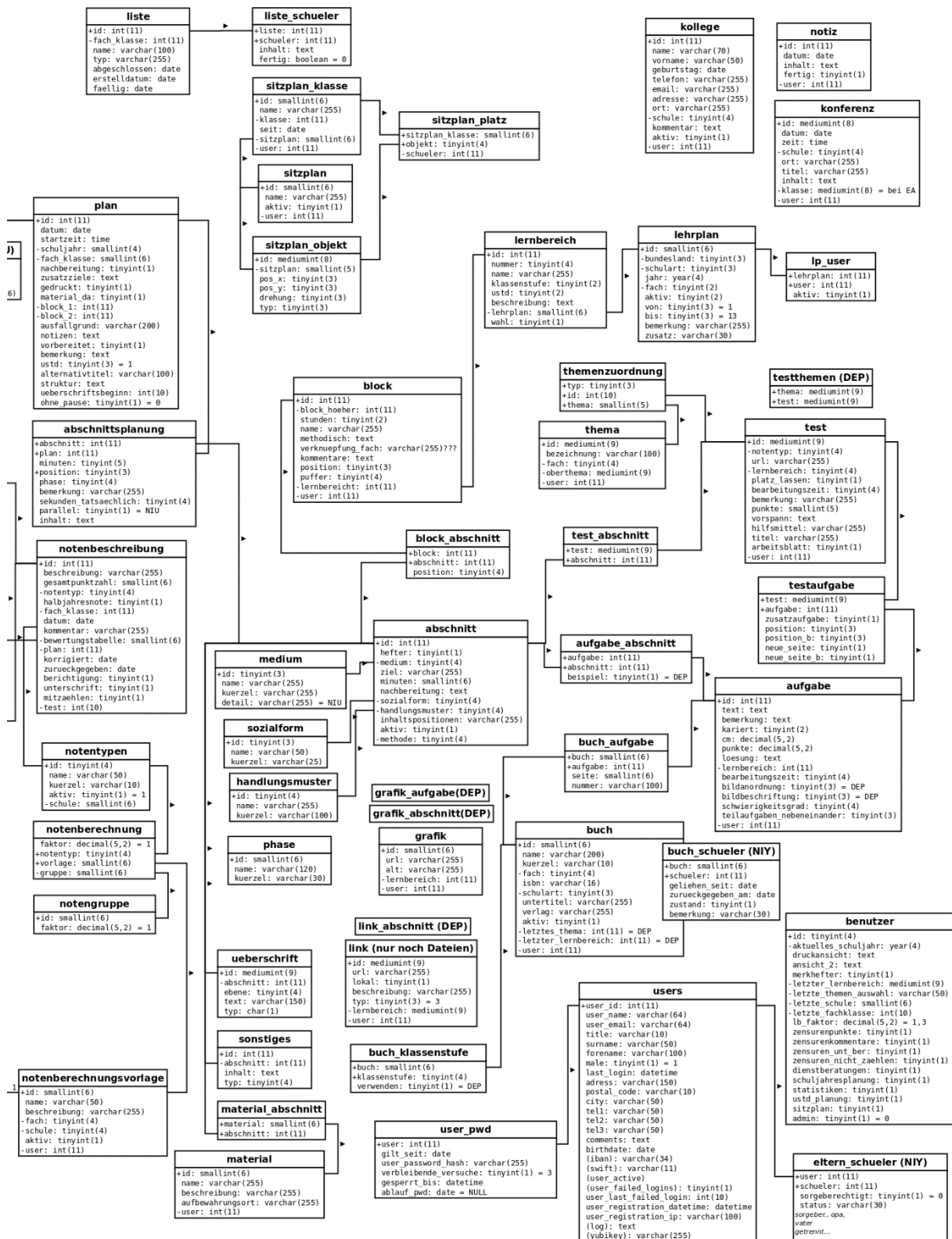
ZILLGENS, Christoph: Responsive Webdesign: Reaktionsfähige Websites gestalten und umsetzen. München, 2012

## Anhangverzeichnis

**Anhang 1**          Datenbankstruktur des bestehenden Kredas







## Ehrenwörtliche Erklärung

"Ich erkläre hiermit ehrenwörtlich",

1. dass ich meine Bachelorthesis mit dem Thema: *Analyse & Planung der Neustrukturierung einer bestehenden Software zur Verwaltung von Schulen* ohne fremde Hilfe angefertigt habe,
2. dass ich die Übernahme wörtlicher Zitate aus der Literatur sowie die Verwendung der Gedanken anderer Autoren an den entsprechenden Stellen innerhalb der Arbeit gekennzeichnet habe und
3. dass ich meine Bachelorthesis bei keiner anderen Prüfung vorgelegt habe.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Zwiesel, 29.07.15

Ort, Datum



Unterschrift