



TECHNISCHE HOCHSCHULE MITTELHESSEN

**THM**

**CAMPUS  
GIESSEN**

**MNI**

Mathematik, Naturwissenschaften  
und Informatik

# Ausarbeitung eines Konzepts für Datenschutz und Datensicherheit in E-Learning Systemen anhand des Beispiels der Lernkarten Plattform THMcards

## MASTER-THESIS

im Studiengang Informatik

zur Erlangung des akademischen Grades

MASTER OF SCIENCE

**Autor:**

Matthias Zimny  
matthias.zimny@mni.thm.de  
MatNr. 5073647

**Version vom:**

22. November 2016

**1. Referent:**

Prof. Dr. Klaus Quibeldey-Cirkel

**2. Korreferent:**

Hr. Christoph Thelen (M.Sc.)



# Kurzzusammenfassung

In dieser Arbeit wird die Konzeption und Implementierung geeigneter Datenschutz- und Datensicherheitsmaßnahmen in E-Learning Systemen aufgezeigt, deren Ziel es ist, dem Leser darzulegen, welche datenschutzrechtlichen Aspekte durch solche Systeme einzuhalten sind und Maßnahmen vorgestellt, um ein E-Learning System in der technischen Sicherheit zu verbessern. Hierzu werden als Basis die Komponenten eines E-Learning Systems, die durch die vorgestellten Maßnahmen geschützt werden sollen, näher erläutert. Anschließend werden die Grundsätze des Datenschutzes, die bei der Nutzung von personenbezogenen Daten zu beachten sind, näher gebracht sowie die Haftung und Verantwortlichkeiten eines Online-Dienstes bei eingestellten rechtswidrigen Inhalten dargelegt. Darüber hinaus sollen die Schutzziele sowie die technischen und organisatorischen Maßnahmen, die berücksichtigt werden müssen, um die Datensicherheit von E-Learning Systemen zu gewährleisten, aufgezeigt werden. Anschließend wird die Lernkartenplattform THMcards vorgestellt, um die nötige Grundlage zu schaffen, die vorgestellten Datenschutz- und Datensicherheitsmaßnahmen in das System zu integrieren. Zum Schluss werden diese Maßnahmen durch die Implementierung in THMcards verdeutlicht.



# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>vii</b>
------------------------------	------------

<b>Tabellenverzeichnis</b>	<b>viii</b>
----------------------------	-------------

<b>1. Einleitung</b>	<b>1</b>
1.1. Motivation . . . . .	1
1.2. Ziel der Arbeit . . . . .	2
1.3. Gliederung . . . . .	3
<b>2. Analyse von E-Learning Webseiten</b>	<b>5</b>
2.1. Funktionsbereiche eines E-Learning Systems . . . . .	5
2.1.1. Administrationswerkzeuge . . . . .	6
2.1.2. Evaluierungswerkzeuge . . . . .	6
2.1.3. Autorenwerkzeuge und Darstellung von Inhalten . . . . .	7
2.1.4. Kommunikations- und Kooperationswerkzeuge . . . . .	8
2.1.5. Weitere Werkzeuge . . . . .	8
2.2. Technische Bausteine eines E-Learning Systems . . . . .	9
2.2.1. Internet-Anbindung . . . . .	9
2.2.2. Aktive Inhalte und dynamische Webseiten . . . . .	9
2.2.3. Datenbanksysteme . . . . .	10
2.2.4. Zahlungssysteme . . . . .	12
<b>3. Datenschutz</b>	<b>13</b>
3.1. Personenbezogene Daten . . . . .	14
3.2. Identitätsinfrastruktur . . . . .	15
3.2.1. Identitätsmanagement . . . . .	16
3.3. Datenschutzgrundsätze . . . . .	17
3.3.1. Verbotsvermutung und Erlaubnisvorbehalt . . . . .	17
3.3.2. Datenvermeidbarkeit und Datensparsamkeit . . . . .	18
3.3.3. Zweckbindung . . . . .	19
3.3.4. Transparenz . . . . .	19
3.3.5. Datensicherheit . . . . .	20
3.4. Weitere datenschutzrechtliche Aspekte . . . . .	20
3.4.1. Impressumspflicht . . . . .	21
3.4.2. Cookie-Richtlinien . . . . .	22

<b>4. Haftung von Online-Diensten</b>	<b>25</b>
4.1. Internet Service Provider . . . . .	25
4.1.1. Content-Provider . . . . .	25
4.1.2. Access-Providers . . . . .	26
4.1.3. Host-Provider . . . . .	26
4.2. Rechtsverletzungen . . . . .	26
4.2.1. Persönlichkeitsverletzung . . . . .	27
4.2.2. Urheberrechtsverletzungen . . . . .	27
4.2.3. Hyperlinks auf fremde Inhalte . . . . .	29
4.3. Verantwortlichkeit der Host-Provider . . . . .	31
4.3.1. Festes Verfahren zur Löschung eines rechtswidrigen Inhalts . . . . .	31
<b>5. Datensicherheit</b>	<b>33</b>
5.1. Schutzziele . . . . .	35
5.1.1. Vertraulichkeit . . . . .	35
5.1.2. Integrität . . . . .	36
5.1.3. Verfügbarkeit . . . . .	37
5.1.4. Authentizität . . . . .	37
5.1.5. Verbindlichkeit . . . . .	38
5.2. Technische und organisatorische Maßnahmen . . . . .	38
5.3. Angriffsszenarien . . . . .	41
5.3.1. SQL- und NoSQL-Injection . . . . .	41
5.3.2. Cross-Site-Scripting . . . . .	43
5.3.3. Session Hijacking und Session Fixation . . . . .	45
5.3.4. Cross-Site-Request-Forgery . . . . .	46
5.3.5. Denial of Service . . . . .	47
<b>6. Die Open-Source-Anwendung THMcards</b>	<b>49</b>
6.1. Zieldefinition . . . . .	49
6.2. Re-Implementierung von THMcards . . . . .	52
6.2.1. Evaluierung eines geeigneten Web-Frameworks . . . . .	52
6.2.2. Codestruktur . . . . .	58
6.3. Technische Aspekte von THMcards . . . . .	60
6.3.1. Meteor . . . . .	61
6.3.2. Node.js . . . . .	66
6.3.3. MongoDB . . . . .	67
<b>7. Anwendung und Implementierung in THMcards</b>	<b>71</b>
7.1. Implementierung geeigneter Datenschutzmaßnahmen . . . . .	71
7.1.1. Informationspflichten . . . . .	72
7.1.2. Einwilligung . . . . .	76
7.1.3. Urheberrecht . . . . .	77
7.1.4. Back-End . . . . .	77
7.1.5. Notifications . . . . .	79

7.2. Implementierung geeigneter Datensicherheitsmaßnahmen . . . . .	80
7.2.1. Packages . . . . .	81
7.2.2. Datenbank . . . . .	83
7.2.3. Rollen . . . . .	87
7.2.4. Settings-Datei . . . . .	90
<b>8. Abschluss</b>	<b>91</b>
8.1. Zusammenfassung . . . . .	91
8.2. Fazit und Ausblick . . . . .	93
<b>Literaturverzeichnis</b>	<b>i</b>
<b>A. Anhang</b>	<b>xii</b>
A.1. Impressum . . . . .	xii
A.2. Datenschutzerklärung . . . . .	xiii
A.3. Allgemeine Geschäftsbedingungen . . . . .	xvii
A.4. Verfahrensverzeichnis . . . . .	xxiv
<b>Eidesstattliche Erklärung</b>	<b>xxvii</b>



# Abbildungsverzeichnis

2.1. Übersicht über alle gelernten Kartensätze in THMcards . . . . .	7
2.2. Aufbau eines Datenbanksystems [Data] . . . . .	11
3.1. Identitätsmanagement in THMcards . . . . .	17
5.1. PDCA-Zyklus [PDC] . . . . .	35
6.1. Die Commits eines Web-Frameworks . . . . .	54
6.2. Dateistruktur von THMcards . . . . .	59
6.3. Aufbau von Client und Server in Meteor 1.0 [Mon15] . . . . .	62
6.4. Die Architektur von Node.js [Hec16, Seite 12] . . . . .	66
6.5. Das konzeptionelle Modell von Node.js [Nod] . . . . .	67
6.6. Popularität von dokumentenbasierten Datenbanksystemen [DE16]	68
6.7. Aufbau der dokumentenbasierten Datenbank MongoDB [Sut12] .	69
7.1. Cookie Consent-Plug-In in THMcards . . . . .	74
7.2. Elektronische Einwilligung in THMcards . . . . .	76
7.3. Creative-Commons-Lizenzen in THMcards . . . . .	77
7.4. Dashboard-View im Back-End von THMcards . . . . .	78
7.5. Benutzeransicht im Back-End von THMcards . . . . .	79
7.6. Übersicht der Benachrichtigungen im Back-End von THMcards .	80
7.7. Veröffentlichung von Karten . . . . .	84
7.8. Fehlerseite in THMcards . . . . .	88
7.9. Menüleiste eines Admin-User in THMcards . . . . .	89

# Tabellenverzeichnis

5.1. Abgrenzung von Security und Safety [DHF07, Seite 488-510] . . .	34
6.1. Größe der Framework Community (Stand 12.09.2016) . . . . .	53
6.2. Größe der einzelnen Frameworks (Stand 12.09.2016) . . . . .	55

# Listingverzeichnis

5.1.	SQL-Befehl zur Authentifizierung des Benutzers . . . . .	42
5.2.	Manipulierter SQL-Befehl . . . . .	42
5.3.	NoSQL-Befehl zur Authentifizierung des Benutzers . . . . .	42
5.4.	Manipulierter NoSQL-Befehl . . . . .	42
5.5.	Normaler Aufruf der Webseite . . . . .	44
5.6.	HTTP-Anfrage im HTML-Code . . . . .	47
6.1.	Definition einer Collection . . . . .	63
6.2.	Veröffentlichung der Daten im Server . . . . .	63
6.3.	Abonnieren der Daten im Client . . . . .	63
6.4.	Definition einer Funktion im Server . . . . .	64
6.5.	Method Call im Client . . . . .	64
6.6.	Veröffentlichung der Daten im Client . . . . .	64
6.7.	Iteration von Collections in Meteor . . . . .	65
6.8.	Bedingungen in Meteor-Templates . . . . .	65
7.1.	Durch Browser Policy blockierte Aktionen in THMcards . . . . .	82
7.2.	Eine durch Browser Policy definierte Ausnahme in THMcards . . . . .	83
7.3.	Filtern der Dokumente anhand der Benutzer-ID in THMcards . . . . .	84
7.4.	Fehlerhafte Veröffentlichung des Benutzernamen . . . . .	85
7.5.	Veröffentlichung des Benutzernamen in THMcards . . . . .	85
7.6.	allow/deny-Regeln in der Meteor.users-Collection in THMcards . . . . .	86
7.7.	Einsatz vom isInRole-Helper in THMcards . . . . .	89
7.8.	Absicherung der Server-Methoden durch Rollen in THMcards . . . . .	90
7.9.	Festlegung des Admin-Users in THMcards . . . . .	90



# 1. Einleitung

Die Revolution des Lernens durch E-Learning begann bereits vor dreißig Jahren. Wissen, das man sich früher mit selbst gesteuertem Lernen durch offene Fernkurse aneignen konnte, entwickelte sich zum computergestützten Lernen durch das Web. Heute werden inzwischen an nahezu allen Hochschulen für die Durchführung des Lernens internetbasierte Systeme verwendet, die das E-Learning unterstützen. Typische Beispiele sind Moodle als Open-Source-Plattform oder Blackboard als marktführendes kommerzielles Produkt. Beim aktuellen Stand der Technik fällt auf, dass viele der entwickelten E-Learning Systeme den Fokus auf die technische Umsetzung gelegt haben, um Lerninhalte anzubieten und zu überbringen. Dabei wurden jedoch die datenschutzrechtlichen Anforderungen sowie die Sicherheit der Daten völlig vernachlässigt.

Das Datenschutzgesetz verlangt von E-Learning Systemen, dass die Sicherheit der gesammelten und verwendeten Daten ausreichend zu gewährleisten ist. Gesammelte Daten wie personenbezogene Informationen, Lernfortschritte, erreichte Zertifikate oder Lerninhalte sind durch das System sicher zu schützen. Besonders die Verarbeitung personenbezogener Daten muss anhand der rechtlichen Vorgaben des Datenschutzgesetzes erfolgen, um die Interessen der Benutzer von E-Learning Systemen zu schützen.

## 1.1. Motivation

Mit E-Learning eröffnet sich für die Weiterbildung die Chance, mehr Studenten und Schüler effektiver mit weniger Aufwand mit Lerninhalten zu fördern. Eine wichtige Voraussetzung für das Lernen mit solchen E-Learning Systemen ist jedoch, dass sich die Anwender bei der Benutzung sicher fühlen. Das Vertrauen in ein solches System kann nur über einen abgesicherten Webauftritt erreicht werden. Zum Beispiel würden Studenten oder Schüler nur Software zum Ler-

nen benutzen, in der eine angemessene Sicherheit für personenbezogene Daten gewährleistet werden kann.

Des Weiteren können böswillige Attacken Schaden bei dem Betreiber und die Benutzer des E-Learning Systems verursachen. Das kann durch externe Zugriffe auf die Datenbank, durch den versuchten Aufruf bestimmter Links oder schadhafem Code geschehen. Dadurch ist es möglich, die Daten der Benutzer einzusehen oder zu manipulieren und die Funktionsfähigkeit des E-Learning Systems einzuschränken. Hier liegt die Motivation darin, das System vor solchen Angriffen zu schützen und möglichen Schaden durch Dritte zu begrenzen oder gar zu vermeiden.

Folglich ist es für ein E-Learning System im Internet wichtig, dass die Sicherheit der Daten gewährleistet wird und der Webaufruf immer zuverlässig funktioniert. Dieser sollte für die Benutzer zum Lernen immer erreichbar sein und den allgemeinen Sicherheitsstandards entsprechen.

## 1.2. Ziel der Arbeit

Im Rahmen dieser Masterarbeit soll eine Konzeption und Implementierung geeigneter Datenschutz- und Datensicherheitsmaßnahmen in E-Learning Systemen erarbeitet werden, deren Ziel es ist, aufzuzeigen, welche datenschutzrechtlichen Aspekte durch ein solches System einzuhalten sind und welche Maßnahmen erforderlich sind, um den Schutz der Daten zu gewährleisten.

Es wird zunächst der Aufbau eines E-Learning Systems, das durch die Datenschutz- und Datensicherheitsmaßnahmen geschützt werden soll, näher erläutert. Hierzu werden dem Leser die Funktionsbereiche des eigentlichen Systems und die dazugehörigen technischen Bausteine beschrieben. Anhand der Grundlage der Komponenten werden geeignete Datenschutz- und Datensicherheitsmaßnahmen entwickelt, um die vorgeschriebenen Gesetze und rechtlichen Rahmenbedingungen aus der Perspektive des Bundesdatenschutzgesetzes einzuhalten und die Sicherheit von Daten zu gewährleisten.

Basierend auf den gewonnenen Erkenntnissen von Datenschutz und Datensicherheit werden die herausgearbeiteten Maßnahmen in die Lernkartenplattform THMcards aus der Sicht eines Front-End-Entwicklers implementiert. Die Plattform wird um Informationspflichten, die vom Datenschutz vorgegeben werden, und um ein Back-End zur Verwaltung von Lerninhalten, Benutzern sowie Zu-

griffsrechten erweitert. Darüber hinaus werden vom JavaScript-Framework Meteor vorgegebene Maßnahmen eingepflegt, um die Plattform in Bezug auf die Datensicherheit zu optimieren.

## 1.3. Gliederung

Im Anschluss an diese Einleitung werden dem Leser in Kapitel 2 alle Komponenten eines E-Learning Systems aufgezeigt, die bei der Ausarbeitung geeigneter Datenschutz- und Datensicherheitsmaßnahmen zu berücksichtigen sind. Dazu werden in Abschnitt 2.1 die Funktionsbereiche des eigentlichen Systems und in Abschnitt 2.2 die dazugehörigen technischen Bausteine beschrieben.

Kapitel 3 befasst sich mit dem Thema Datenschutz. Zunächst wird erläutert, was unter personenbezogenen Daten verstanden wird und wie diese in einem E-Learning System verwaltet werden. Anschließend sollen in Abschnitt 3.3 die Grundsätze des Datenschutzes, die allgemeine Grundregeln bei der Verwendung personenbezogener Daten definieren, näher gebracht werden. Des Weiteren werden dem Leser in Abschnitt 3.4 weitere datenschutzrechtliche Aspekte des Telemediengesetzes aufgezeigt, die bei der Entwicklung eines E-Learning Systems beachtet werden müssen.

Kapitel 4 zeigt, welche Haftung und Verantwortlichkeiten Online-Dienste bei eingestellten rechtswidrigen Inhalten übernehmen. In Abschnitt 4.1 werden zunächst die verschiedenen Internetdienstanbieter des Telemediengesetzes vorgestellt. Außerdem wird in Abschnitt 4.2 geklärt, auf welche Rechtsverletzungen als Betreiber eines E-Learning Systems zu achten sind.

Kapitel 5 gibt einen detaillierten Überblick zur Datensicherheit. Es werden Maßnahmen betrachtet, die zur Vermeidung beziehungsweise Verminderung von Störungen in der Datenverarbeitung beitragen sollen. Hierzu werden in Abschnitt 5.1 die Schutzziele der Datensicherheit betrachtet. Darüber hinaus sollen in Abschnitt 5.2 die technischen und organisatorischen Maßnahmen, die eingehalten werden müssen, um die Datensicherheit von E-Learning Systemen zu gewährleisten, beschrieben werden und in Abschnitt 5.3 gängige Angriffsszenarien auf eine Webanwendung aufgelistet werden, um die Problematik der Datensicherheit zu verdeutlichen.

In Kapitel 6 wird die Lernkartenplattform THMcards vorgestellt. Dazu wird in Abschnitt 6.2 auf die Re-Implementierung von THMcards eingegangen und in Abschnitt 6.3 die technischen Aspekte der Plattform aufgezeigt.

Anschließend werden in Kapitel 7 die in dieser Arbeit vorgestellten Datenschutz- und Datensicherheitsmaßnahmen in die Lernkartenplattform THMcards eingepflegt, um diese aus datenschutzrechtlicher Sicht und bezüglich der Sicherheit von Daten zu verbessern. Hierfür werden die Pflichten, die sich aus den datenschutzrechtlichen Bestimmungen ergeben, und Techniken, die in Bezug auf die Sicherheit bei der Entwicklung mit Meteor eingehalten werden müssen, die THMcards implementiert sind, beschrieben.

Zum Abschluss werden in Kapitel 8 die Erkenntnisse dieser Arbeit zusammengefasst. Ferner soll dem Leser zusätzlich ein Ausblick auf die auf mögliche Verbesserungen und Weiterentwicklung gegeben werden.

## **2. Analyse von E-Learning Webseiten**

Unter dem Begriff Electronic Learning - oder kurz E-Learning - versteht man im Allgemeinen das Lehren und Lernen mit Hilfe von digital gespeicherten Lerninhalten und softwareunterstützten Lernumgebungen [Wag05]. Dazu werden digitale Medien wie Computer oder auch das Internet eingesetzt. Das E-Learning System bildet hierbei die technische Basis einer E-Learning Infrastruktur, um die Steuerung von Lernprozessen oder auch die Bereitstellung verschiedener Kommunikationsformen zu ermöglichen.

Durch die Verwendung von E-Learning Systemen kommt es zu einer stärkeren Flexibilität bei der Wahl von Lernorten und Lernzeiten im Vergleich zum herkömmlichen Lernen. Lernende sind nun nicht mehr an einen bestimmten Ort gebunden, da der Zugang zu den Lernmaterialien mit dem Computer von fast allen Orten aus möglich ist und Lernzeiten sowie Lerndauer frei gewählt werden können. Diese verstärkte Individualisierung des Lernprozesses erlaubt, dass Lernangebote von Personen genutzt werden können, die üblicherweise keinen Zugriff darauf hätten.

Bei der Analyse von E-Learning Systemen soll nun geklärt werden, aus welchen Komponenten ein solches System besteht, die mit Datenschutz- und Datensicherheitsmaßnahmen geschützt werden müssen. Diese Komponenten setzen sich aus den Funktionsbereichen des eigentlichen Systems und den dazugehörigen technischen Bausteinen zusammen.

### **2.1. Funktionsbereiche eines E-Learning Systems**

E-Learning Systeme bieten verschiedene Funktionsbereiche an, um den gesamten Umfang von E-Learning nutzen zu können. Hierzu werden Administrationswerkzeuge zur Verwaltung der Benutzer und Inhalte, Evaluierungswerkzeuge zur Be-

urteilung von Lernenden, Autorenwerkzeuge zur Erstellung, Veröffentlichung und Verwaltung von Lerninhalten und Kommunikationswerkzeuge zur Verfügung gestellt. Die im Folgenden aufgeführten Funktionsbereiche sind in jedem E-Learning System unterschiedlich stark ausgeprägt.

### 2.1.1. Administrationswerkzeuge

Um die Administration von Benutzern und Inhalten zu vereinfachen, stehen unterschiedliche Werkzeuge zur Verfügung. Sie unterstützen die Vergabe von Rechten und Rollen (Administratoren, Dozenten oder Studenten). Die Rechte werden hierbei auf den Rollen basierend vergeben und definieren, welche Rollen Zugang zu welchen Inhalten und Bereichen in einem E-Learning System erhalten. [Kat07]

Des Weiteren werden alle Aktivitäten eines Benutzers in einem E-Learning System dokumentiert. Dazu gehören Informationen über die Anmeldung in das System (Zeitpunkt des Beginns und des Endes einer Sitzung, die IP-Adresse des Benutzers), der Aufruf einzelner Seiten des E-Learning Systems und ausgeführte Aktionen wie das Uploaden und Downloaden von Dateien. [Kat07]

### 2.1.2. Evaluierungswerkzeuge

Mit Werkzeugen für die Evaluation kann in E-Learning Systemen der Lernerfolg von Lernenden bewertet werden. Das Verfahren für die Beurteilung ermittelt hierbei die Fähigkeiten und den Wissensstand der Lernenden. Durch die Evaluierungswerkzeuge können Online-Fragebögen, Quizze und Tests erstellt, ausgewertet und verwaltet werden. Beispielsweise kann am Ende einer absolvierten Lernphase ein Test durchgeführt werden, der Auskünfte über den Lernerfolg der Lernenden oder Hinweise für den weiteren Lernprozess gibt. [Kat07]

Zur Prüfung der Lernleistung von Lernenden stehen den Lehrenden in einem E-Learning System mehrere Werkzeuge zur Verfügung. So kann beispielsweise eine Überprüfung des Abgabezeitpunktes erfolgen und auf eine Bewertungs- und Rückmeldemöglichkeit zurück gegriffen werden. [Kat07]

Ein Evaluierungswerkzeug von THMcards bildet die Export-Funktion der Lernstatistik eines Kartensatzes in der Lernphase. Die exportierte Datei bietet eine Übersicht über den Lernstand der aktiven Lernenden des Kartensatzes. So ist ersichtlich, welche Lernkarten sich in welchem Fach befinden oder gar vollständig gelernt wurden. Die Fächer werden hierbei nach dem Leiter Algorithmus gestaltet.

Des Weiteren bieten integrierte Charts in der Profilansicht oder Lernansicht eine Übersicht über den Lernstand eines Benutzers. Abbildung 2.1 zeigt den Lernstand eines Benutzers in der Profilansicht.

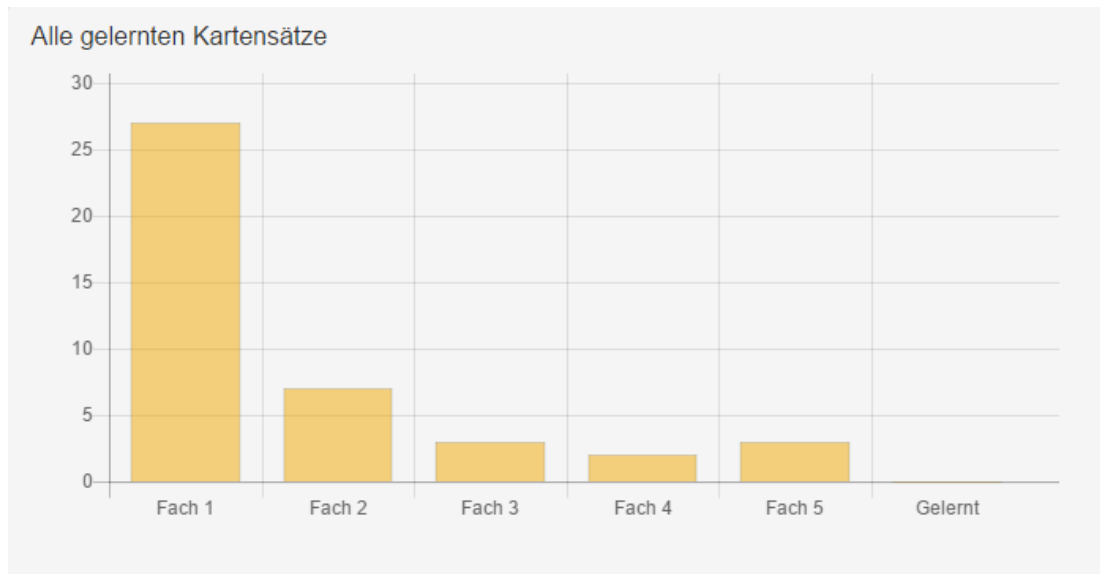


Abbildung 2.1.: Übersicht über alle gelernten Kartensätze in THMcards

### 2.1.3. Autorenwerkzeuge und Darstellung von Inhalten

Autorenwerkzeuge unterstützen den Benutzer bei der Erstellung von Lerninhalten und ermöglichen es, diese in einem E-Learning System in unterschiedlicher Form darzustellen. Die erstellten Lerninhalte werden für den Benutzer in einer klaren Struktur dargestellt, sodass eine effektive Nutzbarkeit gewährleistet wird. Hierbei unterstützen Autorenwerkzeuge den Benutzer, indem fertige Lösungen für die Medieneinbindung und Interaktionsgestaltung angeboten werden, ohne dass in traditioneller Weise programmiert werden muss. [Uni]

Autorenwerkzeuge können dem Benutzer die Erstellung und Gestaltung von Lerninhalten auf unterschiedlichste Art und Weise vereinfachen [Uni]:

- Lehr- und Lerninhalte ohne Programmierkenntnisse erstellen.
- Darstellung von Lerninhalten vereinfachen und vereinheitlichen.
- Unterschiedliche Dateiformate (z.B. Word, PDF, HTML), Medientypen (Grafik, Audio, Video) und interaktive Elemente (z.B. Flash) einbinden.

- Vorlagen erstellen als Grundlage für neue Lerninhalte oder für die spätere Wiederverwendung.
- Lerninhalte importieren/exportieren.

### 2.1.4. Kommunikations- und Kooperationswerkzeuge

Die Kommunikationswerkzeuge stellen einen wichtigen Funktionsbereich eines E-Learning Systems dar. Durch die Kommunikation mit Lehrenden oder anderen Lernenden wird der Lernprozess des Benutzers direkt beeinflusst und stellt somit eine Voraussetzung für die gemeinsame Erarbeitung von Lerninhalten in einer virtuellen Lernumgebung dar [Kat07]. Man unterscheidet bei den Kommunikationswerkzeugen zwischen zwei Klassen [Kat07]:

- Werkzeuge für die *synchrone Kommunikation*: Chat, Videokonferenz.
- Werkzeuge für die *asynchrone Kommunikation*: Diskussionsforen, Dateiaustausch, E-Mails, Benachrichtigungssysteme.

Während bei der synchrone Kommunikation die Kommunikation zeitgleich erfolgt, nehmen die Benutzer bei der asynchronen Kommunikation zeitlich versetzt miteinander Kontakt auf. [Wikd]

### 2.1.5. Weitere Werkzeuge

In einem E-Learning System gibt es neben den bereits aufgezählten Funktionsbereichen noch weitere Werkzeuge wie [Kat07]:

- Eine Suchfunktion nach Lerninhalten oder Benutzern.
- Eine Kalenderfunktion mit Benachrichtigungen.
- Lesezeichen für die Erleichterung der Navigation in der Lernplattform.
- Annotationen an Lerninhalten anbringen und Offlinebearbeitung.

Diese Werkzeuge stehen allen Benutzern des E-Learning Systems (Administratoren, Lehrenden, Lernenden) zur Verfügung.

## 2.2. Technische Bausteine eines E-Learning Systems

Um eine E-Learning Webseite zu realisieren sind eine Reihe von technischen Bausteinen erforderlich. Eine Webseite kann nur dann technisch und sicher funktionieren, wenn diese konsequent beachtet werden. Zu den technischen Bausteinen, die aktiv während der Planung und des Designs der Webseite berücksichtigt werden sollten, gehören: Die Internet-Anbindung, aktive Inhalte, das Datenbanksystem und integrierte Zahlungssysteme.

### 2.2.1. Internet-Anbindung

Wird ein E-Learning System über das Internet angeboten, stellt sich die Frage, welche Variante des Hosting verwendet werden soll. Mit Hosting wird die Bereitstellung eines speziellen Speicherplatzes im Web beschrieben. Die E-Learning Webseite wird dadurch im Internet verfügbar gemacht. Man unterscheidet hierbei zwischen zwei Varianten: *Inhouse-Hosting* und *Webhosting*.

Beim Inhouse-Hosting findet das Hosting im eigenen "Haus" statt. Die Daten der Webseite werden auf einem eigenen Server abgelegt und eigenständig verwaltet. Dadurch kann unmittelbar auf die Daten zugegriffen werden und die Kontrolle über diese bleibt vollständig erhalten. Da die Verwaltung eines eigenen Servers die Einstellung eigener Mitarbeiter nach sich zieht, sind die Kosten dieser Variante entsprechend hoch. Bei einer professionellen Lösung kann mit mehreren Tausend Euro pro Monat gerechnet werden. [Gra11]

Beim Webhosting wird der Speicherplatz für die Webseite von einem externen Internetdienstanbieter zur Verfügung gestellt. Dadurch kann vom KnowHow, der Flexibilität und den Sicherheitstechniken des Anbieters profitiert werden. Darüber hinaus können Kosten eingespart werden, da lediglich für den Support gezahlt werden muss. Jedoch geht mit dieser Lösung die direkte Kontrolle über die Daten verloren. [Sch] [Ser]

### 2.2.2. Aktive Inhalte und dynamische Webseiten

Bei der Realisierung einer E-Learning Webseite geht es auch darum festzulegen, wie die dargestellten Inhalte technisch umgesetzt werden sollen. Grundsätzlich wird zwischen aktiven Inhalten und dynamischen Webseiten unterschieden. Wäh-

rend aktive Inhalte ein Risiko für den Benutzer darstellen, bringen dynamische Webseiten Gefährdungen für den Betreiber mit sich.

Mit aktiven Inhalten wird der statische Inhalt von Webseiten um Funktionalitäten erweitert, die mit reinem HTML nicht möglich sind. So kann mit einfachen Mitteln ein Maß an Flexibilität und Dynamik auf der Webseite für den Benutzer hergestellt werden, indem sie zur Laufzeit direkt auf dem Rechner des Benutzer ausgeführt werden. Bei aktiven Inhalten entsteht für den Benutzer ein hohes Risiko, da die Ausführung unbemerkt im Hintergrund der Webseite erfolgt. So kann die fehlerhafte Programmierung von aktiven Inhalten Kontrollfunktionen des Webbrowsers aufheben oder eine böswillige Programmierung Zugriff auf den Rechner des Benutzer herstellen, um Daten zu lesen, zu verändern oder zu speichern. Zu den aktive Inhalte, die am häufigsten verwendet werden, gehören JavaScript, Java, VBScript und ActiveX. [Bsia] [Bsib]

Dynamische Webseiten bauen auf der Technik von statischen Webseiten auf. Während bei statischen Webseiten alle Seiten einer Webseite als separate Dateien entwickelt und auf dem Server gespeichert werden, werden dynamische Webseiten erst beim Aufruf der Webseite erzeugt [Cre]. Dynamische Webseiten arbeiten auf Basis von Datenbanken [Cre]. Inhalte werden getrennt vom Layout aufbewahrt, beim Aufruf der Webseite aus der Datenbank gelesen und zu einer Webseite zusammengestellt [Cre]. So finden dynamische Webseiten Anwendung, wenn personalisiert Inhalte dargestellt werden sollen (z.B. individuelle Vorschläge in einem Online-Shop, abhängig von bisher erworbenen Artikeln) oder aktuelle Informationen für den Benutzer enthalten sind (z.B. Nachrichtenticker). Dynamische Webseiten befinden sich auf dem Server des Betreibers. Ist einem Angreifer eine Sicherheitslücke in dem System bekannt, kann dieser durch böswillige Eingabedaten oder Codezeilen dem Server oder den Daten des Betreibers Schaden zufügen.

### 2.2.3. Datenbanksysteme

Wird in E-Learning Systemen eine größere Menge an Daten, wie erstellte Lerninhalte oder Benutzer des Systems, verwaltet, ist der Einsatz einer Datenbank - oder auch Datenbanksystem (DBS) - unvermeidbar. Mithilfe von Datenbanken lässt sich eine große Sammlung von Daten strukturiert speichern und in logischen Zusammenhängen für den Benutzer digital darstellen [Datb] [Datc].

Eine Datenbank besteht aus zwei Komponenten: Der Software zur Verwaltung der Daten, Datenbankmanagementsystem (DBMS) genannt, und aus den zu verwaltenden Daten, der eigentlichen Datenbank (DB) oder auch Datenbasis [Wika]. Das Datenbankmanagementsystem organisiert die strukturierte Speicherung der Daten und bietet effiziente Zugriffsverfahren (lesende und schreibende Zugriffe), um diese zu verarbeiten [Datc]. Als Schnittstelle zur Anwendersoftware wird für das DBMS eine Datenbanksprache, wie beispielsweise SQL, benötigt (siehe Abbildung 2.2). Dadurch lassen sich Abfragen wie das Einfügen und Ändern von Daten sowie administrative Befehle von Benutzern ausführen [Datc]. Bei der Bearbeitung der Abfragen stellt das DBMS sicher, dass die Integrität (dauerhafte Verfügbarkeit und Korrektheit der Daten) und die Sicherheit (Zugriff nur durch autorisierte Benutzer) der Daten gewährleistet wird [Rou14].

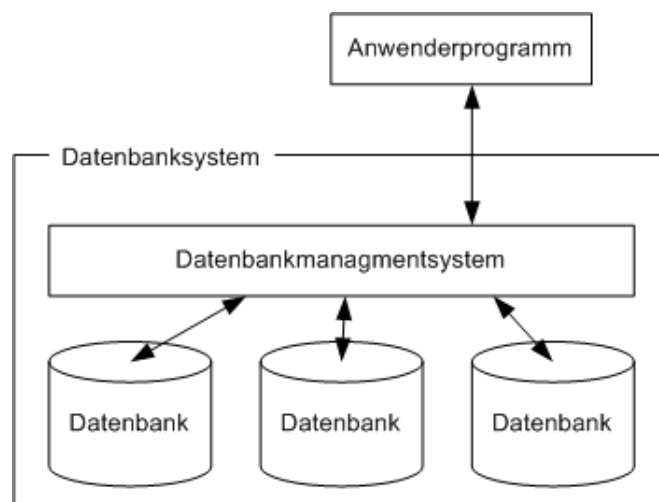


Abbildung 2.2.: Aufbau eines Datenbanksystems [Data]

In E-Learning Systemen wie THMcards wird eine Datenbank sowohl zur Speicherung als auch zur Verwaltung der erstellten Lerninhalte und Benutzer verwendet. Die erstellten Lerninhalte werden in der Datenbank gespeichert und bilden die Basis des Lernangebots, das dem Benutzer im E-Learning System dargestellt wird. Bei einem großen Bestand an Lerninhalten sollte für den Benutzer eine Suchfunktion integriert sein, um eine effektive Recherche zu ermöglichen. Da THMcards Lerninhalte zahlungspflichtig anbieten, sollte eine umfangreiche Beschreibung des Lerninhalts gegeben sein, um eine Kaufentscheidung zu unterstützen.

### 2.2.4. Zahlungssysteme

Elektronische Zahlungssysteme sind eine bequeme Lösung, um Benutzern eines E-Learning Systems den Erwerb zahlungspflichtiger Lerninhalte zu ermöglichen. Hierzu wird im System eine Schnittstelle zu einem Serviceprovider für elektronische Zahlungssysteme, wie Lastschriftverfahren, Kreditkartenzahlung oder PayPal, hergestellt. Diese Lösung wird am häufigsten eingesetzt, da sich die Einbindung mehrerer Zahlungsverfahren durch die Verwendung externer Zahlungssysteme im Vergleich zu Eigenlösung deutlich einfacher gestaltet. Ob Lerninhalte tatsächlich verkauft werden, hängt davon ab, ob die eingesetzten Zahlungssysteme benutzerfreundlich, sicher und zuverlässig sind [Webc]. Vor allem im Bereich Sicherheit sollte das jeweilige Zahlungssystem gewährleisten, dass sensible Daten des Benutzers vor Diebstahl und Betrug geschützt werden.

Die rechtlichen Rahmenbedingung und Datensicherheitsmaßnahmen, um ein E-Learning System bei finanziellen Transaktionen abzusichern, sind aufgrund des rechtlichen Umfangs nicht Bestandteil dieser Arbeit und werden nicht weiter behandelt. Mehr Informationen zu dem Thema sind in der Masterarbeit *Erfolgreiche Monetarisierung einer Open-Source Webanwendung durch die Integration eines innovativen Geschäftsmodells* von Marius Trautrim zu finden [Tra16].

### 3. Datenschutz

Bei der Verwendung von E-Learning Systemen werden Zugangsdaten und Aufzeichnungen über einzelne Aktivitäten von Benutzern des Systems gespeichert und für die Auswertung verwendet. Diese ausgewerteten Daten können genutzt werden, um beispielsweise die Beteiligung der Studenten an einer Lehrveranstaltung zu überprüfen oder Dozenten einer Veranstaltung zu bewerten. Jedoch entsteht durch die Menge der gesammelten Daten und deren Verwendungsmöglichkeiten das Risiko, die Privatsphäre und den Schutz personenbezogener Daten der Benutzer solcher E-Learning Systeme zu verletzen.

Der Begriff Datenschutz beschäftigt sich mit dem Thema, dem Missbrauch personenbezogener Daten entgegen zu wirken [Bsi13b]. Im Bundesdatenschutzgesetz (BDSG) wird im § 1 Absatz 1 BDSG der Begriff Datenschutz wie folgt definiert:

„Zweck dieses Gesetzes ist es, den Einzelnen davor zu schützen, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Persönlichkeitsrecht beeinträchtigt wird.“ [BDSa]

Beim Datenschutz wird also gewährleistet, dass jede Person das Recht hat, über Verwendung und Preisgabe personenbezogener Daten selbst zu bestimmen [Bsi13b].

Im deutschen Recht wird zur allgemeinen Bestimmung des Datenschutzrechts unterschieden, ob die Datenverarbeitung im öffentlichen oder privaten Bereich stattfindet. Zu den allgemeinen Datenschutzgesetzen gehören das Bundesdatenschutzgesetz (BDSG) und das jeweilige Landesdatenschutzgesetz (LDGS) der einzelnen Bundesländer. Das LDGS regelt, unter welchen Voraussetzungen die Behörden und sonstige öffentliche Stellen des jeweiligen Bundeslandes personenbezogene Daten verarbeitet werden dürfen. Die Datenschutzerfordernungen, die nicht-öffentliche Stellen wie Banken, Wirtschaftsunternehmen oder Rechtsanwälte zu beachten haben, regelt das BDSG. [LDI]

Öffentliche Hochschulen werden ebenfalls als Stelle des öffentlichen Bereichs des Datenschutzes zugeordnet [Zen]. Da es sich bei THMcards um ein Projekt der

Technischen Hochschule Mittelhessen handelt, kommt aufgrund dieser Regelung das Hessische Datenschutzgesetz vorrangig zum Einsatz.

Im weiteren Verlauf dieses Kapitels wird zunächst erläutert, welche Informationen mit personenbezogenen Daten beschrieben werden. Danach werden Möglichkeiten betrachtet, wie personenbezogene Daten mithilfe des Identitätsmanagements verwaltet werden können. Zum Schluss wird ein Überblick über die Grundprinzipien des Datenschutzes gegeben und es werden weitere wichtige datenschutzrechtliche Aspekte näher erläutert, die in einem E-Learning System eingehalten werden müssen.

## 3.1. Personenbezogene Daten

Der Personenbezug ist der bedeutendste Aspekt des Datenschutzes. Fast alle Datenschutzgesetze befassen sich ausschließlich mit personenbezogenen Daten. Es muss nun deutlich gemacht werden, welche Daten eigentlich personenbezogen sind und welche nicht. Gemäß § 3 Absatz 1 BDSG lautet die Definition personenbezogener Daten wie folgt:

„Personenbezogene Daten sind Einzelangaben über persönliche oder sachliche Verhältnisse einer bestimmten oder bestimmbaren natürlichen Person (Betroffener).“ [BDSb]

Es handelt sich also beim Datenschutz nicht um den Schutz von Unternehmensgeheimnissen, sondern um den Schutz von Angaben, aus denen man eine bestimmte Person erkennen kann oder die einer bestimmten Person zugeordnet werden können [Dat14]. Darüber hinaus bezieht sich das Gesetz nur auf Daten natürlicher Personen. Die Daten juristischer Personen, die per Gesetz geschaffen werden und volle Rechtsfähigkeit erlangen, wie GmbH, Kapitalgesellschaften oder Vereine, werden nicht durch das Datenschutzgesetz geschützt [DPN]. Eine Ausnahme besteht nur, wenn die Angaben zu einer juristischen Person keine Rückschlüsse auf eine natürliche Person zulässt [DPN]. Dies kann der Fall sein, wenn bei einer GmbH einer Einzelperson enge finanzielle, persönliche oder wirtschaftliche Verflechtungen zwischen der natürlichen und der juristischen Person bestehen [Pro].

Bei *Einzelangaben* handelt es sich um Informationen, um eine einzelne Person identifizieren zu können. Die *persönlichen und sachlichen Verhältnisse* decken alles ab, was sich auf eine Person bezieht: [Dat15b]

- Name, Geburtsdatum, Alter, Familienstand.
- Anschrift, Telefonnummer, E-Mail Adresse.
- Kontonummer, Kreditkartennummer.
- Augenfarbe, Schuhgröße.

Ganz besonders sind *besondere Arten* personenbezogener Daten (§ 3 Absatz 9 BDSG) zu schützen. Diese enthalten Informationen über [DPN]:

- die ethnische Herkunft.
- die politische Meinung.
- die religiöse oder philosophische Überzeugung.
- eine Gewerkschaftszugehörigkeit.
- die Gesundheit.
- das Sexualleben.

*Bestimmt* ist eine Person dann, wenn sie direkt namentlich genannt wird oder wenn sich aus dem Inhalt oder dem Zusammenhang der Daten ein Bezug auf die Person unmittelbar erschließen lässt. *Bestimmbar* ist eine Person dann, wenn eine Person mittels Zusatzwissen (z.B. öffentlich zugängliche Quellen, wie Telefonbücher oder Handelsregister) festgestellt werden kann. [Pro] [Dat15b]

## 3.2. Identitätsinfrastruktur

Das Identitätsmanagement befasst sich mit der Verwaltung der personenbezogenen Daten, die entstehen, wenn sich Benutzer in ein E-Learning System authentifizieren [Kat07]. Bei dem Vorgang ist eine Abbildung der gesamten Identität einer realen Person in die digitale Welt nicht möglich [Kat07]. Es können wichtige Teile einer Identität, wie Name, E-Mail Adresse oder Telefonnummer in der digitalen Welt abgelegt werden, um dadurch verschiedene digitale Identitäten für unterschiedliche Anwendungen anzulegen. Man unterscheidet bei der Identität einer Person zwischen verschiedenen Teilaspekten: *Anonymität*, *Pseudoidentität* und *persönliche Identität*.

Bei der *Anonymisierung* wird nach § 3 Absatz 6 BDSG auf alle personenbezogene Daten verzichtet, sodass die erhobenen Einzelangaben zu einer Person nicht mehr oder nur mit einem erheblichen Aufwand an Zeit, Kosten und Arbeitskraft zugeordnet werden können. Die personenbezogenen Daten werden so reduziert, dass kein Personenbezug mehr besteht und damit die Daten in voller Freiheit genutzt werden können. [Dat15a]

Gemäß § 3 Absatz 6a BDSG liegt eine *Pseudonymisierung* vor, wenn personenbezogene Angaben wie Name oder ein anderes Identifikationsmerkmale durch ein Kennzeichen ersetzt und weitere Hinweise auf die individuelle Person entfernt werden. Damit wird die eindeutige Bestimmung einer Person ausgeschlossen oder erschwert. [JH]

Die *persönliche Identität* setzt sich aus allen Daten einer realen Identität zusammen, die durch das Datenschutzgesetz abgedeckt werden. Diese digitale Identität wird aus dem Namen und der E-Mail Adresse gebildet. Sie repräsentiert die Abbildung der Person in elektronischen Medien, die nur zum Teil wiederherstellbar und damit nicht eindeutig ist. Jede digitale Identität ist durch eine Menge von Identitätsattributen beschrieben. [Kat07]

In THMcards läuft die Authentifizierung in das System über die Angabe des Benutzernamens und Passworts ab. Zusätzlich werden, abhängig von der Wahl des Sozialen Netzwerks, die E-Mail Adresse sowie weitere spezifische Attribute erfasst.

#### 3.2.1. Identitätsmanagement

Das Identitätsmanagement eines E-Learning Systems wird von der Benutzerverwaltung übernommen. Darüber hinaus wird über die Benutzerverwaltung die Zuteilung der Rollen verwaltet, um Rechte und Ressourcen an die individuellen Rollen des Systems zu vergeben. [Kat07]

Mithilfe eines Identitätsmanagers können die Benutzer selbstständig entscheiden, welche Informationen über das eigene Profil herausgegeben werden. Die Erstellung eines solchen Profils spielt in E-Learning Systemen wie THMcards eine entscheidende Rolle, da über die digitale Identität Lerninhalte eingestellt, bewertet und Lernphasen abgeschlossen werden können.

Abbildung 3.1 zeigt einen Teil der Benutzerprofilansicht des E-Learning Systems THMcards. Der Benutzer ist in der Lage, seinen Vor- und Nachnamen sowie eine E-Mail Adresse im System zu hinterlegen. Darüber hinaus kann der Benut-

Profil

Profil öffentlich ☒ Ja ☐ Nein

E-Mail matthias.zimny@mni.thm.de

Kennung Matthias Zimny

Titel Akademischer Titel (öffentlich)

Nachname Zimny

Vorname Matthias

Abbrechen Speichern

Abbildung 3.1.: Identitätsmanagement in THMcards

zer darüber entscheiden, ob sein Profil für andere Benutzer des Systems öffentlich zugänglich ist oder nicht. Die Information über den akademischen Titel wird von den Administratoren des Systems in der Benutzerverwaltung des Back-Ends ergänzt.

### 3.3. Datenschutzgrundsätze

Im Datenschutzgesetz werden Grundsätze für den Datenschutz definiert. Dabei handelt es sich um allgemeine Grundregeln, die eingehalten werden müssen, wenn personenbezogene Daten verwendet werden sollen. Mit diesen Grundregeln wird das gesamte Datenschutzrecht aufgespannt. Diese Regeln sind maßgeblich bei der Nutzung personenbezogener Daten, sodass auf keine verzichtet werden kann. Die folgenden Kapitel zeigen, wie die Grundsätze allgemein aufgeteilt werden können.

#### 3.3.1. Verbotsvermutung und Erlaubnisvorbehalt

Der Grundsatz der Verbotsvermutung und des Erlaubnisvorbehalts beschreibt, dass das Erheben, Verarbeiten und Nutzen von personenbezogenen Daten grundsätzlich verboten ist. Ausnahmen bestehen nur dann, wenn die Verarbeitung personenbezogener Daten gesetzlich erlaubt oder angeordnet ist oder eine Einwilligung des Betroffenen vorliegt. [Dat14]

Die Datenerhebung, also das Beschaffen von Daten, ist nur unmittelbar beim Betroffenen zulässig. Die Beschaffung von Daten ist also nur unter Mitwirkung des Betroffenen erlaubt. Die Einwilligung muss dabei auf einer freien Entscheidung des Betroffenen beruhen. Das bedeutet, dass die Person bei der Einwilligung nicht unter Druck gesetzt oder zur Einwilligung gezwungen wird und alle nötigen Einsichten (Grundsatz der Transparenz) für die Situation gegeben sind. Die Person ist auf die Freiwilligkeit seiner Angaben hinzuweisen und gegebenenfalls über die Folgen der Verweigerung von Angaben zu informieren. [Dat14] [Kei]

Die Einwilligung erfolgt grundsätzlich schriftlich, in der Regel durch eine Unterschrift (§ 4a Absatz 1 S. 3 BDSG). Um einen Medienbruch zu vermeiden, ist jedoch auch die Einwilligung durch die elektronische Form nach § 13 Absatz 2 Telemediengesetz (TMG) möglich. [Datd]

Besondere Vorsicht gilt bei der Verarbeitung besonderer Arten personenbezogener Daten. Die Einwilligung muss sich nach § 3 Absatz 9 BDSG ausdrücklich auf diese Daten beziehen. [Gag15]

#### **3.3.2. Datenvermeidbarkeit und Datensparsamkeit**

Im Grundsatz der Datenvermeidbarkeit und Datensparsamkeit wird beschrieben, keine oder so wenig personenbezogene Daten wie möglich zu erheben, zu verarbeiten und zu nutzen [Kei]. Nur wenn eine Nutzung personenbezogener Daten nicht vermieden werden kann, sollen diese in einem kleinen Umfang verwendet werden. Darüber hinaus dürfen nur die Daten genutzt werden, die auch tatsächlich für die Verarbeitung notwendig sind [Dat14].

Hierbei stellen die Anonymisierung und Pseudonymisierung von personenbezogenen Daten eine geeignete Methode dar, personenbezogene Daten zu vermeiden oder sparsam zu verwenden [Bsi13b]. Anonymisierung wird grundsätzlich nur für Anwendungsfälle eingesetzt, wenn Daten verarbeitet werden müssen, aber die Identität der Person nicht von Bedeutung ist [Kei]. Beispielsweise spielt bei der Einsicht einer öffentlich zugänglichen Nachrichtenseite die Identität einer Person für den Anbieter der Seite keine Rolle. Pseudonymisierung wird eingesetzt, wenn allgemein eine anonyme Datenverarbeitung möglich ist, jedoch die Daten auf die Person gegebenenfalls zurückführen möchte [Kei]. Beispielsweise ist ein pseudonymer Einkauf im Internet möglich, indem die Daten des Käufers so aufgeteilt werden, dass der Verkäufer nur die Kaufdaten und die Transaktionsnummer er-

hält, das Logistikunternehmen nur den Namen, Lieferanschrift sowie Transaktionsnummer des Käufers.

### 3.3.3. Zweckbindung

Bei der Erhebung, Verarbeitung und Nutzung personenbezogener Daten unterliegen öffentliche sowie nicht-öffentliche Stellen gemäß §§ 14, 28, 29 BDSG dem Grundsatz der Zweckbindung. Das bedeutet, dass bei jeder Verarbeitung von personenbezogenen Daten ein bestimmter Zweck zugrunde liegen muss. Dieser Zweck muss vor der Verarbeitung der Daten festgelegt und dokumentiert worden sein. Eine Weiterverarbeitung der personenbezogenen Daten ist nur auf die in der Zweckbindung vereinbarten Weise erlaubt. [Gag15] [Dat14]

Die Verarbeitung und Speicherung der personenbezogenen Daten wird auf einen Zeitraum begrenzt, bis der vorher dafür definierte Zweck erreicht worden ist. Eine Vorratsspeicherung personenbezogener Daten für noch nicht festgelegte Zwecke ist somit unzulässig. Sollten diese Daten für einen weiteren Zweck genutzt werden, muss eine erneute Einwilligung des Betroffenen eingeholt werden. Jedoch gibt es Ausnahmen, um personenbezogene Daten für andere Zwecke verwenden zu dürfen: [Gag15] [Kei]

- Die Wahrung berechtigter Interessen der verantwortlichen Stelle stehen im Vordergrund.
- Die Daten allgemein zugänglich sind oder veröffentlicht werden dürfen.
- Wissenschaftliche Zwecke damit verknüpft werden.
- Zum Zwecke der Werbung oder der Markt- oder Meinungsforschung bei listenmäßiger Übermittlung.

### 3.3.4. Transparenz

Dieser Grundsatz verpflichtet die Verwender zur Transparenz von personenbezogenen Daten. Die Betroffenen haben das Recht zu wissen auf welche Art, zu welchem Umfang und zu welcher Dauer die Daten gespeichert werden. Darüber hinaus besteht für die Verwender der personenbezogenen Daten die Pflicht, den Betroffenen diese Informationen zugänglich zu machen. Der Betroffene kann selbst überprüfen, ob das Bild, das durch die Datenverarbeitung entworfen wurde, auch

dem entspricht, was er selbst preisgeben will. Somit wird ihm das Recht gewährt, über personenbezogene Daten selbst zu bestimmen. [Dat14] [Kei]

Es ergeben sich beim Betroffenen folgende Rechte [Bsi13b]:

- Auskunft und Einsichtnahme über gespeicherte Daten zur Person.
- Berichtigung, Sperrung und Löschung personenbezogener Daten.
- Widerspruch aus besonderem Grund, sofern die Datenverarbeitung nicht durch eine Rechtsvorschrift verlangt wird.
- Recht auf Schadensersatz bei einer unzulässigen oder unrichtigen Verarbeitung personenbezogener Daten.

#### 3.3.5. Datensicherheit

Die erhobenen personenbezogene Daten sind durch die von den Daten verarbeitenden Stellen durch technische und organisatorische Maßnahmen ausreichend zu sichern. Hierzu werden im Bundesdatenschutzgesetz konkrete Schutzmaßnahmen gemäß § 9 BDGS beschrieben, die bei der Erhebung, Verarbeitung und Nutzung personenbezogener Daten einzuhalten sind. Vor allem sind die Daten vor Verlust, unerlaubter Kenntnisnahme oder Verfälschung zu schützen. [Gag15] [Bsi13b]

In welchem Maße Vorkehrungen getroffen werden müssen, um die Datensicherheit eines E-Learning Systems zu gewährleisten, wird in Kapitel 5 *Datensicherheit* aufgeführt.

## 3.4. Weitere datenschutzrechtliche Aspekte

Neben den allgemeinen Datenschutzgesetzen gibt es noch weitere Datenschutzgesetze, die für spezifische Bereiche die Datenschutzbestimmungen definieren. Ist ein Sachverhalt eines bestimmten Bereiches, für das es ein eigenes Datenschutzgesetz gibt, nicht durch dieses spezifische Gesetz geregelt, gilt als letzte Regelung immer das BDSG. Sollte dagegen ein Sachverhalt im BDSG als auch von einem spezifischen Datenschutzgesetz geregelt sein, gilt immer die Regelung des spezifischen Datenschutzgesetzes. Aus diesem Grund wird das BDSG auch als Auffanggesetz bezeichnet. [Buc03]

Für die Regelung rechtlicher Rahmenbedingungen so genannter Telemedien kommt in Deutschland das Telemediengesetz (TMG) zum Einsatz. Bei Telemedien handelt es sich um einen aus *Telemedien* und *Mediendiensten* gebildeten Oberbegriff für elektronische Informations- und Kommunikationsdienste. Dazu gehört eine Vielzahl von Angeboten aus dem Internet wie Webshops, Chatrooms und Webportale, aber auch private Webseiten und Blogs. Umgangssprachlich wird es auch als *Internetgesetz* bezeichnet. Das Telemediengesetz löst das Teledienstegesetz (TDG), das Teledienstedatenschutzgesetz (TDDSG) sowie weitestgehend den Mediendienste-Staatsvertrag (MDStV) ab. Im Telemediengesetz werden folgende Vorschriften geregelt: [Jur] [Bre]

- Impressumspflicht für Telemediendienste.
- Verhindern von Spam (durch Vorgaben für den Inhalt und Absender von Werbe-E-Mails).
- Haftung von Internetdiensteanbietern für gesetzeswidrige Inhalte in Telemediendiensten.
- Datenschutz beim Betrieb von Telemediendiensten und bei der Herausgabe von Daten.

Nachfolgend sollen die Regelungen der Impressumspflicht von Telemediendiensten und die Cookie-Richtlinien im Rahmen der Datenschutzregelung von der Herausgabe von Daten genauer betrachtet werden.

#### 3.4.1. Impressumspflicht

Jede öffentliche Webseite ist nach dem TMG verpflichtet, bestimmte Informationen über sich bekannt zu geben. Durch die Pflicht ein Impressum zu führen, werden die Benutzer der Webseite über den jeweiligen Betreiber informiert, um rechtliche Ansprüche gegen diesen erheben zu können [Sie]. Zur Vermeidung kostspieliger Abmahnungen und Bußgelder in Höhe von bis zu 50.000 Euro (§ 16 Absatz 2 Nr. 1, Absatz 3 TMG), empfiehlt sich die Einhaltung der Impressumspflicht [Wie]. Die zuvor benannte Pflicht bezieht sich hierbei nicht nur auf den Internetdiensteanbieter, sondern auch auf die Eigentümer der Webseite (Landgericht Berlin vom 17.09.2002, Az.: 103 O 102/02) [Wie].

Gegen Entgelt angebotene Telemedien unterliegen nach § 5 des TMG der Impressumspflicht. Dies betrifft also Internetdiensteanbieter, die durch Inhalte,

Waren (Online-Shops) oder Dienstleistungen (Web-Hoster, Softwarevermietung) mit ihrer Webseite Geld verdienen. Bedeutsam ist ebenfalls § 55 Absatz 2 des Rundfunkstaatsvertrages (RstV), welcher die Impressumspflicht journalistisch-redaktionell gestalteter Webseiten vorschreibt. [Ott] [Sie]

Von der Impressumspflicht ausgenommen sind Webseiten, die ausschließlich privaten Zwecken dienen, also Webseiten, die sich nur auf persönliche und familiäre Inhalte beschränken. Hier gibt es jedoch zwei Punkte, die zu beachten sind. Zum Einen ist die Regelung, ob es sich bei einer Webseite um ein gegen Entgelt angebotenes Telemedium handelt, sehr streng. Die Platzierung eines Werbeanbanners oder die Teilnahme an einem Partnerprogramm kann dazu führen, dass eine Webseite nicht mehr als privat eingestuft wird. Solche Webseiten sollten also ein Impressum beinhalten, auch wenn mit der Werbung keine oder minimale Umsätze generiert werden. Zum Anderen ist die Frage, wann es sich bei einem Angebot im Internet um eine journalistisch-redaktionelle gestaltete Webseite handelt, im Gesetz nicht oder nicht ausführlich definiert. So bleibt beispielsweise die Rechtslage zahlreicher Internet-Tagebücher, so genannten Blogs, sowie Foren unklar. Deshalb sollten diese zur Absicherung ein Impressum beinhalten. [Ott] [Sie]

Das in § 5 Absatz 1 TMG geforderte Impressum ist so zu gestalten, dass es leicht erkennbar, unmittelbar erreichbar und ständig verfügbar ist [Sie]. Die Angaben sollten deshalb in einem eigenen Menüpunkt der Navigation eingebunden und von jeder Unterseite aus erreichbar sein. Das Impressum ist mit der Bezeichnung *Impressum* oder *Kontakt* zu kennzeichnen.

Welche allgemeinen Informationspflichten für das Impressum einer Webseite festgelegt werden, können im § 5 TMG oder unter dem Link [https://www.gesetze-im-internet.de/tmg/\\_5.html](https://www.gesetze-im-internet.de/tmg/_5.html) nachgelesen werden.

#### 3.4.2. Cookie-Richtlinien

Bei Cookies handelt es sich um kleine Datenmengen, die vom Internetdienstanbieter einer Webseite auf dem Rechner des Benutzers als Textdatei gespeichert werden. Sie werden typischerweise eingesetzt, damit Webserver die Möglichkeit haben, sich auf die Bedürfnisse des Benutzers einzustellen bzw. das Angebot des gewählten Webserver auf den Benutzer abzustimmen. Dadurch ist beispielsweise ein wiederholter Zugriff eines Benutzers auf eine Webseite erkennbar, sodass dieser sich bei einem erneuten Besuch eines verschlüsselten Bereiches nicht erneut anmelden muss. Leider ist es mit Cookies auch möglich, ein Benutzerprofil zu

erstellen, das Auskunft über komplexes privates Internetverhalten gibt und dieses an einen Empfänger übermittelt. Anders als beim Trojanischen Pferd ist ein Cookie jedoch nicht versteckt und kann vom Benutzer eingesehen und gelöscht werden. [Bäu00, Seite 13] [Kla02, Seite 111-112]

Wie mit Cookies rechtlich umgegangen werden muss, regelt in der EU die Richtlinie 2009/136/EG über die Verarbeitung personenbezogener Daten und den Schutz der Privatsphäre in der elektronischen Kommunikation, auch *Cookie-Richtlinie* oder *E-Privacy-Richtlinie* genannt. Diese sieht bei der Verwendung von Cookies im Internet eine ausdrückliche Einwilligung des Benutzers, ein so genanntes Opt-In-Verfahren, vor. Allerdings blieb Deutschland in Folge des Erlasses der Richtlinie auch über die Umsetzungsfrist, die 2011 endete, hinaus untätig, da die Bundesregierung die Ansicht vertritt, dass die bereits geltenden Regelungen des TMG dafür ausreichend seien. Demnach umfasst § 13 Absatz 1 TMG bereits den Einsatz von Cookies und die ausdrückliche Einwilligung in der Verwendung sei in § 12 Absatz 1 TMG geregelt. [Ros15]

Im Hinblick auf die Regelungen des TMG ist bei der datenschutzrechtlichen Einordnung von Cookies zu unterscheiden, ob mit diesen personenbezogene Daten verwendet werden oder nicht. Bei Cookies ohne Personenbezug findet das Datenschutzrecht keine Anwendung, sodass diese vom Internetdienstanbieter ohne weitere Vorkehrungen verwendet werden können [Hoe16, Seite 456]. Hierzu gehören beispielsweise Cookies, die zur automatischen Spracheinstellung einer Webseite dienen.

Werden von Cookies jedoch personenbezogene Daten gespeichert, müssen genaue Informationspflichten eingehalten werden und es ist die Einwilligung des Benutzer einzuholen [Hoe16, Seite 455]. Der rechtlich sicherste Weg in Deutschland wäre die Benutzer vor der Speicherung von Cookies auf die Verwendung dieser hinzuweisen und eine ausdrückliche Einwilligung einzufordern [Hub16]. Für die Einwilligung wäre somit eine vorgeschaltete Webseite notwendig, die zunächst über Cookies informiert und den Benutzer die Entscheidung überlässt, die Webseite zu nutzen. Ein Nachteil jedoch ist, dass Besucher diese wenig benutzerfreundliche Lösung als lästig empfinden könnten und dadurch weniger auf die Webseite zugegriffen wird. Eine elegantere Zwischenlösung wäre es, beim ersten Aufruf der Webseite einen Einwilligungstext durch einen auf den ersten Blick sichtbaren Banner/Balken für den Benutzer einzublenden [Hub16]. Der eingebundene Text sollte auf die Verwendung von Cookies hinweisen und darüber informie-

ren, dass der Benutzer durch die weitere Nutzung der Webseite der Einbindung von Cookies zustimmt. Ebenfalls sollte der Text eine Bestätigung per Mausklick beinhalten. Darüber hinaus muss ein entsprechender Abschnitt zu Cookies in der Datenschutzerklärung für den Benutzer formuliert sein, der Auskunft darüber gibt, welche Cookies eingebunden werden und wie das Setzen dieser über die Browsereinstellung verhindert werden kann [Ros15]. Dadurch wird eine erleichterte Opt-In-Lösung in die Webseite eingebunden, die davon ausgeht, dass die Einwilligung stillschweigend durch die weitere Nutzung der Webseite erteilt wird.

## 4. Haftung von Online-Diensten

Die Haftung im Internet spielt sowohl national als auch international im Online-Recht eine wesentliche Rolle. Kommt es auf einer Internetseite zu einer Rechtsverletzung, beispielsweise einer Persönlichkeits- oder Urheberrechtsverletzung, stellt sich die Frage, ob auch der Webseitenbetreiber für die eingestellten Informationen und Inhalte verantwortlich gemacht werden kann. Die rechtliche Grundlage für dieses Thema beschreibt das Telemediengesetz (TMG). Dieses Gesetz richtet sich beim Umfang der Haftung nach den unterschiedlichen Kategorien, in die Internetdiensteanbieter aufgeteilt werden können.

In diesem Kapitel sollen die verschiedenen Unterteilungen von Internetdiensteanbietern des Telemediengesetzes aufgezeigt werden. Außerdem ist es wichtig zu klären, auf welche Rechtsverletzungen als Webseitenbetreiber eines E-Learning Systems wie THMcards zu achten sind. Des Weiteren spielen im Rahmen des Projektes THMcards die Verantwortlichkeiten eines Host-Providers eine wichtige Rolle und sollen näher betrachtet werden.

### 4.1. Internet Service Provider

Bei einem Internet Service Provider (ISP), im deutschsprachigen Raum Internetdiensteanbieter oder auch oft nur kurz Provider, handelt es sich grundsätzlich um einen kommerziellen Anbieter verschiedener technischer Leistungen, die für die Nutzung oder den Betrieb von Diensten und Inhalten im Internet erforderlich sind [Wikc]. Nach dem Telemediengesetz unterscheidet man zwischen drei verschiedenen Providern: Dem Content-Provider (§ 7 Absatz 1 TMG), dem Access-Provider (§§ 8, 9 TMG) und dem Host-Provider (§ 10 TMG).

#### 4.1.1. Content-Provider

Beim Content-Provider handelt es sich um einen Informationslieferanten. Dieser stellt eigene Informationen wie redaktionelle Beiträge zur Verfügung. Bei eigenen

Informationen handelt es sich um eigens eingestellte oder sich zu eigen gemachte Inhalte. Zu den eigenen Informationen gehören auch solche, für deren Verbreitung der Provider seinen eigenen Internetauftritt auf Mietbasis für die Nutzung dritter Personen bereit hält. Die rechtliche Verantwortung über die verbreiteten Inhalte trägt hierbei der Provider. Erst wenn sich der Provider konkret und ausdrücklich von betreffenden Inhalten distanziert, kann nicht mehr von eigenen Informationen gesprochen werden. [Hoe16, Seite 479-481]

##### **4.1.2. Access-Providers**

Zu den Aufgaben des Access-Providers gehört die Vermittlung fremder Informationen im Internet oder Netzwerkbetrieb [ItR]. Darüber hinaus wird mithilfe bereitgestellter Wahlverbindungen, Breitbandzugänge und Standleitungen Benutzern der Zugang zum Internet ermöglicht. Beim Access-Provider werden weder Speicherplatz noch eigene Inhalte zur Verfügung gestellt [ItR]. Bekannte Access-Provider für Privatkunden in Deutschland sind beispielsweise T-Online, Arcor, AOL, freenet.de oder 1&1.

##### **4.1.3. Host-Provider**

Host-Provider stellen Speicherplatz im Internet, beispielsweise durch eigene Webseiten oder Webserver, für fremde Informationen und Inhalte zur Verfügung. Der entscheidende Unterschied zum Content-Provider liegt darin, dass sich die Betreiber solcher Webseiten konkret und ausdrücklich von den eingestellten Inhalten distanzieren. [Hoe16, Seite 488-492]

## **4.2. Rechtsverletzungen**

Rechtsverletzende Inhalte werden im Internet häufig anonym von den jeweiligen Autoren eingestellt. Betroffene haben damit nicht die Möglichkeit, sich mit den bestehenden Abwehransprüchen direkt an den Schädiger zu wenden. Darüber hinaus wird der Autor von rechtsverletzenden Inhalten in Deutschland durch das Datenschutzgesetz vor der Preisgabe seiner Identität geschützt. Dem Betroffenen bleibt damit in vielen Fällen nichts anderes übrig, als sich an den jeweiligen Webseitenbetreiber zu halten, um die gegebene Rechtsverletzung zu beseitigen. Dadurch geraten Provider solcher Webseiten häufig ins Visier der Betroffenen.

Für die Benutzer von THMcards besteht durch das virtuelle Teilen selbst erstellter Lernkarten die Möglichkeit, mit anderen Lernenden des Systems eine gemeinschaftliche Wissenssammlung aufzubauen. Dadurch können auf THMcards rechtsverletzende Inhalte entstehen, für die sich der Provider verantworten muss. Im Folgenden soll nun aufgezeigt werden, welche Rechtsverletzungen daraus resultieren können.

### 4.2.1. Persönlichkeitsverletzung

Das Allgemeine Persönlichkeitsrecht dient dem Schutz der Persönlichkeit einer Person vor Eingriffen in ihren Lebens- und Freiheitsbereich. Es ist ein absolutes, umfassendes Recht auf Achtung und Entfaltung der Persönlichkeit. Das Recht auf freie Entfaltung der Persönlichkeit sichert jedem Einzelnen einen eigenständigen Lebensbereich, in dem seine Individualität entwickelt und gewahrt werden kann. Ein Eindringen oder Einblick durch andere soll durch das Recht verhindert werden. [Gru] [Wikg]

Beiträge dürfen nicht mit falschen Behauptungen, beleidigenden oder vergleichbaren rechtswidrigen Inhalten veröffentlicht werden. Durch die Verletzungen des Allgemeinen Persönlichkeitsrechts durch Dritte können dem Verletzten Unterlassungs-, Beteiligungs- (§ 1004 BGB) sowie Schadenersatzansprüche (§ 823 Absatz 1, 2 BGB) zustehen. [Wikg]

### 4.2.2. Urheberrechtsverletzungen

Bei der Verwendung von E-Learning Systemen ist vor allem das Urheberrecht von Bedeutung. Grundsätzlich stehen dem Urheber, also dem Schöpfer eines Werkes wie eines Fotos, Videos oder ähnlichem, die Rechte der Veröffentlichung und Verwertung selbst zu [DFN15]. Er kann also entscheiden, ob und wie sein Werk zu veröffentlichen ist (§ 12 Absatz 1 Urheberrechtsgesetz (UrhG)). Diese Rechte sind allein dem Urheber vorbehalten [DFN15]. Grundsätzlich dürfen geschützte Werke nur genutzt werden, wenn der Urheber dem zugestimmt hat [DFN15]. Jedoch gibt es auch Ausnahmen von dieser Regel, sodass Werke, die nicht unter einer freien Lizenz stehen, unter bestimmten Zwecken ohne Zustimmung genutzt werden können [Kre07]. Diese sogenannten *Schrankenbestimmungen* spielen bei der Herstellung und Verwertung von Inhalten auf E-Learning Systemen eine wichtige

Rolle, wenn Lerninhalte Werke Dritter enthalten. Folgende Schrankenbestimmungen haben in der Lehre eine besondere Bedeutung:

**Das Zitatrecht (§ 51 UrhG):** Das Zitatrecht erlaubt es, urheberrechtlich geschützte Werke oder Werkteile in einem eigenen Werk zu verwenden. Das zitierende Werk darf dann mit den enthaltenen Werken Dritter im Internet vervielfältigt oder veröffentlicht werden. Gestattet ist ein Zitat, wenn ein Zitatzweck vorliegt (z.B. eine inhaltliche Auseinandersetzung mit dem Werk), die Quelle angegeben (§ 63 UrhG) und das fremde Werk nicht verändert wurde (§ 62 UrhG). [Kre07]

**Öffentliche Zugänglichmachung für Unterricht und Forschung (§ 52a UrhG):** Diese Regelung bezieht sich auf technische Hilfsmittel zur Verwertung und Herstellung von Lerninhalten im Rahmen von Unterricht und Forschung und ist damit auch für das E-Learning von Bedeutung. Durch die öffentliche Zugänglichmachung geschützter Werke ist eine zustimmungsfreie Nutzung zu Wissenschafts- oder Lehrzwecken möglich. Dadurch kann in E-Learning Systemen geschütztes Material verwendet werden. [Kre07]

**Vervielfältigung zum privaten und eigenen Gebrauch (§ 53 UrhG):** Dieses Recht erlaubt es, urheberrechtlich geschützte Werke zum privaten (§ 53 Absatz 1 UrhG) oder eigenen Gebrauch (§ 54 Absatz 2 UrhG) zu vervielfältigen. Der private Gebrauch beschränkt sich hierbei auf das ungehinderte Nutzen von geschützten Werken zu Hause oder im Freundeskreis. Vervielfältigungen zum eigenen Gebrauch können auch von Unternehmen und öffentlichen Einrichtungen sowie zu beruflichen Zwecken verwendet werden. Die Vervielfältigung ist nach §§ 54 ff. UrhG zu vergüten. Somit fließt ein geringer Anteil der eingenommenen Preise, die durch die Kopiergeräte- und Leermedienabgaben erhoben werden, an den Urheber der Werke. [Kre07]

Werden Rechte des Urhebers verletzt, kann dieser auf Unterlassung und Schadensersatz klagen (§ 97 Absatz 1, 2 UrhG). Darüber hinaus ist bei einer öffentlichen Verwertung eines urheberrechtlich geschützten Werkes ohne Einwilligung des Urhebers gemäß § 106 Absatz 1 UrhG mit einer Sanktion zu rechnen. [DFN15]

Grundsätzlich sind fast alle Inhalte, die im Internet veröffentlicht werden, urheberrechtlich geschützt. Sollte kein ausdrücklicher Hinweis angebracht sein, bei-

spielsweise das „©“ für Copyright, muss trotzdem davon ausgegangen werden, dass fremde Inhalte nicht einfach genutzt werden dürfen. Es ist jedoch erlaubt, Inhalte zu verwenden, die vom Urheber ausdrücklich zur Verwendung freigegeben worden sind. Hierbei handelt es sich um Inhalte unter sogenannten *freier Lizenzen* [iRi09]. Unter diesen Lizenzen veröffentlichte Inhalte können ohne Einwilligung der Urheber öffentlich verwertet werden. Es können jedoch von den Urhebern Bedingungen festgelegt werden, die bei der Weiterverarbeitung einzuhalten sind, beispielsweise ob Inhalte nicht verändert werden dürfen oder eine kommerzielle Nutzung erlaubt ist [iRi09]. Die wohl bekannteste Lizenz für freie Inhalte ist die *Creative Commons*. Diese besitzt für Urheber, die Inhalte unter eine freie Lizenz stellen wollen, ein flexibles System von Nutzungsbedingungen [bpb13]. Der Lizenzbaukasten von Creative Commons besteht aus folgenden Modulen:

- **Namensgebung:** Die Vervielfältigung, Verwertung und Veröffentlichung von Inhalten ist nur unter der Bedingung erlaubt, dass der Name des Urhebers genannt wird. [bpb13]
- **Keine Bearbeitung:** Es dürfen nur unveränderte Kopien der Werke vom Urheber vervielfältigt, verwertet und veröffentlicht werden. Die Bearbeitung solcher Werke ist nicht erlaubt. [bpb13]
- **Nicht-kommerzielle Nutzung:** Die Vervielfältigung, Verwertung und Veröffentlichung von Inhalten ist nur unter nicht gewerblichen Zwecken erlaubt. [bpb13]
- **Weitergabe unter gleichen Bedingungen:** Bearbeitete Inhalte von Urhebern dürfen nur unter der gleichen Lizenzvereinbarung vervielfältigt, verwertet und veröffentlicht werden wie die ursprüngliche Version des Werkes. Wird beispielsweise ein Remix eines Musikstücks erstellt, muss auch der Remix unter der gleichen CC-Lizenz veröffentlicht werden. [bpb13]

#### 4.2.3. Hyperlinks auf fremde Inhalte

Das Setzen von Links zu Inhalten anderer Webseiten ist eine häufig genutzte Möglichkeit, fremde Informationen für andere Benutzer öffentlich zugänglich zu machen. Bei einer Verlinkung auf fremde Webseiten stellt sich jedoch die Frage, ob dieser Verweis überhaupt rechtlich zulässig ist. Hierbei wird zwischen drei unterschiedlichen Typen von Links unterschieden:

- **Surface-Link:** Beim Surface-Link handelt es sich um einen "normalen" Link, der auf die Startseite eines Webauftritts verweist. [Bet]
- **Deep-Link:** Mit einem Deep-Link wird direkt auf eine Unterseite des Webauftritts verwiesen. [Bet]
- **Framing:** Bei Framing werden fremde Inhalte in eine Webseite eingebettet. Dies ist für den Nutzer häufig nicht unmittelbar ersichtlich, da kein Link zu der ursprünglichen Webseite angezeigt wird. [Bet]

Das Setzen eines einfachen Hyperlinks wie Surface- und Deep-Link auf Inhalte einer anderen Webseite ist in der Regel urheberrechtlich zulässig, da hierbei keine Vervielfältigung oder öffentliche Zugänglichmachung fremder Informationen stattfindet. Jedoch kann es beim Setzen eines Deep-Links aus wettbewerbsrechtlicher Sicht zu Problemen kommen. Dadurch besteht nämlich die Möglichkeit, sonstige Inhalte der verlinkten Seite sowie auf den übersprungenen Seiten erscheinende Werbung des Anbieters zu umgehen. Bei der sogenannte *Paperboy-Entscheidung* wurde durch den Bundesgerichtshof eine Entscheidung bezüglich des Problems getroffen (BGH, Urteil vom 17.7.2003 - I ZR 259/00). Anbieter einer Webseite haben mit Verweisen zu rechnen und müssen sich mit Verlinkungen ihrer Webseite einverstanden erklären, sofern die Inhalte nicht durch technische Maßnahmen geschützt worden sind. [Sch11a] [Bet]

Anders verhält es sich, wenn fremde Inhalte durch Framing in die Webseite eingebunden werden. Handelt es sich bei den eingebundenen Informationen um urheberrechtlich geschützte Inhalte, ist eine solche Nutzung fremder Inhalte urheberrechtswidrig. Framing ist erst erlaubt, wenn der Anbieter der verlinkten Inhalte einer Vervielfältigung ausdrücklich zugestimmt hat. Werden verlinkte Inhalte nicht urheberrechtlich geschützt, kann es zumindest zu wettbewerbsrechtlichen Problemen kommen. Hier besteht die Möglichkeit der Irreführung von Kunden (§ 5 Gesetz gegen den unlauteren Wettbewerb). [Bet]

Unter Umständen kann es dazu kommen, dass man bei gesetzten Hyperlinks für den Inhalt der verlinkten Webseite haften muss. Hier ist von Bedeutung, ob der Benutzer von den rechtswidrigen Inhalten Kenntnis hat oder sich diese zu eigen macht [jus]. Externe Links sollten also immer als solche gekennzeichnet und nicht als eigene Inhalte eingebettet werden. Sollte bei den eingebetteten Inhalten ein rechtswidriger Verstoß vorliegen oder erkennbar sein, dass bewusst auf eine

rechtswidrige Webseite verwiesen wurde, könnte der Benutzer haftbar gemacht werden.

## 4.3. Verantwortlichkeit der Host-Provider

E-Learning Systeme wie THMcards stellen ihren Benutzern Speicherplatz auf einem Server zur Verfügung, um eigene Lerninhalte zu kreieren und mit anderen Benutzern online in einer virtuellen Lernumgebung zu teilen. Werden auf diesen Servern Inhalte abgelegt, die zu einer Rechtsverletzung (Persönlichkeits- oder Urheberrechtsverletzungen) führen, stellt sich die Frage, ob für diese Rechtsverletzung auch der Host-Provider verantwortlich gemacht werden kann.

Bei der Haftung im Internet ist zu unterscheiden, ob es sich bei den bereitgestellten Informationen auf einer Webseite um eigene oder fremde Inhalte handelt [Nag08]. Wer eigene Inhalte bereitstellt, ist gemäß § 7 Absatz 1 TMG für die volle Haftung verantwortlich, wenn durch diese Rechte verletzt worden sind [Nag08]. Zu eigenen Inhalten gehören auch Daten, die durch Dritte bezogen und sich zu Eigen gemacht wurden, wie beispielsweise durch das Setzen von Hyperlinks oder durch Framing.

Schwieriger gestaltet sich die Rechtslage beim Angebot von fremden Inhalten, die Host-Provider zur Nutzung bereithalten. Nach § 10 TMG sind Internetdiensteanbieter, die fremde Informationen für einen Benutzer speichern, grundsätzlich nicht verantwortlich. Sie sind erst dann haftbar, wenn diese positive Kenntnisse von der rechtswidrigen Handlung haben, also nachgewiesen werden kann, dass sie von der Rechtswidrigkeit der Inhalte wussten. Ein Internetdiensteanbieter ist auch dann haftbar, wenn dieser trotz starken Verdachts keinerlei Bemühungen zur Klärung unternommen hat. [Nag08]

### 4.3.1. Festes Verfahren zur Löschung eines rechtswidrigen Inhalts

Sobald ein Internetdiensteanbieter Kenntnis über die Rechtswidrigkeit seiner Inhalte hat, muss dieser unverzüglich tätig werden, um die Information zu entfernen oder den Zugang zu dieser zu sperren. Trotz der Verpflichtung, rechtswidrige Inhalte zu sperren oder zu entfernen, trifft den Internetdiensteanbieter gemäß § 7 Absatz 2 TMG keine Überwachungspflicht [Nag08]. Eine Entfernungs- und

Sperrungspflicht erfolgt erst, wenn der Anbieter Kenntnis über die Rechtswidrigkeit seiner Inhalte hat. Nach einer Kenntnisnahme und Entfernung eines Inhaltes müssen vom Internetdienstanbieter Vorkehrungen getroffen werden, um weiteren Rechtswidrigkeiten zu vermeiden [DAS15].

Der Bundesgerichtshof (BGH) schreibt ein klares Verfahren vor, wenn rechtswidrige Inhalte auf einer Webseite des Host-Providers vorliegen (BGH, Urteil vom 25.10. 2011, Az.: VI ZR 93/10) [Jae11]:

- Der Betroffene setzt den Host-Provider in Kenntnis, dass eine Rechtsverletzung vorliegt. Dieser Hinweis muss konkret verfasst sein, sodass er *ohne eingehende rechtliche und tatsächliche Überprüfung bejaht werden kann*.
- Diese Stellungnahme muss vom Host-Provider an den Verantwortlichen weitergeleitet werden. Meldet sich dieser nicht innerhalb einer angemessenen Frist, muss der beanstandete Inhalt vom Host-Provider gelöscht werden.
- Bleibt der Verantwortliche des Inhalts bei seiner Aussage, hat der Betroffene die Möglichkeit darauf zu reagieren und eine Rechtsverletzung zu beweisen. Bleibt eine Stellungnahme des Betroffenen aus oder legt dieser die erforderlichen Nachweise nicht vor, bleibt der Inhalt bestehen.
- Gelingt es dem Betroffenen mit der Stellungnahme oder den vorgelegten Nachweisen einen rechtswidrigen Inhalt zu beweisen, wird dieser gelöscht.

## 5. Datensicherheit

Das Thema Datensicherheit ist für die Beteiligten eines E-Learning Systems (Administrator, Lernende, Lehrende) von großer Wichtigkeit. Es besteht grundsätzlich die Gefahr, dass die erhobenen Daten von Personen eingesehen werden, die dazu nicht berechtigt sind. Besonders problematisch wird es, wenn es sich hierbei um sensible Daten handelt, die dem Datenschutz unterliegen.

Im Unterschied zum Datenschutz befasst sich die Datensicherheit mit dem Schutz von Daten im Allgemeinen. Durch die Datensicherheit werden also Daten geschützt, unabhängig davon, ob diese personenbezogen sind oder nicht. Diese Daten können sowohl in digitaler als auch analoger Form, wie z.B. auf Papier, vorliegen. Datensicherheit betrifft sämtliche Maßnahmen, die notwendig sind, um Daten vor Verlust, Verfälschung, Beschädigung oder Löschung zu schützen. Im Kontext des Datenschutzes gemäß § 9 BDSG ist der Schutz von Daten durch die Umsetzung geeigneter technischer und organisatorischer Maßnahmen zu gewährleisten. [Dat16] [bit16]

E-Learning Systeme müssen also vor unbeabsichtigten Fehlern und Ereignissen (z.B. technischen Fehlern, Fahrlässigkeit, Programmierfehlern) und beabsichtigten Angriffen (z.B. Abhören, Manipulation und Zerstören von Informationen) von außen (Outsider wie Hacker) und innen (Insider wie Administratoren, Programmierern) geschützt werden. Im Englischen wird diesbezüglich zwischen den Begriffen *Security* und *Safety* unterschieden. *Security* beschreibt den Schutz vor beabsichtigten Angriffen und *Safety* den Schutz vor unbeabsichtigten Ereignissen. Datenschutz und Datensicherheit beschäftigt sich vor allem mit dem Schutz vor beabsichtigten Angriffen (*Security*). Hierbei wird zwischen den drei Schutzziele textitVertraulichkeit, textitIntegrität und textitVerfügbarkeit unterschieden (siehe Tabelle 5.1). [DHF07, Seite 488-510]

Um die permanente Sicherheit eines Systems zu gewährleisten, ist eine kontinuierliche Überwachung und Weiterentwicklung der Datensicherheitsmaßnahmen notwendig, da Änderungen am System oder neu auftretende Schwachstellen neue Sicherheitsherausforderungen darstellen können. Daher sollte Datensicher-

Security		Safety	
Vertraulichkeit:	Anonymität, Unbeobachtbarkeit, Unverkettbarkeit, Pseudonymität, Abhörsicherheit, Sicherheit gegen unbefugten Gerätezugriff	Verfügbarkeit:	Funktionssicherheit, technische Sicherheit
Integrität:	Zurechenbarkeit, Übertragungsintegrität, Abrechnungssicherheit	Sonstige Schutzziele:	Maßnahmen gegen hohe Gesundheitsbelastung
Verfügbarkeit:	Ermöglichen von Kommunikation		

Tabelle 5.1.: Abgrenzung von Security und Safety [DHF07, Seite 488-510]

heit nicht als einmaliges Ereignis, sondern als wiederkehrender Prozess verstanden werden. Ein häufig verwendetes Schema des Prozessablaufes in der Datensicherheit ist der sogenannte PDCA-Zyklus (Plan, Do, Check, Act), der in Abbildung 5.1 dargestellt ist [Kap07, Seite 10-11]. Mithilfe der fortlaufenden Abarbeitung der vier Prozessschritte können die Schutzziele sichergestellt und damit eine hohe Datensicherheit gewährleistet werden. Folgende Prozessschritte werden im Zyklus definiert:

- **Plan:** Die Anforderungen an die Datensicherheit werden analysiert und die strategischen Ziele festgelegt. Zur Umsetzung der Ziele wird ein Konzept entwickelt.
- **Do:** Das entwickelte Konzept wird durch technische und/oder begleitende organisatorische Maßnahmen umgesetzt. [Kap07, Seite 10-11]
- **Check:** Die Wirksamkeit der Maßnahmen wird sowohl technisch als auch organisatorisch überwacht. Hierzu werden Daten gesammelt, ausgewertet und mit den erwarteten Ergebnissen verglichen. [Kap07, Seite 10-11]
- **Act:** Mögliche Differenzen zwischen erwarteten und erreichten Ergebnissen werden analysiert. Mithilfe der Analyse werden Änderungen am Konzept vorgenommen oder ein neues Konzept aufgestellt. [Kap07, Seite 10-11]

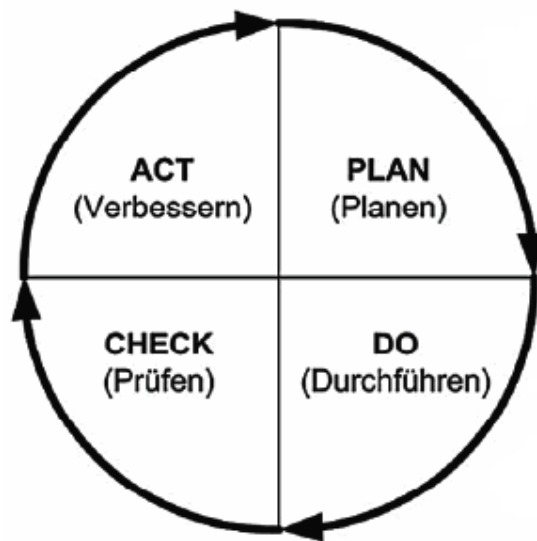


Abbildung 5.1.: PDCA-Zyklus [PDC]

In diesem Kapitel wird die Datensicherheit im Sinne von Security näher betrachtet. Hierzu werden die Schutzziele der Datensicherheit genauer erläutert. Anschließend soll erklärt werden, welche technischen und organisatorischen Maßnahmen konkret eingehalten werden müssen, um die Sicherheit von Daten zu gewährleisten. Am Ende sollen Angriffstechniken auf eine Webseite eines E-Learning System aufgezeigt werden, um die Problematik der Datensicherheit nochmals zu verdeutlichen.

## 5.1. Schutzziele

Um Angriffe auf Daten oder Informationen besser beschreiben zu können, werden Schutzziele der Datensicherheit in verschiedene Kategorien unterteilt. Zu den klassischen Schutzzielen gehören die *Vertraulichkeit*, *Integrität* und *Verfügbarkeit*. Jedoch sind diese für E-Learning Systeme nicht ausreichend. Aus diesem Grund werden an dieser Stelle zusätzlich die Schutzziele *Authentizität* und *Verbindlichkeit* beschrieben.

### 5.1.1. Vertraulichkeit

Vertraulichkeit bedeutet, dass Daten nur von befugten Personen eingesehen und verwendet werden dürfen. Bedrohungen der Vertraulichkeit betreffen nicht nur die Daten selbst, sondern auch Systeme oder Konfigurationen. Eine unbefug-

te Informationsgewinnung, beispielsweise durch das Ausspähen von Login-Daten durch einen Unbefugten, stellt eine Verletzung der Vertraulichkeit dar. Deshalb ist es notwendig, Sicherheitsmaßnahmen zu treffen, damit ein unbefugter Zugriff auf gespeicherte oder übermittelte Daten des Systems verhindert werden kann. [Dat16]

Für die Vertraulichkeit in E-Learning Systemen besteht sowohl vom Benutzer als auch vom Anbieter ein Interesse, die unbefugte Manipulation von Daten zu verhindern. Der Anbieter möchte beispielsweise vermeiden, dass kommerziell angebotene Lerninhalte beliebig oft kopiert und vervielfältigt werden. Benutzer eines E-Learning Systems erwarten, dass personenbezogene Daten nach den Grundsätzen des Datenschutzes gespeichert und verarbeitet werden.

Grundsätzlich kann eine angemessen starke Verschlüsselung der gespeicherten oder zu übertragenden Daten die Vertraulichkeit gewährleisten. Übertragene Daten innerhalb eines Systems können mithilfe des Secure Sockets Layer (SSL) gesichert werden. Darüber hinaus kann eine entsprechende Rechte- und Benutzerverwaltung des E-Learning Systems dabei helfen, die Vertraulichkeit sicherzustellen.

### 5.1.2. Integrität

Die Integrität impliziert das Daten korrekt, unverändert und verlässlich sein müssen. Diese soll verhindern, dass unerwünschte Veränderungen an Daten vorgenommen werden oder dass diese beschädigt werden. Die Integrität ist auch dann betroffen, wenn die Soft- oder Hardware durch fehlerhafte Funktionen falsche Ergebnisse liefert. Dadurch können Angriffe nicht nur absichtlich sondern auch versehentlich erfolgen. [Dat16]

In E-Learning Systemen ist es wichtig, sicherzustellen, dass Inhalte nicht unbefugt manipuliert werden können. Beispielsweise kann das unbefugte Verändern oder Löschen zertifizierter Lerninhalte dazu führen, dass ein E-Learning System an Glaubwürdigkeit und Vertrauenswürdigkeit verliert.

Um die Integrität eines E-Learning Systems zu gewährleisten, ist ebenfalls die Festlegung einer entsprechenden Rechte- und Benutzerverwaltung erforderlich. Es muss außerdem sichergestellt werden, dass die eingesetzte Software fehlerfrei läuft und keine Hintertüren oder andere Sicherheitslücken aufweist.

### 5.1.3. Verfügbarkeit

Mit der Verfügbarkeit wird gewährleistet, dass Daten und Systeme zur Verfügung stehen und von autorisierten Benutzern jederzeit genutzt werden können, wenn diese gebraucht werden. Die Verfügbarkeit ist verletzt, wenn eine ungewollte Unterbrechung, beispielsweise ein Serverausfall oder eine gestörte Internetverbindung, vorliegt. Verzögerungen, die aus normalen Verwaltungsmaßnahmen resultieren, stellen keine Verletzung der Verfügbarkeit dar. [Dat16]

Die Zeit in der ein System tatsächlich nutzbar ist, wird als Prozentsatz der Zeit angegeben. Die Verfügbarkeit lässt sich dann wie folgt berechnen [Wikk]:

$$\text{Verfügbarkeit} = \frac{\text{Gesamtzeit} - \text{Ausfallzeit}}{\text{Gesamtzeit}}$$

Anschließend kann aus der sich ergebenden Verfügbarkeit die maximal erlaubte Ausfallrate eines Systems wie folgt berechnen:

$$\text{Ausfallrate} = 1 - \text{Verfügbarkeit}$$

Geht man von einer Verfügbarkeit von 0,999 (also 99,9%) aus, erhält man eine Ausfallzeit von 8,76 Stunden pro Jahr, wenn von einem durchgängigen Betrieb des Systems (also 24 x 7 x 365 Stunden) ausgegangen wird.

Ist ein Benutzer berechtigt, einen Zugriff auf einen Lerninhalt eines E-Learning Systems durchzuführen, sollte gewährleistet werden, dass dieser Zugriff ohne Unterbrechung möglich ist. Die Forderung nach der Verfügbarkeit von E-Learning Systemen spielt eine entscheidende Rolle, wenn Lerninhalte als Prüfungsvoraussetzung elektronisch in einer vorgegebenen Zeit abgewickelt werden müssen.

### 5.1.4. Authentizität

Unter Authentizität wird die Echtheit, Zuverlässigkeit und Glaubwürdigkeit eines Objekts oder Subjektes verstanden. Bei der Authentizität muss sichergestellt werden, dass die Herkunft der versendeten Informationen eindeutig nachvollzogen werden kann. Dementsprechend ist die Authentizität verletzt, wenn unbefugt Informationen erstellt werden, beispielsweise unter einer falschen Identität. [Dat16]

In einem E-Learning System sollte sichergestellt werden, dass nur autorisierte Benutzer auf geschützte Objekte zugreifen können, wie beispielsweise der rechtmäßige Käufer eines Lerninhalts. Hierzu muss zunächst die Identität des Benut-

zers überprüft werden. In E-Learning Systeme müssen zuverlässige Verfahren zur Authentifizierung verwendet werden, um die eindeutige Identität eines Benutzers zu erkennen und damit die Authentizität zu gewährleisten. Diesbezüglich werden üblicherweise digitale Signaturen und/oder Passwörter eingesetzt.

### 5.1.5. Verbindlichkeit

Die Verbindlichkeit eines Systems wird garantiert, wenn ein Zugriff bzw. die Durchführung einer Aktion im Nachhinein eindeutig einem Benutzer zugeordnet werden kann. Damit soll das Abstreiten einer gesendeten Information durch den Absender oder Empfänger verhindert werden. Der Empfänger einer Information soll beweisen können, dass Daten tatsächlich vom berechtigten Absender stammen. Außerdem kann auch der Sender beweisen, dass die Daten dem Empfänger zugegangen sind. [Dat16]

Dieses Konzept der Verbindlichkeit ist in E-Learning Systemen nötig, wenn eingereichte Hausaufgaben, ausgearbeitete Lerninhalte oder andere bewertete Dokumente eindeutig einem Benutzer zugeordnet werden sollen.

## 5.2. Technische und organisatorische Maßnahmen

Ein wichtiger Bereich der Datensicherheit sind die technischen und organisatorischen Maßnahmen, die von allen öffentlichen und privaten Bereichen in Deutschland getroffen werden müssen, um personenbezogene Daten zu erheben, zu verarbeiten oder zu nutzen. Mit den Maßnahmen wird sichergestellt, dass personenbezogene Daten vor Missbrauch, Fehlern und Unglücksfällen geschützt sind [Bsi13c].

Im Gesetz werden eine Reihe von Maßnahmen als Anforderungskatalog festgehalten, um die Datensicherheit zu gewährleisten. Allerdings wird darauf verzichtet, diese Maßnahmen als verpflichtend vorzuschreiben, da diese von der Art der Daten und der Aufgaben, für die sie verwendet werden, sowie von den organisatorischen Bedingungen, den räumlichen Verhältnissen, der personellen Situation und anderen Rahmenbedingungen abhängig sind. So wird vom Gesetz nur verlangt *technische und organisatorische Maßnahmen zu treffen, die erforderlich sind, um die Ausführung der Vorschriften zu gewährleisten*. [Bsi13c]

Unter technischen Maßnahmen sind alle Schutzmaßnahmen zu verstehen, die physisch umsetzbar sind, wie beispielsweise allgemeine Baumaßnahmen, Alarm-

anlagen oder Überwachungseinrichtungen oder Maßnahmen, die in Soft- und Hardware umgesetzt werden können, wie der Einsatz von Benutzerkonten oder die Passwörterzwingung[Wiki]. Bei organisatorischen Maßnahmen handelt es sich um Schutzmaßnahmen, die durch Verfahrens- und Vorgehensweisen umgesetzt werden [Wiki]. Dazu gehören Maßnahmen wie die Benutzeranmeldung oder Intervalle zur Stichprobenprüfung. Welche technischen und organisatorischen Maßnahmen einzuhalten sind, wird im Datenschutzgesetz in der Anlage zu § 9 Satz 1 BDSG gelistet:

**Zutrittskontrolle:** Mit dem Begriff *Zutritt* wird der physische Zugang von Personen in ein Gebäude und Räumlichkeiten, in denen die IT-Systeme genutzt werden, bezeichnet [sof]. Hierbei kann es sich beispielsweise um Rechenzentren mit Web-Servern, Applikationsservern oder Datenbanken oder um Arbeitsräume mit Arbeitsrechnern handeln. Mit der Zutrittskontrolle soll verhindert werden, dass Unbefugte Zugang zu Datenverarbeitungsanlagen haben, die personenbezogene Daten verwerten oder nutzen [Bsi13c]. Beispiele für die Umsetzung der Maßnahme wären die Verwendung von Berechtigungsausweisen oder der Einsatz von Alarmanlagen oder Überwachungseinrichtungen.

**Zugangskontrolle:** Unter *Zugang* wird das Eindringen in das Datenverarbeitungssystem selbst verstanden [sof]. Ergänzend zur Zutrittskontrolle soll die Zugangskontrolle verhindern, dass Datenverarbeitungssysteme, die personenbezogene Daten verwerten oder nutzen, von Unbefugten verwendet werden können [sof]. Hierzu kommt beispielsweise ein effektives Passwortmanagement und eine entsprechende Protokollierung der Passwortnutzung zum Einsatz.

**Zugriffskontrolle:** Die Zugriffskontrolle soll gewährleisten, dass die Benutzer eines Datenverarbeitungssystems ausschließlich Zugriff auf Daten haben, für die eine Zugriffsberechtigung besteht [Bsi13c]. Außerdem soll verhindert werden, dass Daten bei der Verarbeitung, Nutzung und nach der Speicherung nicht unbefugt gelesen, kopiert, verändert oder entfernt werden können [Bsi13c]. Solche Maßnahmen können durch ein rollenbasiertes Berechtigungskonzept umgesetzt werden.

**Weitergabekontrolle:** Durch die Weitergabekontrolle soll verhindert werden, dass personenbezogene Daten bei der elektronischen Übertragung, während ihres

Transports oder ihrer Speicherung auf Datenträger unbefugt gelesen, kopiert, verändert oder entfernt werden können [Date]. Darüber hinaus soll überprüfbar und nachvollziehbar sein, an welche Stellen personenbezogene Daten im Datenverarbeitungssystem übermittelt werden [Date]. Um die elektronische Übertragung zu sichern, kommen Maßnahmen wie Verschlüsselungsverfahren oder Firewalls zum Einsatz. Für die Sicherung beim Transport werden beispielsweise verschlossene Behälter verwendet.

**Eingabekontrolle:** Mit der Eingabekontrolle wird sichergestellt, dass nachträglich überprüft und festgestellt werden kann, ob und von wem personenbezogene Daten zu welcher Zeit eingegeben, verändert oder gelöscht worden sind [Bsi13c]. Dazu gehören Maßnahmen wie die Protokollierung von Dateneingaben und Datenlöschungen.

**Auftragskontrolle:** Werden personenbezogene Daten im Auftrag von Dritten verarbeitet, muss der Auftragsnehmer gewährleisten, dass diese nur nach den Weisungen des Auftraggebers verarbeitet werden. Im Rahmen der Auftragsdatenverarbeitung hat der Auftraggeber entsprechende Weisungen zu erteilen und die Einhaltung dieser durch den Auftragnehmer regelmäßig zu kontrollieren, z.B. durch Vorortkontrollen oder Stichprobenprüfungen. [Date]

**Verfügbarkeitskontrolle:** Die Maßnahmen zur Verfügbarkeit sollen sicherstellen, dass personenbezogene Daten vor zufälliger Zerstörung oder Verlust geschützt sind [Bsi13c]. Dazu gehören insbesondere Vorfälle wie Stromausfälle, Blitzschläge, Feuer- und Wasserschäden [Date]. Um für solche Vorfälle vorzusorgen, sollten Sicherheitsmaßnahmen wie z.B. Brandschutzmaßnahmen, der Einsatz einer unterbrechungsfreien Stromversorgung, das Erstellen von Backups und der Schutz vor Diebstahl vorgenommen werden.

**Trennungsgebot:** Ziel dieser Maßnahme ist, zu gewährleisten, dass personenbezogene Daten, die zu unterschiedlichen Zwecken erhoben werden, getrennt voneinander verarbeitet werden können [Bsi13c]. Dieses Ziel kann z.B. durch entsprechende Zugriffsberechtigungen oder durch eine getrennte Ordnerstruktur realisiert werden.

## 5.3. Angriffsszenarien

Um bei der Entwicklung von Webanwendungen einen effektiven Schutz aufbauen zu können, ist es notwendig, sich zunächst mit den verschiedenen Angriffsmethoden auseinanderzusetzen. Sobald Webanwendungen Interaktionsfunktionen für den Benutzer bereit halten, können Sicherheitslücken entstehen, die zur Entwendung von personenbezogenen Daten bis hin zur Zerstörung dieser Daten in den Datenbanken führen können.

Es gibt keine Methoden, um Webanwendungen hundertprozentig vor diesen Angriffen zu schützen, da immer wieder neue Sicherheitslücken entdeckt und ausgenutzt werden. Auch Hacker erfinden immer wieder neue Methoden, um sich Zugang zu den Daten einer Webanwendung zu beschaffen. Bereitet man sich jedoch auf diese Angriffe vor, kann man den Hackern die Arbeit erschweren, sodass ein Angriff schwieriger und zeitaufwändiger wird.

In diesem Kapitel sollen die gängigsten Angriffsszenarien auf eine Webanwendung vorgestellt werden. Dabei wird auf die potentiell kritischen Auswirkungen in Hinsicht auf ein E-Learning System eingegangen.

### 5.3.1. SQL- und NoSQL-Injection

E-Learning Systeme sind in vielen Bereichen auf die Verwendung von Datenbanken angewiesen. Sie werden beispielsweise zur Verwaltung und Speicherung von Lerninhalten und Benutzern des Systems verwendet. SQL- und NoSQL-Datenbanken stellen eine Reihe von Befehlen zur Verfügung, um Datensätze zu manipulieren (hinzufügen, bearbeiten und löschen) und abzufragen.

Eine SQL- und NoSQL-Injection ermöglicht es dem Angreifer eigenen Programmiercode in eine Datenbankabfrage einzuschleusen, der in der Datenbank ausgeführt wird [Owa]. Werden Benutzereingaben in einer Webanwendung nicht ausreichend validiert, ist die Datenbank verwundbar.

Durch SQL- und NoSQL-Injections können eine Vielzahl von Schäden in einem E-Learning System verursacht werden [Wikh]:

- Veränderung von Daten.
- Ausspähen von Daten.
- Einschleusen von beliebigem Code.

- Erlangen von Administrationsrechten.

Wie eine SQL- und NoSQL-Injection aussehen kann, soll folgendes Beispiel zeigen. In beiden Fällen bekommt der Angreifer alle Benutzer zurück, die Rechte eines Administrators besitzen, ohne das Passwort zu kennen.

### SQL-Injection

Zunächst wird angenommen, dass folgender SQL-Befehl zur Authentifizierung des Benutzers mit Benutzername und Passwort genutzt wird:

```
1 SELECT * FROM accounts WHERE username = '$username'
2     AND password = '$password'
```

Listing 5.1: SQL-Befehl zur Authentifizierung des Benutzers

Wird dieser Befehl nicht ausreichend überprüft, kann ein Angreifer durch den Befehl `admin' --` im Feld `username` Zugriff auf alle Administratoren erlangen, indem die Bedingung für das Passwort umgangen wird. Der manipulierte SQL-Befehl würde wie folgt aussehen:

```
1 SELECT * FROM accounts WHERE username = 'admin' --
2     AND password = ''
```

Listing 5.2: Manipulierter SQL-Befehl

### NoSQL-Injection

Das Äquivalent des SQL-Befehls als NoSQL-Befehl einer MongoDB-Datenbank wird wie folgt formuliert:

```
1 db.accounts.find({username: username, password: password});
```

Listing 5.3: NoSQL-Befehl zur Authentifizierung des Benutzers

In MongoDB erhält man mit dem `$gt`-Selektor Dokumente zurück, in denen der Wert des Feldes größer ist als der festgelegte Wert. Der folgende Befehl vergleicht die Größe des Passworts mit einem leeren String, was immer `true` zurückliefert:

```
1 {
2   "username": "admin",
3   "password": {"$gt": ""}
4 }
```

Listing 5.4: Manipulierter NoSQL-Befehl

Um SQL- und NoSQL-Angriffe zu verhindern, müssen Maßnahmen in der Webanwendungen und nicht in der Datenbank getroffen werden. In der Webanwendung sollte eine Validierung der Eingabedaten durchgeführt werden. Darüber hinaus sollte durch eine geeignete Verwaltung der Benutzerrechte sichergestellt werden, dass nicht alle Methoden zur Manipulation der Datensätze von allen Benutzern des Systems genutzt werden können.

Hat ein Angreifer durch eine SQL- oder NoSQL-Injection den Zugriff auf eine Datenbank eines E-Learning Systems erhalten, kann dieser Lerninhalte verfälschen oder löschen. Werden in der Datenbank sensible Daten wie Authentifizierungsdaten, Kreditkartennummern oder Accountdaten gespeichert, können diese vom Angreifer entwendet werden.

### **5.3.2. Cross-Site-Scripting**

Angriffe mit Cross-Site-Scripting (XSS) nutzen Sicherheitslücken in Webanwendungen aus, indem ein schadhafter Code (in der Regel im Browser ausführbare Skripte wie JavaScript) eingeschleust wird, um Sessions von Benutzern zu übernehmen, Daten zu entwenden oder Seiteninhalte zu verändern [Bsi13a]. Sicherheitslücken, die von XSS-Angriffen ausgenutzt werden, entstehen meist durch Fehler in der Programmierung von Webanwendungen, indem nicht vertrauenswürdige Daten entgegen genommen und an einen Webbrowser gesendet werden, ohne diese entsprechend zu validieren [Webb]. Der schadhafte Code wird schließlich beim Anwender ausgeführt.

Es werden grundsätzlich drei Angriffsarten von Cross-Site-Scripting unterschieden: Reflektiertes, persistentes (beständiges) und DOM-basiertes (lokales) XSS.

#### **Reflektiertes XSS**

Beim reflektierten XSS wird ein schadhafter Code in die Variablen und Parametern einer URL eingefügt. Klickt ein potentieller zu schädigender Benutzer auf die präparierte URL, wird der schadhafte Code vom Server der Webanwendung übernommen und im Webbrowser des Opfers ausgeführt. Das kann der Angreifer erreichen, indem die präparierte URL über eine E-Mail an das Opfer gesendet wird. [Bsi13a]

Ein Angriff kann wie folgt ablaufen: Die vom Angreifer präparierte URL wird von einem Opfer angeklickt und somit als Anfrage an den Server der Webanwendung gesendet. Dadurch wird der dynamisch generierte HTML-Code der Weban-

wendung verändert. Das Opfer sieht die manipulierte Webseite im Webbrowser. [Glo]

### Persistentes XSS

Auch das persistente XSS läuft serverseitig ab. Die Veränderungen wird ebenfalls vom Server vorgenommen und der schadhafte Code an den Webbrowser zurück gesendet [Glo]. Es unterscheidet sich dadurch vom reflektierten XSS, dass der schadhafte Code auf dem Server der Webanwendung gespeichert wird und damit dauerhaft auf der Webseite eingebunden ist [Webb].

Das erreicht der Angreifer dadurch, indem der schadhafte Code in einem Kommentar auf einem Blog oder in einem Eintrag im Gästebuch gespeichert wird. Beim Aufruf des manipulierten Eintrages wird der darin enthaltene schadhafte Code des Angreifers im Webbrowser des Opfers ausgeführt. [Mül10a] [Webb]

### DOM-basiertes XSS

Im Gegensatz zu den reflektierten und persistenten XSS spielt sich der DOM-basierte XSS-Angriff nur im Webbrowser des Benutzers ab, sodass der Server der Webanwendung nicht beteiligt ist. Der schadhafte Code wird zur Ausführung direkt einem clientseitigem Skript übergeben. [Eil15]

Wird über JavaScript in der URL ein Argumentwert zur Ausgabe von Daten verwendet, ohne diesen ausreichend zu validieren, kann ein Angriff wie folgt ablaufen: In einem clientseitigen Skript wird ein Argumentwert aus der URL ausgelesen: `http://beispiel.com/index.html?name=Max`. Beim Aufruf der Webseite wird der Benutzer nun mit seinem Namen begrüßt.

```
1 Hallo Max, Willkommen auf unserer Webseite ...
```

Listing 5.5: Normaler Aufruf der Webseite

Wird die URL wie folgt manipuliert: `http://beispiel.com/index.html?name=<script>alert('XSS')</script>`, würde der durch das clientseitige Skript erstellte HTML-Code eine Alertbox öffnen.

Neben dem sauberen Programmieren sollte sichergestellt werden, dass alle Daten, die in jeder Form von außen manipuliert werden können, auf ihre Richtigkeit überprüft werden. Das gilt sowohl für Eingabefelder in Formularen als auch für Variablen und Parameter in einer URL. Entsprechend sollten keine ungefilterten

Strings weiterverarbeitet werden. Um diese abzusichern, ist es nötig, Metazeichen durch entsprechende Zeichenreferenzen zu ersetzen, damit diese als normale Zeichen behandelt werden. [Webb] [Glo]

Cross-Site-Scripting bildet die Grundlage für viele andere Angriffe auf eine Webanwendung, wie Session Hijacking und Session Fixation. So ist es mit XSS-Angriffen möglich, die Session-ID eines Benutzers zu stehlen und vollen Zugriff auf einen fremden Account zu erlangen. Der Angreifer könnte unter einer fremden Identität Lerninhalte bereitstellen oder kostenpflichtige Lerninhalte erwerben. Für ein E-Learning System würde es einen großen Imageverlust bedeuten, wenn bekannt werden würde, dass durch Sicherheitslücken XSS-Angriffe auf Benutzer möglich wären.

### 5.3.3. Session Hijacking und Session Fixation

Beim Session Hijacking wird die Session-ID eines Benutzers entwendet und sich damit unberechtigten Zugang zu bestimmten Bereichen der Webanwendung sowie Zugriff auf die Daten des Opfers verschafft. Auf diese Weise wird nach der Anmeldung mit der geklauten Session-ID dem Server vorgetäuscht, dass es sich beim Angreifer und beim Opfer um die gleiche Person handelt. Die Session-ID kann u.a. durch passives "Abhören" entwendet werden. Dazu verschafft sich der Angreifer Zugang zum Netzkabel bzw. zum WLAN des Opfers oder leitet die Kommunikation über sich als Zwischenstelle um, um alle für den Angriff nötigen Informationen zu sammeln. Auch Cross-Site-Scripting-Angriffe können verwendet werden, um an die Session-ID eines Benutzers zu gelangen. [Webe]

Bei Session Fixation ist der Vorgang ähnlich. Die Session-ID des Opfers wird durch eine gewünschte Session-ID ausgetauscht, die der Angreifer bei sich selbst setzt. Nach der Anmeldung des Opfers gilt die gefälschte Session-ID des Opfers und des Angreifers für den Server als dieselbe. Auch hier erlangt der Angreifer vollen Zugriff auf alle Daten des Opfers und unberechtigten Zugriff auf bestimmte Bereiche der Webanwendung. Um die Session-ID auszutauschen, gibt es mehrere Möglichkeiten: Sie kann dem Opfer beispielsweise durch eine präparierte URL untergeschoben werden. Voraussetzung dafür ist, dass die Webanwendung Session-IDs per Parameter akzeptieren muss. Eine weitere Möglichkeit wäre, wenn eine Sicherheitslücke in der Webanwendung vorliegen sollte, die es erlaubt, Session-IDs durch einen Cross-Site-Scripting-Angriff unterzuschieben. [Webd] [Mül10b]

Es sind folgende Maßnahmen möglich, um gegen Session Hijacking und Session Fixation vorzugehen [Webe] [Webd]:

- Die Übertragung der Daten zum Server sollte über das HTTPS-Protokoll verschlüsselt werden. Der Angreifer muss, um an die Session-ID zu gelangen, die Verschlüsselung entziffern. Dies kann für den Angreifer aufwendig bis unmöglich sein.
- Sicherheitslücken in der Webanwendung sollten geschlossen werden, um Cross-Site-Scripting Angriffe zu unterbinden. Dadurch wird die Gefahr reduziert, dass ein fremder JavaScript Code in die Webanwendung eingeschleust wird und Session-Cookies ausgelesen werden.
- Die Session-ID sollte nicht über die URL übermittelt werden.
- Nach jedem Einloggen sollten Session-IDs neu generiert und zugewiesen werden. Dadurch wird die alte Session-ID für den Angreifer unbrauchbar.
- Sessions sollten durch ein Timeout nach einer bestimmten Zeit auslaufen.

Gelingt es einem Angreifer, die Session-ID eines Benutzers eines E-Learning System zu stehlen, hat dieser vollen Zugriff auf einen fremden Account. Er kann unter der falschen Identität frei agieren und somit unter fremdem Namen Lerninhalte bereitstellen oder kostenpflichtige Lerninhalte erwerben. Es könnte dem Ruf des geklauten Benutzers schaden, sollten fehlerhafte Lerninhalte durch den Angreifer erstellt werden. Erlangt der Angreifer Zugriff auf den Account eines Administrators, hat dieser sogar Zugang zu allen Lerninhalten und Benutzern des Systems und damit die Möglichkeit, sensible Daten zu entwenden.

### 5.3.4. Cross-Site-Request-Forgery

Bei Cross-Site-Request-Forgery, kurz CSRF oder XSRF, wird eine HTTP-Anfrage von einem Angreifer an einen Benutzer einer Webanwendung gesendet. Die HTTP-Anfrage wird vom Angreifer so konstruiert, dass beim Aufruf dieser die Webanwendung die vom Angreifer gewünschte Aktion ausführt. Ist der Benutzer, der die Anfrage ausführt, bei der Webanwendung angemeldet, wird die HTTP-Anfrage mit den Rechten des Benutzers ausgeführt. [Bsi13d]

Einen typischen CSRF-Angriff stellt die Benutzung von HTML Image-Tags oder JavaScript Image-Objekten dar. Die HTTP-Anfrage wird in das `src`-Attribut

des `<img>`-Tags eingebettet und durch den Webbrowser beim Laden der Seite auf die Inhalte zugegriffen, um diese zu laden: [Weba]

```
1 
```

Listing 5.6: HTTP-Anfrage im HTML-Code

Um eine Webanwendung vor CSRF-Angriffen zu schützen, kann ein geheimes Token eingeführt werden, das nur schwer durch den Angreifer entschlüsselt werden kann. Dieses Token wird bei jedem Seitenaufruf der Webanwendung als Parameter in der URL oder als verstecktes Element in einem Formular übertragen. Bei jeder Anfrage wird vom Server überprüft, ob der übertragene Token mit dem während der Session hinterlegtem Wert übereinstimmt. Dadurch kann ein Angreifer keine gültige HTTP-Anfrage nachstellen. [Bsi13d]

Werden in E-Learning Systemen von Angreifern nachgestellte HTTP-Anfragen ausgeführt, ist die Integrität der Lerninhalte dieser gefährdet. Eine Schwachstelle existiert, wenn fälschlicherweise PUT oder DELETE Befehle auf dem Server ausgeführt werden, sodass der Angreifer Daten in der Datenbank verändern oder löschen kann. Dadurch können Preise oder der Inhalt von Lerninhalten manipuliert werden. Verändert der Angreifer die Zugangsdaten der Benutzer, sodass alle Benutzer vom System ausgesperrt werden, könnten beispielsweise Deadlines von Lernphasen, die für die Klausurzulassung notwendig sind, nicht mehr eingehalten werden.

### 5.3.5. Denial of Service

Mit Denial of Service, oder kurz DoS, wird ein Angriff bezeichnet, der einen Rechner, Server oder ein ganzes Netzwerk in seiner Funktionalität beeinträchtigen und somit Benutzern nur eingeschränkt zur Verfügung stehen soll [Ele]. Dieser versucht zu verhindern, dass diese Systeme genutzt werden können, indem begrenzte Ressourcen wie CPU-Rechenzeit, Arbeitsspeicher, Plattenplatz, Netzbandbreite oder Ähnliches mutwillig überlastet werden [Bsi11b]. Wird die Überlastung durch mehrere Systeme gleichzeitig verursacht, spricht man von Distributed Denial of Service (DDoS) [Wikb].

Bei einem DoS-Angriff werden so viele Anfragen an einen Dienst gesendet, bis die Netzwerkverbindung oder der Dienst selbst überlastet wird. Um das nötige Datenverkehrsaufkommen zu erreichen, werden diese Anfragen in der Regel über

Bot-Netze versendet. Dies hat zur Folge, dass der Dienst durch die Überlastung keine Anfrage mehr beantworten kann. [Bsi11a]

Das Ziel von DoS-Angriffen ist für gewöhnlich nicht der Diebstahl und die Manipulation von Daten, sondern den Ruf eines Unternehmens mit Internetpräsenz zu schädigen, indem ihre Arbeit behindert wird und die Webseiten für die Benutzer nur eingeschränkt nutzbar sind. [Rou13]

Um DoS-Angriffen entgegenzuwirken, sollten einige Schutzmaßnahmen umgesetzt werden [Bsi16]:

- Grenzwerte im System festgelegt, z.B. die vorübergehende Blockierung einer Ressource oder des Benutzerkontos nach wiederholten Fehlzugriffen.
- Zeitspannen zwischen Anfrage und Verarbeitung durch die Webanwendungen künstlich verzögern z.B. bei wiederholter erfolgloser Anmeldung.
- IP-Adressen bei Verdacht auf einen Angriff temporär blockieren.
- CAPTCHAs verwenden.
- Eingaben bei Eingabefeldern in Webanwendung verifizieren, bevor rechenintensive Operationen ausgeführt werden.

Wird durch ein DoS-Angriff der Absturz eines E-Learning Systems verursacht, kann dieser für einen gewissen Zeitraum von den Benutzer nicht mehr verwendet werden. Folglich können keine Lerninhalte mehr erstellt oder gelernt werden und der Ruf des Betreibers würde unter dem Ausfall des Systems leiden.

## 6. Die Open-Source-Anwendung THMcards

Dieses Kapitel ist in Zusammenarbeit mit Marius Trautrimis entstanden. Nachfolgend wird auf die Re-Implementierung von THMcards eingegangen und die technischen Aspekte von THMcards erläutert. In der Re-Implementierung findet die Evaluierung eines geeigneten Web-Frameworks statt. Zudem werden die Anpassungen in der Codestruktur dargestellt. Die in THMcards verwendeten Technologien werden im Kapitel 6.3 *Technische Aspekte* beschrieben.

### 6.1. Zieldefinition

Aus den folgenden Ideen und Anforderungen von Prof. Dr. Klaus Quibeldey-Cirkel [QC13] ist die E-Learning Plattform THMcards entstanden:

„Gegen das Aufschieben des Lernens bis zum Klausurtermin und das Vergessen kurz danach soll Leitners Lernsystem in digitalisierter Form in die Vorlesungen gebracht werden. Leitners Lernkartei ist so organisiert, dass sie dem Anspruch der Lernmethode ‚Distributed Practice‘ bestens genügt und effektiv und nachhaltig dem Vergessen des Gelernten entgegenwirkt.

Der Fünf-Fächer-Algorithmus nach Leitner sieht vor, gelernte Inhalte in Form von Lernkarten (Vorderseite: Lernfrage, Rückseite: richtige Antwort) in bestimmten Zeitintervallen zu wiederholen: Neue Lernkarten sind im vordersten Fach des Karteikastens und wandern bei richtiger Beantwortung ins zweite Fach. Die Lernkarten dort werden alle zwei bis drei Tage zum wiederholten Lernen vorgelegt. Bei richtiger Beantwortung wandert die Lernkarte ins dritte Fach, dessen Karten alle zehn Tage wiederholt werden. Lernfragen im vierten Fach werden monatlich wiederholt und im fünften Fach nach drei Monaten. Da-

nach bleiben sie nachhaltig im Langzeitgedächtnis haften. Bei falscher Beantwortung wandert jede Lernkarte, egal aus welchem Fach, zurück ins erste. Der Fünf-Fächer-Algorithmus kompensiert quasi den Verlauf der Vergessenskurve.

Eine digitale Neuauflage der Lernkartei bietet viele weitere Vorteile: multimediale Lernkarten mit eingebetteten Video- und Audiodateien, komplexe Formeln mit TeX-Formatierung, Lernstand-Berechnung und mehr. [...]

Im Wintersemester 2012/13 habe ich die Online-Lernkarten-Software CoboCards in zwei Kursen mit jeweils ca. 75 Studierenden parallel zum Vorlesungsbetrieb eingesetzt und von den Teilnehmenden evaluieren lassen. Die Quintessenz: Das Online-Lernen mit Lernkarten wurde allgemein begrüßt und als effektiv eingeschätzt; [...]. Allerdings wurden die Kollaborationsfunktionen von CoboCards, das Erstellen und Abfragen von Lernkarten online im Team, und auch die sozialen Funktionen, wie Team-Pinnwand und Diskussionsforum, kaum genutzt. Außerdem fehlte die intrinsische Motivation, Lernkarten von sich aus zu nutzen. Nur eine extrinsische Motivation in Form von Bonuspunkten auf die Klausurnote für besonders gut formulierte Antworten auf vom Dozenten gestellte Lernfragen erhöhte die Nutzung von CoboCards. Einstimmig gewünscht waren offizielle, das heißt vom Lehrenden erstellte Lernkarten. Hier setzt das Neuartige meiner geplanten Lehrinnovation an:

Es soll eine Online-Plattform THMcards entwickelt werden, auf der Lehrende Lernkarten zu ihren Modulen erstellen und aktualisieren können. Studierende können auf die Lernkarten ihrer Kurse per Smartphone, Tablet und Laptop zugreifen. Der individuelle Lernstand wird automatisch ermittelt und angezeigt. Die Lernkarten-Funktionalität der Plattform wird in das mobile TED-System ARSnova integriert; [...]. Vorbereitungs-, Vorlesungs- und Konzeptfragen, die mit ARSnova in verschiedenen Frageformaten (Single/Multiple Choice, Likert- und Noten-Skalen, Freitext) erstellt wurden, können als Lernkarten nach THMcards exportiert werden.

Meine geplante Lehrinnovation will die mediengestützten Lehr- und Lernmethoden der Hörsaal-Didaktik um das Leitnersche Lernsystem

erweitern: Neben ‚Peer Instruction‘ und ‚Just-in-Time Teaching‘, die mithilfe des TED-Systems ARSnova in den letzten beiden Semestern in mehreren Vorlesungen erfolgreich praktiziert wurden, soll in den kommenden Semestern die Lernkartei-Plattform THMcards eingesetzt werden. Ich verspreche mir nicht nur eine messbare Verbesserung der Behaltensquote über einen langen Zeitraum. Neben dem Lernen im Hörsaal soll auch ein verteiltes Lernen der Vorlesungsinhalte per Lernkarten in informellen Lernsituationen überall und jederzeit ermöglicht werden. ‚Mobile Learning‘ postuliert das Lernen in kleinen Portionen (‚Lernhäppchen‘); Lernkarten erfüllen genau diese Anforderung.

Es bleibt das Risiko der Motivation zur Nutzung der Lernkarten. Hier kommt die Gamifizierung des Lehrens und Lernens ins Spiel. Laut dem ‚NMC Horizon Report: 2013 Higher Education Edition‘ zählt Gamifizierung zu den Schlüsseltechnologien im Bildungsbereich und wird in den kommenden zwei bis drei Jahren bedeutende Auswirkungen auf die Hochschullehre haben. [...]

Zusammengefasst soll mit THMcards eine Lernplattform geschaffen werden, die ihrem Namen gerecht wird. Im Gegensatz zu Moodle, ILIAS, OLAT oder Stud.IP, die in aller Regel nicht als Plattform zur Unterstützung von Lehr- und Lernprozessen auf dem Campus und unterwegs eingesetzt werden, sondern nur für die Organisation der Kurse und Verteilung der Vorlesungsmaterialien, soll das Lehren und Lernen tatsächlich auf der Plattform THMcards und mit ARSnova auch im Hörsaal und unterwegs unterstützt werden.“

Aufbauend auf diesem Zitat ist die erste Version von THMcards durch die Studenten Jan Christopher Kammer und Daniel Knapp entstanden. Dazu können auch die Masterarbeiten *Implementierung von Spaced Repetition Algorithmen zur effektiven Abfrage von Lernkarten innerhalb der eLearning Plattform THMcards* [Kam14] und *Implementierung von Spielmechaniken zur Steigerung der Lernmotivation von Studierenden am Beispiel der Lernkarten Plattform THMcards* [Kna13] eingesehen werden.

Im Rahmen des Entwicklungsprojekts galt es die erste Version von THMcards in ihrem Aufbau zu optimieren. Hierzu sollte ein Redesign und eine Re-

Implementierung der bestehenden Use Cases mit einem für THMcards geeigneterem Web-Framework realisiert werden.

### 6.2. Re-Implementierung von THMcards

Die Gründe für die Re-Implementierung von THMcards sind hauptsächlich auf eine mangelhafte Softwarequalität zurückzuführen. Sowohl auf die Verständlichkeit als auch auf die Wartbarkeit der Software wurde hierbei keine Rücksicht genommen. Durch die Nichteinhaltung vorgegebener Programmierparadigmen bei der Erstellung von Programmcode und der daraus resultierenden unübersichtlichen Codestruktur, war die Verständlichkeit der Anwendung nicht mehr gegeben. Weiterhin wurde mit Backbone ein JavaScript-Framework gewählt, das an Bekanntheit und somit an Support verloren hat. Eine dauerhafte Wartung von THMcards wäre aufgrund des zunehmenden Projektumfangs durch die anstehenden Erweiterungen nicht ohne größeren Energieaufwand möglich. Die vorhandene Grundlage war somit für eine erfolgreiche Weiterentwicklung von THMcards nicht geeignet.

In den folgenden Abschnitten werden die getroffenen Entscheidungen zur Re-Implementierung von THMcards erläutert. Diesbezüglich wird mit Hilfe einer Evaluierung ein geeignetes Web-Framework für das Projekt THMcards ermittelt. Darüber hinaus wird auf die vorgenommenen Änderungen der Codestruktur eingegangen.

#### 6.2.1. Evaluierung eines geeigneten Web-Frameworks

Um ein geeignetes Web-Framework für das Projekt THMcards zu finden, wurde im Rahmen des Entwicklungsprojekts die Evaluierung der populärsten MV\* Frameworks für Web-Anwendungen durchgeführt. Zu diesen gehören Sencha Touch bzw. Ext JS, Angular, Backbone, Ember und Meteor.

Bevor der eigentliche Vergleich durchgeführt wurde, konnten zwei dieser Frameworks bereits ausgeschlossen werden. Da die Web-Applikation THMcards primär als Desktop-Anwendung genutzt werden soll, ist das Framework Sencha Touch, das speziell für Mobile-Web entwickelt wurde, für das Projekt nicht geeignet. Darüber hinaus war es in diesem Vergleich nicht notwendig auf das proprietäre Ext JS einzugehen, da bei der Entwicklung von THMcards ausschließlich Open-Source Produkte zum Einsatz kommen sollten.

Bei der Evaluierung spielten folgende Kriterien eine wichtige Rolle: Community, Framework Size und Templating.

### Community

Die Community ist eine der wichtigsten Faktoren bei der Auswahl eines geeigneten Frameworks. Bei einer großen Community stehen dem Benutzer mehr Tutorials, Third-Party Module sowie generelle Hilfestellungen zur Verfügung. In der folgenden Tabelle wird die Größe der Community der einzelnen Frameworks gegenüber gestellt:

Metrik	Angular	Backbone	Ember	Meteor
GitHub Stars	52k	25,5k	16,8k	35,3k
GitHub Contributors	1,5k	290	614	310
GitHub Commits	8k	3,3k	13,3k	17,5k
Third-Party Module	2k	275	2,8k	11,4k
Stack Overflow Fragen	196,7k	20k	19,7k	23,3k
YouTube Ergebnisse	181k	29,8k	28,1k	33,9k

Tabelle 6.1.: Größe der Framework Community (Stand 12.09.2016)

In Tabelle 6.1 ist zu erkennen das Angular die größte Community hat. Während die Frameworks Ember und Backbone an Popularität verlieren, konnte Meteor und Angular ein starkes Wachstum verzeichnen. Außerdem ist der Abbildung 6.1 zu entnehmen, dass hinter Meteor eine große Community steht, die stetig damit bemüht ist das Framework weiter zu entwickeln.

### Framework Size

Die Ladezeiten bzw. der Seitenaufbau sind Fundamental für den Erfolg einer Webseite. Für den im Allgemeinen ungeduldigen Benutzer ist es wichtig, im Web schnellstmöglich an Informationen zu gelangen. Aus diesem Grund sind lange Ladezeiten auf Webseiten zu verhindern. Diesbezüglich sind bei der Wahl des richtigen Frameworks zwei ausschlaggebende Faktoren zu beachten: Die Frameworkgröße und die Zeit, die benötigt wird, um das Framework zu laden.

Im Allgemeinen werden JavaScript-Dokumente minifiziert und komprimiert dem Benutzer bereitgestellt. Aus diesem Grund wird in dieser Gegenüberstellung die Dateigröße der minifizierten und komprimierten Versionen verglichen. Da einige Frameworks nicht eigenständig nutzbar sind, sollte die Dateigröße der benötigten Dependencies ebenfalls bei diesem Vergleich berücksichtigt werden.

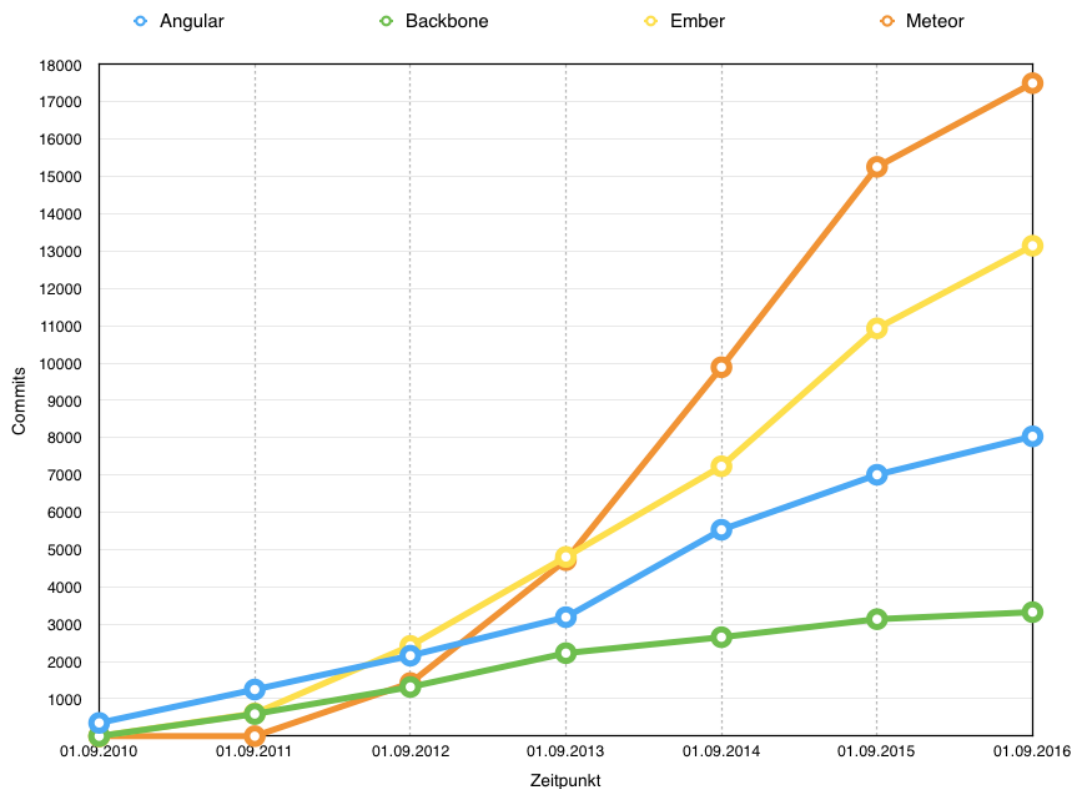


Abbildung 6.1.: Die Commits eines Web-Frameworks

Der Tabelle 6.2 ist zu entnehmen, dass es sich bei Backbone um das Framework mit der kleinsten Dateigröße handelt. Berücksichtigt man jedoch alle benötigten Dependencies, wird der Abstand zwischen Backbone und Angular deutlich kleiner, da Angular eigenständig nutzbar ist.

### Templating

Angular und Ember sind bereits mit ihrer eigenen Template Engine ausgestattet. Backbone und Meteor hingegen ermöglichen die Auswahl unterschiedlicher Template Engines. Die Template Engine von Angular setzt sich aus einfachem HTML und Binding Expressions zusammen, während bei Ember Handlebars zum Einsatz kommt. In Backbone wird meistens Underscore genutzt, da dieses zu den benötigten Dependencies von Backbone gehört. Durch den einfachen Aufbau von Underscore hat der Nutzer die Möglichkeit JavaScript in den HTML Code einzubinden. Meteor hingegen nutzt üblicherweise Spacebars, eine an Handlebars angelehnte Template Engine, ermöglicht jedoch auch die Verwendung von Jade.

Framework	Version	Größe	Dependencies	Gesamtgröße
Angular	1.5.8	55kb	-	55kb
Backbone	1.3.3	7,4kb	jQuery, Underscore	42,1kb
Ember	2.8.0	108kb	jQuery, Handlebars	156kb
Meteor	1.4	-	Mongo, Blaze, jQuery	135kb

Tabelle 6.2.: Größe der einzelnen Frameworks (Stand 12.09.2016)

## Angular

*Vorteile:* Angular bietet zahlreiche innovative Konzepte für Webentwickler. Unter anderem lässt sich durch *two-way data binding* ein großer Teil des Boilerplate-Code einsparen. Der erforderliche JavaScript-Code muss nicht selbst implementiert werden, sondern lässt sich auf einfachste Weise im HTML-Template durch spezifische Attribute und Expressions einbinden.

Von den Frameworks bietet Angular die größte Community und am meisten Online-Content. Darüber hinaus wird es von Google unterstützt und gefördert. Bei der Weiterentwicklung von Angular spielt die Community eine wichtige Rolle. Design-Entscheidungen werden von dem Angular-Team gemeinsam mit der Community getroffen, indem Anmerkungen direkt im veröffentlichten Design-Dokument geäußert werden können.

Des Weiteren ermöglicht Angular die eigene Applikation in Kategorien einzuteilen. Dazu werden verschiedene Typen zur Verfügung gestellt: Controllers, Directives, Filters, Services und Views (Templates).

Angular setzt viel Wert auf die Möglichkeit die Anwendung zu testen. Dies wird durch die Bereitstellung von Mock-Objekten, Unit Isolation und die Vorteile des Dependency Injection Mechanismus ermöglicht.

*Nachteile:* Für seine Komplexität der Directives API ist Angular oft kritisiert worden. Es nimmt einige Zeit in Anspruch die zur Verfügung gestellten Directives zu beherrschen.

Angular Expressions werden oftmals zu exzessiv im View-Template einer Anwendung verwendet. Die daraus resultierende komplexe Logik führt dazu, dass sich die Anwendung schwer bis gar nicht testen lässt.

### Backbone

*Vorteile:* Backbone ist leichtgewichtig, schnell und beansprucht den Speicher nur gering. Die Lernkurve ist sehr linear, da es nur wenige Konzepte (Models/Collections, Views, Routes) zu verstehen gibt. Die Dokumentation des Codes ist einfach gehalten und ausführlich beschrieben.

Durch den einfachen Aufbau können eigene Frameworks mit Backbone kombiniert werden. Außerdem besteht die Möglichkeit bereits vorhandene Frameworks, wie z.B. Marionette oder Backbone UI zu verwenden.

*Nachteile:* Das größte Problem von Backbone ist, dass es keine Struktur vorschreibt. Jedem Entwickler ist freigestellt, wie die Struktur der Applikation aufgebaut werden soll. Dadurch kann es für externe Entwickler viel Zeit in Anspruch nehmen den Aufbau einer bestehenden Anwendung zu verstehen.

Der einfache Aufbau von Backbone bringt auch negative Aspekte mit sich. Um komplexe Anwendungen mit Backbone zu erstellen ist man gezwungen weitere Plugins einzubinden. Viele Funktionalitäten werden vom Backbone-Framework allein nicht abgedeckt.

Zudem unterstützt das Backbone-Framework kein *two-way data binding*, sodass Entwickler gezwungen sind viele Standardformulierungen im Code selbst zu implementieren, um bei Änderungen im Model den View bzw. bei Änderungen im View das Model zu aktualisieren.

Ein weiteres Problem von Backbone ist, dass es das Document Object Model (DOM) direkt manipuliert und somit die Durchführung der Unit-Tests erschwert.

### Ember

*Vorteile:* Genau wie in Angular kann bei Ember ein großer Teil des Boilerplate-Codes eingespart werden, da viele Funktionalitäten durch die Konfiguration von Ember abgedeckt werden.

Im Gegensatz zu den anderen beiden Frameworks, die mit einem minimalen Data-Layer ausgestattet sind, bietet Ember ein voll entwickeltes Daten-Modul welches hervorragend mit RESTful JSON APIs harmoniert.

Darüber hinaus spielt bei Ember die Performance eine wichtige Rolle. Konzepte wie *The Run Loop* helfen dabei schnelle Ladezeiten der Applikation zu ermöglichen.

*Nachteile:* Aufgrund vieler grundlegender Änderungen in der Ember-API gibt es viele veraltete Informationen und Beispiele die nicht mehr mit aktuellen Versionen des Frameworks kompatibel sind. Das führt bei Anfängern zu einem erschwerten Einstieg.

Die aktuell noch oft verwendete Handlebars Template Engine, bei der im DOM viele `<script>`-Tags eingebunden werden, führt schnell zu einem unübersichtlichen DOM-Tree. Darüber hinaus kann es vorkommen, dass das CSS-Styling nicht mehr richtig dargestellt wird oder andere Frameworks nicht mehr funktionieren.

### **Meteor**

*Vorteile:* Der wesentliche Vorteil von Meteor ist, dass man nur eine Applikation für Client und Server entwickeln muss. Damit eignet sich Meteor vor allem für Einsteiger, da man sich keine Gedanken über die Kommunikation zwischen Client und Server machen muss.

Ein weiteres nützliches Feature ist die sofortige Aktualisierung der Anwendung im Browser, nachdem Änderungen im Quellcode oder der Datenbank vorgenommen wurden.

Mit der Unterstützung von *Hard Code Push* garantiert Meteor Continuous Deployment. Dadurch ist es möglich JavaScript im Quellcode zu ändern ohne den Benutzer bei seinen Aktionen auf der Webseite zu unterbrechen.

Ein weiterer Vorteil ist, dass neben der vorgegebenen Blaze Bibliothek auch Angular oder React als User Interface Rendering Library genutzt werden können. Dies ermöglicht je nach Art der Anwendung die geeignetste Bibliothek für Projekt auszuwählen.

*Nachteile:* Meteor eignet sich nicht als Framework, wenn hohe Ansprüche an die Performance gestellt werden. So sollten Spiele-Applikationen mit hohen Grafikanforderungen nicht mit Meteor, sondern besser als native App programmiert werden.

Darüber hinaus ist ein Großteil der für Meteor angebotenen Plugins veraltet, da sie nicht von Meteor selbst, sondern von Community-Mitgliedern erstellt worden sind. Oftmals handelt es sich bei dem Angebot um npm-Module, die mittels eines Wrappers zu einem Meteor-Plugin umfunktioniert wurden. Während die npm-Module kontinuierlich weiterentwickelt werden, ist eine Aktualisierung der Meteor-Plugins durch die Community meist nicht gegeben. Jedoch wurde die-

ses Problem durch die direkte Integration von npm in die Meteor-Anwendung relativiert.

### Fazit

Aufgrund des minimalistischen Aufbaus von Backbone und dem daraus resultierenden Einsatz von weiteren Frameworks, ist die Struktur jeder Backbone-Anwendung verschieden. Das gestaltet die fortlaufende Entwicklung der Anwendung schwieriger, da eine große Einarbeitungszeit für neue Entwickler in ein Projekt notwendig ist. Da es sich bei THMcards um ein fortlaufendes Projekt, bei dem zukünftig weitere Studenten beteiligt sind, handelt, ist das Backbone-Framework für dieses Projekt nicht geeignet.

Ember und Angular führen beide für das Projekt ausreichend Vorteile mit sich. Was jedoch eher für die Entscheidung zugunsten von Angular spricht, ist die ausführliche Dokumentation und die mit Abstand größte Community, sowie die Unterstützung von Google.

Aufgrund der hohen Komplexität des Framework Angular und der damit resultierenden langsamen Lernkurve, fällt die Entscheidung jedoch zugunsten des in der Evaluierung verbliebenen Frameworks Meteor. Meteor bietet ein gutes Gesamtpaket, eine einfache Handhabung und lässt sich ohne großen Aufwand mit weiteren Plugins erweitern. Es ist somit für Projekte wie THMcards bestens geeignet.

### 6.2.2. Codestruktur

Bei der ersten Version von THMcards der Masterstudenten Jan Christopher Kammer und Daniel Knapp handelte es sich um eine Anwendung die mit den Frameworks Backbone, Marionette und Node.js, sowie der Datenbank CouchDB erstellt wurde. Für das ausschließlich in JavaScript geschriebene THMcards wurde Node.js im Zusammenspiel mit der CouchDB für die serverseitigen Aufgaben verwendet. Im Front-End kamen die Frameworks Backbone und Marionette zum Einsatz. Da Backbone standardmäßig für häufig auftretende Szenarien keine Funktionalitäten bereithält und Boilerplate-Code vermieden werden sollte, ist der Einsatz von Marionette notwendig gewesen.

Im Laufe der Projektphase gab es immer häufiger auftretende Verständnisprobleme aufgrund essentieller Designfehler in der Codebasis. Der größte Teil der

Anwendungslogik wurde innerhalb von zwei Dateien realisiert, wodurch die Einarbeitung erschwert wurde und die Weiterentwicklung von THMcards unmöglich machte. Ebenfalls zu kritisieren waren die vielen Redundanzen in den ohnehin schon unübersichtlichen Dateien. Zudem ist nicht auf die Trennung von Template und Styles geachtet worden. Viele der CSS-Deklarationen sind direkt innerhalb der HTML-Dokumente vorgenommen worden. Des weiteren wurden große Teile der responsive Funktionen von Bootstrap fehlerhaft eingesetzt oder diese sogar überschrieben.

Um in der Re-Implementierung von THMcards solche Fehler zukünftig zu vermeiden, wird unter anderem die Dateistruktur an die vorgegebenen Richtlinien von Meteor angepasst. Das Beachten der Meteor-Richtlinien ermöglicht nicht nur eine einfache Einarbeitung weiterer Entwickler, sondern ist auch für die Weiterentwicklung der Anwendung hilfreich. Ein weiterer Vorteil dieser Modularisierung ist eine verbesserte Performance der Anwendung, da nur benötigte Module anstatt der gesamten Anwendung geladen werden. Ein Überblick über diese Dateistruktur von THMcards ist in Abbildung 6.2 zu sehen.

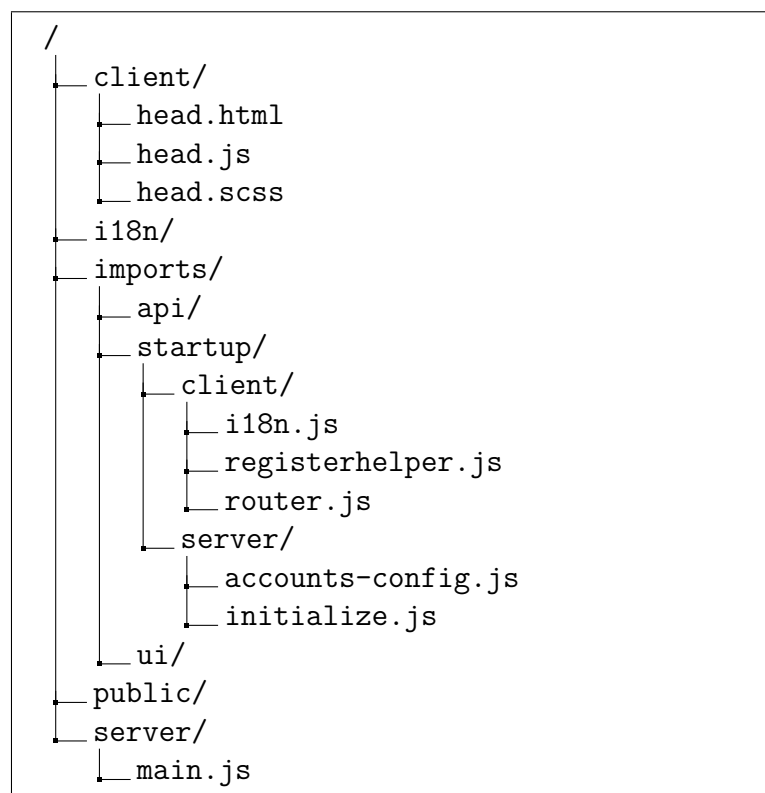


Abbildung 6.2.: Dateistruktur von THMcards

Module die beim Aufruf der Anwendung nicht erforderlich sind, werden im Verzeichnis `imports` abgelegt. Dateien aus `imports` werden nur geladen, sobald diese in einer anderen Datei referenziert wurden. Alle weiteren Dateien außerhalb dieses Verzeichnisses werden nach der von Meteor vorgegeben Reihenfolge aufgerufen. Die Eingangspunkte der Anwendung werden durch `client/head.js` für die Client-Funktionen und `server/main.js` für die Server-Funktionen definiert. Diese Einstiegsdateien sorgen dafür, dass beim Start der Anwendung Startup-Module aus `imports` aufgerufen werden. Da sich alle wichtigen Dateien in `imports` befinden, bietet es sich außerdem an die Anwendungslogik und die grafische Benutzeroberfläche voneinander zu trennen. Hierfür werden die Verzeichnisse `imports/api` und `imports/ui` verwendet.

### 6.3. Technische Aspekte von THMcards

THMcards wurde mithilfe des Open-Source Web-Frameworks Meteor als Single-Page Web Application, oder auch kurz SPA, umgesetzt. Die Besonderheit einer SPA liegt darin, dass die gesamte Anwendung aus einem einzigen HTML-Dokument besteht, dessen benötigte Inhalte dynamisch nachgeladen werden [Brü15]. Dies ermöglicht gerade bei komplexen Anwendungen eine verbesserte User Experience, da im Gegensatz zu klassischen Webanwendungen, die aus mehreren untereinander verlinkten HTML-Dokumenten bestehen, auf das erneute Laden von Unterseiten verzichtet wird. Daraus resultiert bei SPA's eine reaktionsschnelle Benutzeroberfläche, sowie eine erhebliche Kostenersparnis, da nur minimale Datenmengen über das Web transferiert werden. [2BI]

Bei Single-Page Web Applications wird eine klare Schnittstelle zwischen Front-End und Back-End definiert. Alle relevanten Daten einer Webanwendung werden durch klassische Programmiersprachen und Datenbanken verarbeitet und gespeichert. Das Front-End beruht auf HTML, CSS, sowie sehr stark auf JavaScript und dem damit verbundenem JavaScript-Framework. [Sti14] Durch JavaScript stellt das Front-End Anfragen an das Back-End und erhält dadurch die benötigten Daten im JSON-Format zurück. Diese werden an die entsprechenden Stellen gesetzt, sodass ein weiterer Seitenaufruf ausbleibt. [2BI]

Durch den Einsatz des Web-Frameworks Meteor ist THMcards einheitlichen in JavaScript für Client und Server umgesetzt. Hierbei sind alle Funktionalitäten von der Datenbank über den Server bis hin zum Client von Meteor abgedeckt.

Für die Datenbankanbindung wird die in Meteor integrierte MongoDB verwendet, während auf dem Server Node.js zum Einsatz kommt.

### 6.3.1. Meteor

Meteor ist ein Open-Source Web-Framework zur Entwicklung moderner Mobile- und Webanwendungen [Metd]. Diese Anwendungen werden mit Meteor vollständig in JavaScript entwickelt. Folglich wird die Anbindung zur Datenbank, der Server und der Client mit einer einzigen Programmiersprache abgedeckt. Meteor bündelt verschiedene Frameworks zu einer Plattform und integriert diese nahtlos, um dem Benutzer eine einfache Bedienung zu ermöglichen. [Tut15]

Was Meteor so besonders macht und von anderen Web-Frameworks unterscheidet, beschreiben die Entwickler der Plattform mit folgenden Grundsätzen:

- **Data on the Wire:** Meteor sendet kein HTML über das Netzwerk. Es werden lediglich Daten an den Client gesendet, die auf diesem wiederum verarbeitet und dargestellt werden. [NK13]
- **One Language:** Für den Code der Client- und Server-Anwendung wird als einzige Programmiersprache JavaScript verwendet [NK13]. Das ermöglicht nicht nur die schnelle Entwicklung von Applikationen, sondern erfordert auch weniger Umdenken, wenn in verschiedenen Bereichen der Anwendung gearbeitet werden muss [Tur14]. Da es sich bei JavaScript um eine der populärsten Programmiersprachen handelt, gibt es viele Anwendungsfälle und viele Entwickler sind bereits mit der Sprache vertraut [Owe15].
- **Database Everywhere:** Über eine gemeinsame API können sowohl der Client als auch der Server auf die Datenbank zugreifen. [NK13]
- **Latency Compensation:** Durch Vorauswahlen und Modellsimulationen auf dem Client kann eine latenzfreie Datenbankverbindung simuliert werden. [NK13]
- **Full Stack Reactivity:** Meteor erlaubt reaktive Programmierung: Anwendungen werden also in Echtzeit entwickelt. Alle möglichen Schnittstellen einer Anwendung, von der Datenbank bis zu den Templates werden automatisch aktualisiert. Seiten müssen nicht neu geladen werden, um Updates

zu sehen und Änderungen an Dokumenten können ohne Verzögerung gespeichert werden. Das ermöglicht die einfache Zusammenarbeit an einer Anwendung in Echtzeit. [Owe15]

- **Embrace the Ecosystem:** Meteor ist Open-Source und integriert andere Open-Source Frameworks, statt diese zu ersetzen oder nachzubilden. [NK13]
- **Simplicity equals Productivity:** Einfachheit steht bei Meteor an erster Stelle. Mithilfe eines großen Angebots an Packages der Node.js und JavaScript Community wird in Meteor weniger Code benötigt als in anderen gängigen Frameworks wie Angular oder Backbone, um die gleichen Aufgaben zu realisieren [Tut15]. Durch JavaScript in Front-End und Back-End erlaubt Meteor schneller zu programmieren und mit wenig Aufwand ein fertiges Produkt zu entwickeln [Owe15].

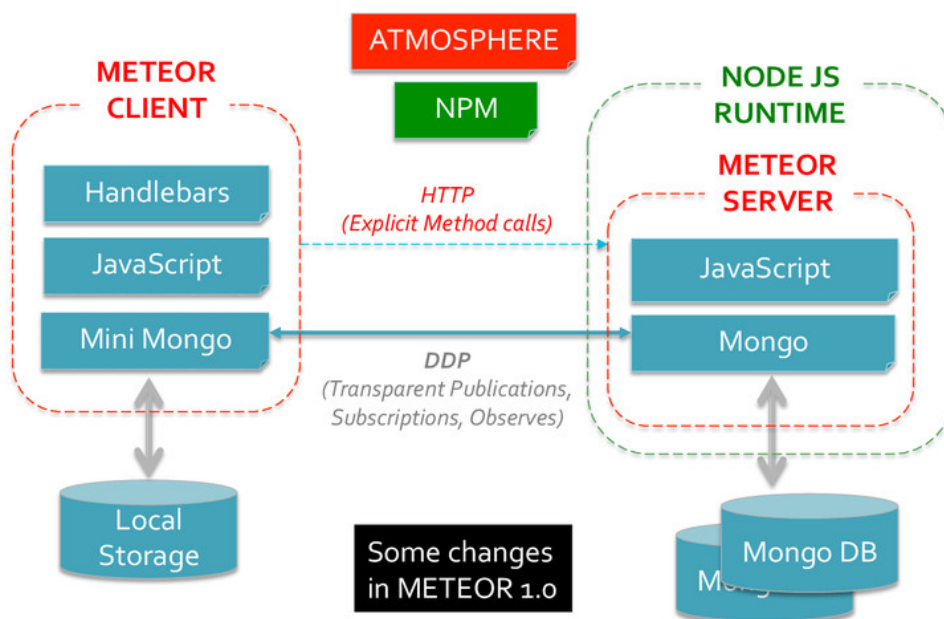


Abbildung 6.3.: Aufbau von Client und Server in Meteor 1.0 [Mon15]

Um Anwendungsdaten über einen langen Zeitraum bereit zu halten verwendet Meteor die Datenbank MongoDB [NK13]. Durch die Implementierung einer an der MongoDB API angelehnten Minimongo Datenbank werden auf dem Client vergleichbare Befehle wie die auf dem Server bereitgestellt. Die Kommunikation und Synchronisation der Daten zwischen Client und Server wird hierbei gänzlich

von Meteor übernommen, sollten Ereignisse über zu lesende oder zu schreibende Änderungen eintreten. Das Zusammenspiel zwischen Client und Server wird nochmal in Abbildung 6.3 veranschaulicht. [Mon15]

Die Daten werden wie in MongoDB üblich in Collections abgebildet, deren Definition einfach gehalten wird. Collections stehen dem Client als auch dem Server zur Verfügung und werden von Meteor automatisch synchronisiert. Jede Collections erhält einen eindeutigen Namen als Argument und wird wie folgt definiert: [NK13]

```
1 export const Cardsets = new Mongo.Collection("cardsets");
```

Listing 6.1: Definition einer Collection

Mit dem Meteor Kommandozeilen-Tool wird eine neue Meteor-Applikation standardmäßig mit einem Autopublish-Verhalten versehen. Das erlaubt jeden Client (auch anonyme, nicht authentifizierte und nicht autorisierte) auf die Collections zuzugreifen und jede beliebige Zugriffsoperation wie das Speichern, Modifizieren, Suchen oder Aggregieren von Daten, auszuführen. Dieses Standardverhalten lässt sich jedoch abschalten, sodass sensible oder rechenintensive Zugriffsoperationen auf dem Server durchgeführt und die aktiven Clients über das Ergebnis benachrichtigt werden. Durch dieses sogenannte *Publish/Subscribe*-Verfahren können mit **publish** die Ergebnisse einer Datenbankabfrage auf dem Server veröffentlicht und mithilfe von **subscribe** die Clients abonniert werden. Auf diese Weise lässt sich verhindern, dass Änderungen einer Collection Einfluss auf alle Clients hat und alle Felder öffentlich sind. [Lie13][NK13]

```
1 Meteor.publish("cardsets", function() {  
2   return Cardsets.find({});  
3 });
```

Listing 6.2: Veröffentlichung der Daten im Server

In der Server-Datei werden die Collections mit **Meteor.publish**, wie in Listing 6.2 dargestellt, veröffentlicht. Der Client abonniert diese mit dem Bezeichner der Collection:

```
1 Meteor.subscribe("cardsets");
```

Listing 6.3: Abonnieren der Daten im Client

Eine weitere Möglichkeit die Logik vom Client auf den Server zu verlagern sind die in Meteor sogenannten *Method Calls*. Mit ihnen lassen sich Funktionen auf

dem Server definieren und auf dem Client aufrufen. Ein Beispiel einer solchen Funktion ist das Entfernen eines Kartensatzes: [NK13]

```
1 Meteor.methods({
2   deleteCardset: function(id) {
3     ...
4     Cardsets.remove(id);
5     ...
6   }
7 });
```

Listing 6.4: Definition einer Funktion im Server

Über `Meteor.call()` können die Server-Funktionen vom Client ohne weitere Konfigurationen aufgerufen werden. Eine solche Definition veranschaulicht Listing 6.5.

```
1 Meteor.call("deleteCardset", id);
```

Listing 6.5: Method Call im Client

Die bereitgestellten Modelle und Daten werden mit Templates in Verbindung mit dem MVVM-Entwurfsmuster (Model, View, ViewModel) in die Views implementiert. Hier kommt in Meteor die Template-Engine Handlebars zum Einsatz, jedoch besteht auch die Möglichkeit andere Template-Engines einzubinden. Handlebar-Templates werden mit den Tags `<template name="...">...</template>` innerhalb eines HTML-Dokuments definiert. Diese Templates lassen sich mit der Anweisung `{{> name}}` mit ihren Namen an verschiedenen Stellen einbinden. [NK13]

Zur Laufzeit werden Template-Variablen an Funktionen gebunden, die beispielsweise Daten aus den Collections abrufen oder manipulieren [NK13]. Wie das Zusammenspiel zwischen Server, Client und der Template-Engine genau funktioniert, soll das nächste Beispiel erläutern. Die durch `Meteor.subscribe` zur Verfügung gestellte Collection `Cardsets` lässt sich mit der Anweisung in 6.6 unter dem `Created`-Template mit der Funktion `cardsetList` verwenden.

```
1 Template.created.helpers({
2   cardsetList: function() {
3     return Cardsets.find({...});
4   };
5 });
```

Listing 6.6: Veröffentlichung der Daten im Client

Ein weiterer wesentlicher Bestandteil unter Meteor ist die `each`-Anweisung mit dessen Hilfe über eine Collection iteriert werden kann. In Listing 6.7 wird die Tabelle mit Inhalten gefüllt und für jeden Kartensatz eine Zeile mit Namen und der zugehörigen Beschreibung erstellt.

```
1 {{#each cardsetList}}
2   <tr>
3     <td>{{name}}</td>
4     <td>{{description}}</td>
5   </tr>
6 {{/each}}
```

Listing 6.7: Iteration von Collections in Meteor

Neben Schleifen sind auch Bedingungen sinnvoll, die Bereiche des Templates je nach Zustand anzeigen oder ausblenden. In Beispiel 6.8 wird die Tabelle nur angezeigt, wenn Einträge in der Collection vorhanden sind.

```
1 {{#if cardsetList.count}}
2   <!-- Inhalt der Tabelle -->
3 {{else}}
4   <> cardsetEmpty<>
5 {{/if}}
```

Listing 6.8: Bedingungen in Meteor-Templates

Zu einer Plattform wie Meteor gehört auch ein Build- und Deployment-System, das die Veröffentlichung einer Anwendung erlaubt. Hierzu wird von Meteor eine eigene Infrastruktur unter dem Namen *Galaxy* zur Verfügung gestellt. [NK13]

Soll auf die Nutzung der Meteor-Infrastruktur verzichtet und eine gänzlich eigene Anwendung betrieben werden, lassen sich mit dem `build`-Kommando von Meteor alle notwendigen Ressourcen, die im Betrieb einer Meteor-Anwendung erforderlich sind, verpacken und in einer eigenen Infrastruktur installieren [?]. Dazu wird neben dem regulären Node.js-Server die Datenbank MongoDB benötigt [NK13].

Meteor bietet zudem ein ständig wachsendes und von der Community unterstütztes Verzeichnis unter dem Namen *Atmosphere* an, dass zahlreiche nützliche Erweiterungen bereithält. Einzelne Packages von *Atmosphere* sind sogar fest im Core von Meteor integriert. Darüber hinaus gibt es zahlreiche Tipps und Tricks rund um das Meteor-Framework. Die Dokumentation von Meteor ist sehr umfangreich und man erhält in Google Docs oder Stackoverflow Antworten aus der Meteor-Entwicklergemeinschaft. [NK13]

### 6.3.2. Node.js

Node.js ist eine plattformübergreifende Laufzeitumgebung für JavaScript-Anwendungen. Sie ermöglicht Entwicklern die leichte Entwicklung von schnellen und skalierbaren JavaScript-Applikationen. Alle Node.js Anwendungen werden typischerweise auf dem Server ausgeführt. [Wikf]

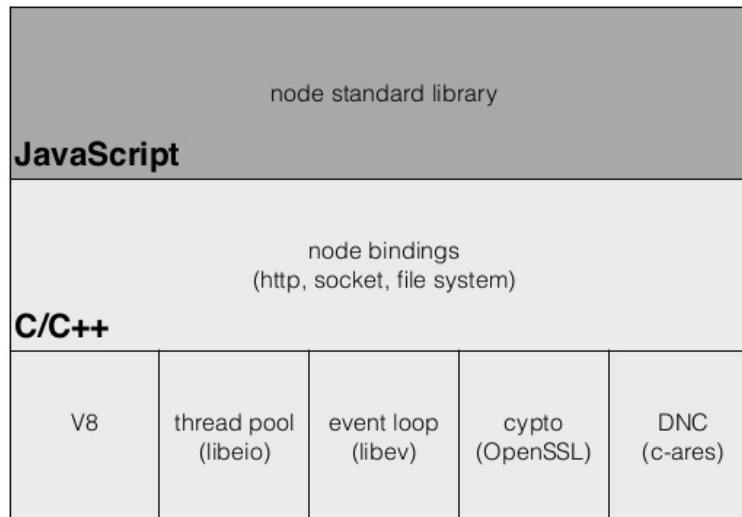


Abbildung 6.4.: Die Architektur von Node.js [Hec16, Seite 12]

Vorgestellt wurde Node.js im Jahr 2009 von Ryan Dahl. Die Node.js-Plattform besteht im Kern aus einer Ansammlung von verschiedensten Bibliotheken. Dieser Aufbau ermöglicht es die Vorteile der einzelnen Bibliotheken innerhalb einer Plattform zu nutzen und gleichzeitig eine unabhängige Weiterentwicklung zu ermöglichen, da die Bibliotheken getrennt voneinander gepflegt werden. In der Architektur von Node.js, wie in Abbildung 6.4 dargestellt, ist die Basis der Plattform in C und C++ programmiert. Um zusätzliche Module ohne große Schwierigkeiten zu erstellen, werden diese in JavaScript entwickelt. Das Herzstück von Node.js ist die V8-Engine von Google. Das primäre Einsatzgebiet der Engine ist der Chrome-Browser, dennoch kann sie auch unabhängig für die Interpretation und Ausführung von JavaScript-Code genutzt werden. [Spr13a, Seite 19-25 und 71-72]

Da Node.js durch den Single-Threaded-Ansatz nur eine Operation gleichzeitig ausführen kann, gibt es den so genannten Event Loop (siehe Abbildung 6.5). Dieser verhindert das Blockieren des Threads, indem alle zeitintensiven Operationen, wie lesende und schreibende Zugriffe, an das Betriebssystem ausgelagert werden.

Nach Abschluss der Aufgaben wird das Ergebnis mittels eines Callbacks an den Eventloop zurückübermittelt. [Spr13b]



Abbildung 6.5.: Das konzeptionelle Modell von Node.js [Nod]

Da es zwischen Joyent, dem Hauptsponsor von Node.js, und der Community zu immer größeren Differenzen bei der Weiterentwicklung kam, entschied sich Ende 2014 die Community dazu einen Fork (Javascript I/O: <https://iojs.org/de/>) von Node.js zu erstellen [Grü14]. Jedoch sind beide Projekte mittlerweile wieder unter der neu gegründeten Node.js-Foundation vereint. Mit Version 4.0.0 wurde der Zusammenschluss im August 2015 komplettiert [Nod15]. Aktuell liegt Node.js in der Version 6.6.0 vor. Die neuste Meteor-Version 1.4.1 verwendet allerdings die Long-Term-Support-Version 4.5.0 von Node.js [New16].

Seit der Meteor-Version 1.3 werden npm<sup>1</sup>-Module vollends unterstützt. Zukünftig sollen alle Atmosphere-Plugins durch npm-Module ersetzt werden. [Meta]

### 6.3.3. MongoDB

MongoDB ist eine dokumentenorientierte und somit eine schemafreie Datenbank. Dadurch haben Datensätze keine einheitliche Struktur und können auf Grund dessen Einträge mit unterschiedlichen Typen enthalten. Darüber hinaus ist die Erfassung mehrerer Werte durch das Einbinden von Arrays, sowie eine verschachtelte Struktur möglich. [Mon13, Wike]

Nach Aussage von Merriman, einem der ursprünglichen Entwickler von MongoDB, leitet sich der Name der Datenbank vom englischen Wort *humongous* (zu

<sup>1</sup>Node Package Manager

Deutsch: riesig, gigantisch) ab, um zu verdeutlichen, dass das Programm den Umgang mit umfangreichen Datenmengen unterstützt. [Wike]

Die Datenbank erschien im Jahr 2009 als Open-Source-Software unter der kommerziellen Lizenz GNU AGPL Version 3.0 der Free Software Foundation. [Wike]

### Popularität

Seit einigen Jahren ist MongoDB die populärste dokumentenbasierte Datenbank. Dies ist im Trend-Diagramm (Abbildung 6.6) dargestellt. Hierbei wird ersichtlich, dass ein erheblicher Vorsprung von MongoDB zu anderen Datenbanksystemen besteht. Betrachtet man alle verfügbaren Datenbanksysteme, liegt MongoDB auch hier auf einem guten fünften Platz. Es wird lediglich von relationalen Datenbanksystemen wie MySQL deutlich geschlagen, die jedoch für große Datenmengen unbrauchbar sind. [DE16]

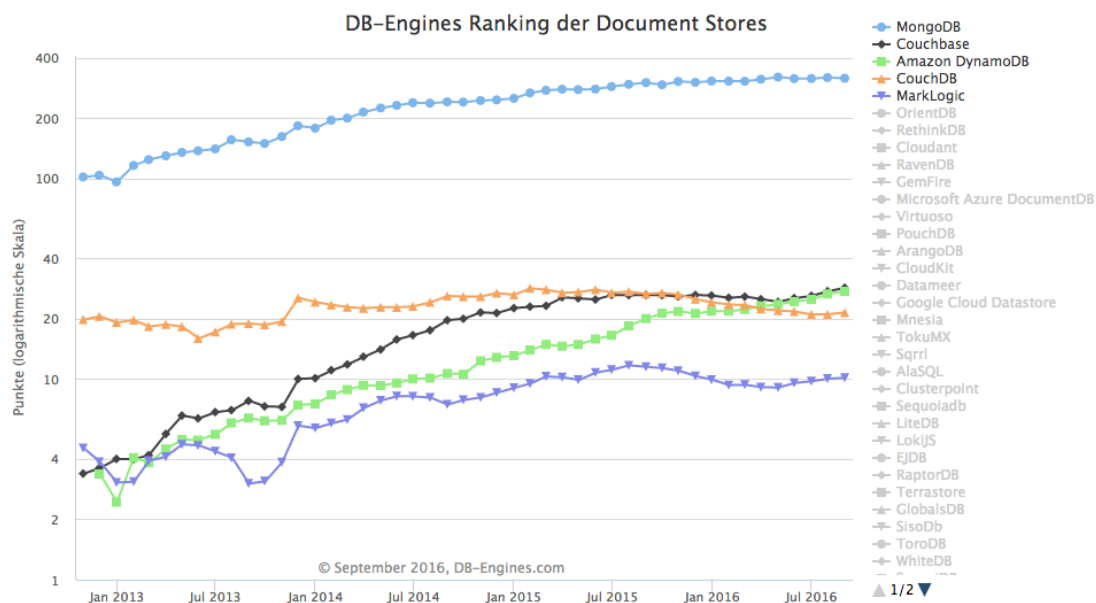


Abbildung 6.6.: Popularität von dokumentenbasierten Datenbanksystemen [DE16]

### Aufbau

Eine MongoDB kann, wie in Abbildung 6.7 dargestellt, mehrere Datenbanken enthalten, die wiederum aus mehreren Collections bestehen. Dabei kann jede Collection ein oder mehrere Dokumente umfassen. Ein Dokument kann Auf-

grund der schemafreien Implementierung von MongoDB unterschiedliche Objekte, auch Key-Value-Paare genannt, beinhalten. Ein Objekt selbst besteht neben dem Schlüsselwort entweder aus einem einfachen Wert, einem Array oder einer Liste von Key-Value-Paaren. [Sut12]

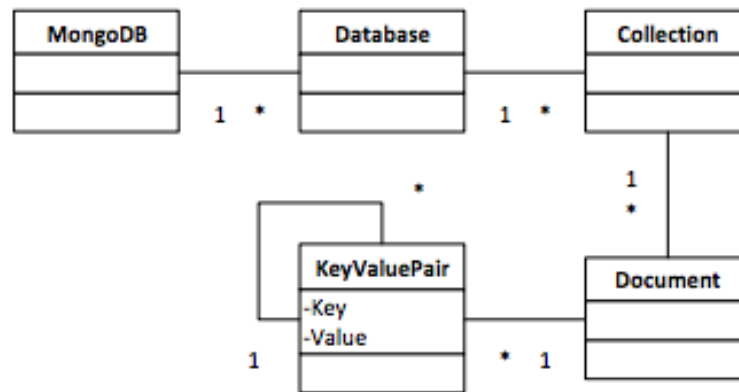


Abbildung 6.7.: Aufbau der dokumentenbasierten Datenbank MongoDB [Sut12]

Für die Datenspeicherung und den Datenaustausch verwendet die Datenbank das sogenannte BSON-Format (kurz für Binary JSON). Dieses Format bietet eine binäre Darstellung von JSON-ähnlichen Dokumenten, die so entwickelt wurde, dass folgende Charakteristiken gewährleistet werden: Leichtgewichtigkeit (Lightweight), eine schnelle Durchsuchung der Inhalte (Traversable), sowie die Effizienz bei der En- und Decodierung des Formates (Efficient). [bso]

### Integration in Meteor

Wie in Abbildung 6.3 dargestellt, läuft Meteor serverseitig innerhalb einer Node.js-Runtime. Dabei wird innerhalb des Meteor-Servers mittels der Mongo-API auf die MongoDB zugegriffen. Auf dem Client wird mithilfe von Minimongo ein lokales Abbild der MongoDB erstellt. Unter Zuhilfenahme von DDP (Distributed Data Protocol) werden alle Daten, die für den Benutzer freigeschaltet sind, lokal auf dem neusten Stand gehalten. Das Laden der Daten findet standardmäßig etwa alle 10 Sekunden oder sobald diese serverseitig durch einen anderen Client modifiziert wurden, statt. Ändert ein Benutzer lokal die Daten, müssen diese durch einen Methoden-Aufruf serverseitig aktualisiert werden, um in die MongoDB übernommen zu werden. [Mar16]

Um eine Validierung der eingefügten Werte zu ermöglichen, ist der Einsatz zusätzlicher Plugins notwendig. Innerhalb von Atmosphere gibt es unter anderem

die Erweiterung *SimpleSchema*, die es ermöglicht, Regeln für Key-Value-Paare einzelner Collection einzuhalten. Zu den Regeln gehören beispielsweise das Festlegen des erlaubten Typs oder die Angabe einer maximalen Länge eines Wertes. Durch die Einhaltung der definierten Vorgaben wird bei einer grundlegend schemafreien Datenbank eine gewisse Ordnung geschaffen. [Metb]

## **7. Anwendung und Implementierung in THMcards**

Im Rahmen dieser Masterarbeit sollen die vorgestellten Datenschutz- und Datensicherheitsmaßnahmen genutzt werden, um das E-Learning System THMcards aus datenschutzrechtlicher Sicht und bezüglich der Datensicherheit zu verbessern. Neben einem Back-End zur Verwaltung der Lerninhalte und der Benutzer von THMcards wurden von Meteor vorgegebene Maßnahmen eingepflegt, um die Datensicherheit einer solchen Anwendung zu gewährleisten. In diesem Kapitel sollen diese Erweiterungen in Bezug auf Datenschutz und Datensicherheit aufgezeigt werden, die ein Front-End-Developer bei der Entwicklung einzuhalten hat.

### **7.1. Implementierung geeigneter Datenschutzmaßnahmen**

E-Learning Systeme wie THMcards, die online personenbezogene Daten verarbeiten und nutzen, sind dazu verpflichtet, datenschutzrechtliche Vorgaben einzuhalten. Dies ist nicht nur wegen der Abmahnungen und Bußgelder von Bedeutung, sondern steigert auch das Vertrauen der Benutzer und damit die Seriosität eines E-Learning Systems. Ein Kunde, der nicht nachvollziehen kann, welche Daten für welchen Zweck gespeichert oder übermittelt werden, nimmt im Zweifel das Angebot des E-Learning Systems nicht in Anspruch. Daher sollte die Einhaltung der datenschutzrechtlichen Vorgaben als vertrauensbildende Maßnahme verstanden werden.

Aus den datenschutzrechtlichen Bestimmungen ergeben sich folgende Pflichten, die in THMcards eingepflegt worden sind:

- Impressumspflicht.
- Auskunft über Cookies.

- Pflicht zur Unterrichtung der Benutzer über die gespeicherten Daten (Datenschutzerklärung).
- Pflicht zur Unterrichtung der Benutzer über die Allgemeinen Geschäftsbedingungen (AGB).
- Einwilligung des Benutzers als Voraussetzung für die Verarbeitung und Übertragung von personenbezogenen Daten.
- Festlegung von Urheberrechten auf Lernkarten.
- Back-End zur Verwaltung von Lerninhalten und Benutzern.
- Einhaltung der Verantwortlichkeiten eines Host-Providers durch ein Benachrichtigungssystem.

Obwohl die nachfolgenden Maßnahmen ausführlich recherchiert worden sind, soll darauf hingewiesen werden, dass nicht für die Vollständigkeit der Angaben garantiert werden kann. Für eine rechtssichere Implementierung sollte eine professionelle Unterstützung eines Anwalts oder Datenschutzbeauftragten in Anspruch genommen werden.

### 7.1.1. Informationspflichten

Jede öffentliche Webseite, die personenbezogene Daten nutzt und verarbeitet, ist dazu verpflichtet, bestimmte Informationen in der Webseite für die Benutzer bereitzustellen. Dazu gehört die Angabe eines Impressums, die Unterrichtung über die Allgemeinen Geschäftsbedingungen, die Auskunft über erhobene und genutzte Daten in einer Datenschutzerklärung oder das Führen eines Verfahrensverzeichnis.

#### Impressum

Wie in Kapitel 3.4.1 *Impressumspflicht* bereits beschrieben wurde, dient das Impressum dazu, dem Benutzer der Webseite über den jeweiligen Betreiber zu informieren, um rechtliche Ansprüche gegen diesen erheben zu können. Voraussetzung der Impressumspflicht ist nach § 5 TMG das geschäftsmäßige Betreiben eines Internetangebots. Da THMcards Lernkarten kostenpflichtig anbietet, ist die Führung eines Impressums folglich Pflicht. Das in THMcards eingepflegte Impressum

kann im Abschnitt A.1 des *Anhangs* dieser Masterarbeit oder unter dem Link <https://arsnova.cards/impressum> gefunden werden.

### Allgemeine Geschäftsbedingungen

Die Allgemeinen Geschäftsbedingungen (AGB) beschreiben den rechtliche Rahmen für Verträge, die der Benutzer einer Webseite mit dem Webseitenbetreiber abschließt. Durch die AGB hat der Betreiber die Möglichkeit Regelungen, die durch das Gesetz nicht abgedeckt werden, festzulegen, solange nicht vom wesentlichen Grundgedanken des Gesetzes abgewichen wird. [Sie16a][Sch11b]

Eine Webseite ist gesetzlich nicht dazu verpflichtet, eigene Geschäftsbedingungen festzulegen. Wird jedoch auch an private Kunden verkauft, müssen vom Gesetz vorgeschriebene Belehrungspflichten in den AGB umgesetzt werden. Darunter fallen Pflichtangaben wie Widerrufsbelehrungspflichten, Hinweise zum Vertragsabschluss oder Informationspflichten zu Preisangaben.[Sie16a]

Lerninhalte, die in THMcards kostenpflichtig zur Verfügung gestellt werden, können nur von Studenten der Technischen Hochschule Mittelhessen, also privaten Verbrauchern, genutzt werden. Damit ist THMcards verpflichtet, Benutzer über die Allgemeinen Geschäftsbedingungen zu unterrichten. Diese können im Abschnitt A.3 des *Anhangs* der Masterarbeit oder unter <https://arsnova.cards/agb> eingesehen werden.

### Datenschutzerklärung

Durch das am 24.02.2016 in Kraft getretene *Gesetz zur Verbesserung der zivilrechtlichen Durchsetzung von verbraucherschützenden Vorschriften des Datenschutzrechts* soll der Schutz von Verbrauchern im Internet verbessert werden, indem effektiver gegen unseriöse Unternehmen vorgegangen wird. Das bedeutet, dass Webseiten mit Abmahnungen rechnen müssen, wenn keine oder eine unvollständige Datenschutzerklärung auf der Webseite eingebunden ist. [Sie16b]

In einer Datenschutzerklärung muss darauf hingewiesen werden, welche personenbezogenen Daten von der Webseite gespeichert werden und was mit diesen Daten geschieht. Die Datenschutzerklärung von THMcards erhält Informationen über:

- Die verwendeten Anwendungsdaten oder Zugriffsdaten.
- Cookies.

- Eingebundene Dienste und Inhalte Dritter.
- Die Datenübermittlung an Dritte.
- Daten, die für die finanzielle Transaktion genutzt werden.
- Das Widerspruchs- und Auskunftsrecht der gespeicherten personenbezogenen Daten.

Die verfasste Datenschutzerklärung ist im Abschnitt A.2 des *Anhangs* dieser Masterarbeit oder unter <https://arsnova.cards/datenschutz> zu finden.

### Cookie Consent

Werden von Cookies personenbezogene Daten bezogen, reicht es nicht, Benutzer lediglich über die Datenschutzerklärung auf die Nutzung von Cookies hinzuweisen. Es müssen genaue Informationspflichten eingehalten und die Einwilligung des Benutzers eingeholt werden.

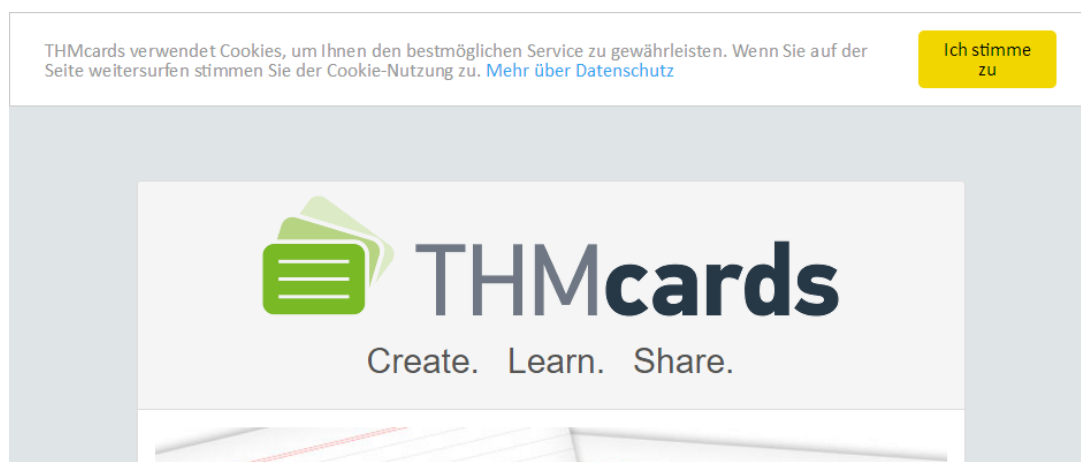


Abbildung 7.1.: Cookie Consent-Plug-In in THMcards

Hierfür ist in THMcards das *Cookie Consent*-Plug-In eingebunden. Über ein einfaches Skript wird ein auf den ersten Blick sichtbarer Banner auf der Webseite dargestellt, der erst ausgeblendet wird, wenn der Benutzer dem Einwilligungstext zustimmt (siehe Abbildung 7.1). Das Verfahren entspricht damit einem erleichterten Opt-In und schränkt die Anwendung nicht in der Benutzerfreundlichkeit ein.

### Verfahrensverzeichnis

Das Verfahrensverzeichnis ist ein Element des Datenschutzes und muss von jeder öffentlichen oder privaten Stelle, die personenbezogene Daten bezieht, geführt werden. Es dient zur Dokumentation des Umgangs mit personenbezogenen Daten. [act] [Wikj]

Man unterscheidet zwischen einem *internen* und *öffentlichen* Verfahrensverzeichnis. Das interne Verfahrensverzeichnis enthält umfangreichere Angaben als das öffentliche und soll eine betriebsinterne Selbstkontrolle ermöglichen. Das öffentliche Verfahrensverzeichnis soll eine Einsicht auf die Verarbeitung personenbezogener Daten für Dritte verschaffen und kommt damit dem Datenschutzgrundsatz der *Transparenz* nach. [Datf]

In THMcards wird ein öffentliches Verfahrensverzeichnis geführt, um Benutzern des Systems den Einblick auf die Verarbeitung personenbezogener Daten zu ermöglichen. Folgende Inhalte müssen nach § 4e BDSG enthalten sein und sind im Verfahrensverzeichnis von THMcards dokumentiert:

- Name oder Firma der verantwortlichen Stelle.
- Inhaber, Vorstände, Geschäftsführer oder sonstige gesetzliche oder nach der Verfassung des Unternehmens berufene Leiter und die mit der Leitung der Datenverarbeitung beauftragten Personen.
- Anschrift der verantwortlichen Stelle.
- Zweckbestimmungen der Datenerhebung, -verarbeitung oder -nutzung.
- Kreis der Betroffenen.
- Mögliche Empfänger bei einer Übermittlung.
- Löschfristen.
- Eine geplante Datenübermittlung in Drittstaaten.
- Technische und organisatorische Maßnahmen zum Schutz der Daten nach § 9 BDSG.

Das Verfahrensverzeichnis von THMcards kann im Abschnitt A.4 des *Anhangs* dieser Masterarbeit oder unter <https://git.thm.de/arsnova/flashcards/wikis/verfahrensverzeichnis> nachgelesen werden.

### 7.1.2. Einwilligung

Es reicht nicht aus, die Allgemeinen Geschäftsbedingungen und die Datenschutzerklärung an einer beliebigen Stelle der Webseite des E-Learning Systems einzubinden. AGB sind nur Bestandteil eines Vertrages, wenn der Benutzer in diesen Vertrag einbezogen wird [Sie16a]. Das bedeutet, dass der Benutzer bei Vertragschluss ausdrücklich auf die AGB hingewiesen wird und er die Möglichkeit hat, diese jederzeit leicht zu erreichen (§ 13 Absatz 2 TMG). Dies gilt auch, wenn personenbezogene Daten auf der Webseite verarbeitet und genutzt werden. Der Benutzer muss die Verwendung seiner Daten immer ausdrücklich zustimmen.

Bei Unklarheiten oder Anregungen wenden Sie sich an folgende E-Mail-Adresse oder an die im Impressum angegebenen Kontaktdaten: [klaus.quibeldy-cirke@transmit.de](mailto:klaus.quibeldy-cirke@transmit.de)

Eine Übersicht über die in THMcards gespeicherten Daten finden Sie im Verzeichnissesverzeichnis unter: <https://git.thm.de/arsnova/flashcards/wikis/verfahrensverzeichnis>

#### Änderungen der Datenschutzerklärung

Wir behalten uns das Recht vor, diese Datenschutzerklärung jederzeit unter Beachtung der geltenden Datenschutzvorschriften zu ändern. Derzeitiger Stand ist der 17.10.2016.

☐ Ich habe die allgemeinen Geschäftsbedingungen und Datenschutzerklärung gelesen und bin damit einverstanden.





Abbildung 7.2.: Elektronische Einwilligung in THMcards

In THMcards wird die Einwilligung elektronisch gelöst. Diese elektronische Einwilligung muss einer persönlichen handschriftlichen Signatur entsprechen, also eine eindeutige und bewusste Handlung darstellen. Bei der ersten Anmeldung in THMcards muss der Benutzer über die AGB und Datenschutzerklärung scrollen, bis er zu einer Schaltfläche gelangt, an dem er diese einwilligen kann. Zunächst muss der Benutzer durch einen Mausklick zu erkennen geben, dass er den Inhalt der Einwilligung gelesen und verstanden hat. Durch das weitere Anklicken der *Akzeptieren*-Schaltfläche wird in einem zweiten Schritt deutlich gemacht, dass der Benutzer die Allgemeinen Geschäftsbedingungen akzeptiert und der Verwendung der in der Datenschutzerklärung beschriebenen Daten zustimmt (siehe Abbildung 7.2).

### 7.1.3. Urheberrecht

Grundsätzlich gilt, dass alle Inhalte, die im Web veröffentlicht werden, urheberrechtlich geschützt sind. Selbst wenn kein ausdrücklicher Hinweis vorhanden ist, muss davon ausgegangen werden, dass diese Inhalte nicht einfach weiterverwendet werden dürfen. Um diese Missverständnisse zu vermeiden und den daraus folgenden Rechtswidrigkeiten vorzubeugen, sind in THMcards die *Creative-Commons*-Lizenzen eingebunden (siehe Abbildung 7.3). Mit diesen kann ein Benutzer festlegen, unter welchen Bedingungen veröffentlichte Kartensätze von anderen Benutzern des Systems weiterverwendet werden dürfen. Dadurch kann in THMcards eine Urheberrechtsverletzung klar bestimmt werden.

The screenshot shows a window titled "Lizenz wählen" (Choose License) with a close button (X) in the top right corner. Below the title bar, there is a section labeled "Rechtemodule" (Legal Modules) containing four icons in circles: a person (BY), a crossed-out Euro symbol (NC), an equals sign (ND), and a circular arrow (SA). Below these icons, there are four lines of text explaining each module's requirement:

-  Der Name des Urhebers muss genannt werden.
-  Das Werk darf nicht für kommerzielle Zwecke verwendet werden.
-  Das Werk darf nicht verändert werden.
-  Das Werk muss nach Veränderungen unter der gleichen Lizenz weitergegeben werden.

At the bottom right of the dialog, there are two buttons: "Abbrechen" (Cancel) and "Lizenz speichern" (Save License), which is highlighted in blue.

Abbildung 7.3.: Creative-Commons-Lizenzen in THMcards

### 7.1.4. Back-End

THMcards stellt den Benutzern Speicherplatz auf einem Server zur Verfügung, um eigene Lerninhalte zu erstellen und diese mit anderen Benutzern des Systems in einer virtuellen Lernumgebung zu teilen. Damit ist THMcards ein Host-Provider

und gesetzlich nicht für die bereitgestellten Inhalte verantwortlich. Erst bei der Kenntnisnahme einer Rechtswidrigkeit muss der Provider von THMcards den Zugang zu dem Inhalt sperren oder diesen löschen.

Um angemessen mit den entstehenden Rechtswidrigkeiten umgehen zu können, ist in THMcards ein Back-End implementiert, das es erlaubt, Lerninhalte und Benutzer des Systems zu verwalten. Hat ein Benutzer die Zugriffsrechte eines Administrators (Admin-User oder Editor-User), kann dieser den Bereich des Back-End betreten.

Das Back-End ist in mehrere Ansichten aufgeteilt, um eine optimale Verwaltung der Inhalte zu ermöglichen. Die *Dashboard*-Ansicht bietet eine Übersicht über alle in THMcards vorhandenen Inhalte (siehe Abbildung 7.4). Hier wird angezeigt, wie viele Kartensätze, Karten und Benutzer im System vorhanden sind und über die derzeit angemeldeten Benutzer informiert.

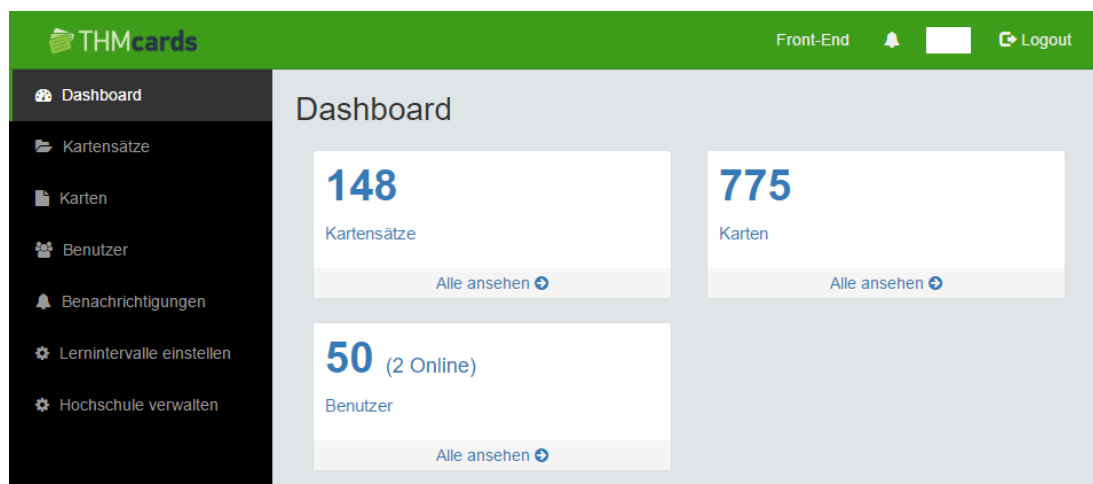


Abbildung 7.4.: Dashboard-View im Back-End von THMcards

Die Ansichten *cardsets*, *cards* und *users* sind alle nach dem gleichen Schema aufgebaut. In einer Tabelle werden alle Einträge der spezifischen Collection dargestellt, z.B. zeigt die Ansicht *users* alle Einträge der *Meteor.users*-Collection (siehe Abbildung 7.5). Die Einträge lassen sich nach den Spalten der Tabelle sortieren, mithilfe einer Suche filtern oder löschen. Über die Schaltfläche, die eine Abbildung eines Stift-Icons zeigt, gelangt man zur Einzelansicht des gewählten Eintrages. In der Einzelansicht werden die Kartensätze, Karten und Benutzer des Systems bearbeitet. Beispielsweise kann ein Administrator in der Einzelansicht von *users* Rollen verteilen, Benutzer blockieren oder aus dem System löschen.

Weitere Ansichten im Back-End sind:

Admin	Benutzer	Pro	Edu	Dozent	Mail	Beigetreten	Blockiert	Bearbeiten	Löschen
✓ (Super Admin)		✓	✓	✓	✉	17. September 2016		✎	
✓ (Super Admin)			✓		✉	19. September 2016		✎	
✓		✓	✓	✓	✉	17. September 2016		✎	✕
✓		✓	✓	✓	✉	18. September 2016		✎	✕
✓		✓	✓	✓	✉	17. September 2016		✎	✕

Abbildung 7.5.: Benutzeransicht im Back-End von THMcards

- *notifications*: Übersicht über erhaltene/versendete Benachrichtigungen (siehe Kapitel 7.1.5 *Notifications*).
- *interval*: Ansicht zur Einstellung der Intervalle für den Lernzeitraum.
- *settings*: Ansicht zur Verwaltung von Hochschulen und Studiengängen des Systems.

### 7.1.5. Notifications

Fühlt sich ein Benutzer von THMcards in seinem Recht verletzt, z.B. durch eine Persönlichkeits- oder Urheberrechtsverletzung, muss dieser die Möglichkeit besitzen, die Rechtswidrigkeit bei den zuständigen Administratoren des Systems zu melden. Hierfür sind im System *Notifications* implementiert. *Notifications* sind in THMcards integrierte Nachrichten, die ein Benutzer bei einer Rechtsverletzung an den Administrator senden kann.

In der Ansicht eines öffentlichen Kartensatzes hat der Benutzer von THMcards die Möglichkeit, einen rechtlichen Verstoß des gewählten Kartensatzes oder des Benutzers, der den Kartensatz erstellt hat, zu melden. Mit einem Mausklick auf die Schaltfläche *Kartensatz/Benutzer melden* kann ein Benutzer in einem Pop-Up-Fenster den Grund der Rechtswidrigkeit wählen und detaillierte Informationen über den Verstoß angeben. Die Beschwerde wird beim Absenden an jeden vom System festgelegten Administrator gesendet.

In der Ansicht *notifications* im Back-End von THMcards erhalten die Administratoren eine Übersicht über alle im System versendeten Benachrichtigungen.

Typ ^	Absender	Grund	Weitere Informationen	Datum	Mail an Beschuldigten	Mail an Absender	Löschen
Gemeldeter Kartensatz	Matthias Tlu	Zahlensysteme	In Ihrem Kartensatz wurde eine Verletzung im Urheberrecht festgestellt. Bitte entfernen Sie den Inhalt.	Dienstag, 8. November 2016 17:27			

Page 1 of 1

Abbildung 7.6.: Übersicht der Benachrichtigungen im Back-End von THMcards

Die Ansicht ist hierzu in *Benutzerbeschwerden*, *Dozenten Anfragen* und *Gesendete Nachrichten* aufgeteilt (siehe Abbildung 7.6). In *Benutzerbeschwerden* werden die von Benutzern gemeldeten Rechtsverletzungen aufgelistet. Liegt eine Rechtsverletzung vor, kann ein Administrator den Beschuldigten auf diese hinweisen. Bei Unklarheiten wendet sich der Administrator an den Absender. Wird die Rechtswidrigkeit nach dem Hinweis eines Administrators nicht behoben, kann dieser den rechtlichen Verstoß beseitigen, indem über das Back-End der rechtswidrige Inhalt gelöscht oder der gemeldete Benutzer aus dem System blockiert wird. Um sich rechtlich abzusichern, werden alle versendeten Benachrichtigungen des Administrators in der Ansicht *Gesendete Nachrichten* angezeigt und in THMcards gespeichert.

Des Weiteren erhält ein Administrator im Tab *Dozenten Anfragen* Benachrichtigungen von Benutzern, die in THMcards als Dozent freigeschaltet werden wollen. Der genaue Vorgang, wie die Rolle eines Dozenten an einen Benutzer von THMcards vergeben wird, kann in der Masterarbeit *Erfolgreiche Monetarisierung einer Open-Source Webanwendung durch die Integration eines innovativen Geschäftsmodells* von Marius Trautrimms nachgelesen werden [Tra16].

## 7.2. Implementierung geeigneter Datensicherheitsmaßnahmen

Um die Datensicherheit von THMcards zu gewährleisten, sind bei der Entwicklung mit Meteor einige Faktoren zu beachten. Hierzu werden von Meteor Techniken für die Verbesserung der Sicherheit solcher Anwendungen angeboten [Metc].

### 7.2.1. Packages

Durch Packages hat man in Meteor die Möglichkeit, Anwendungen mit wenig Aufwand um weitere Funktionalitäten zu erweitern. In Bezug auf die Sicherheit muss darauf geachtet werden, welche Packages sicher sind, welche Schaden anrichten können und welche installiert werden sollten, um die Sicherheit zu verbessern. Bei der Umsetzung geeigneter Sicherheitsmaßnahmen in THMcards gilt es also zunächst zu klären, welche Packages in das System eingepflegt und welche aufgrund aufkommender Sicherheitslücken entfernt werden müssen.

#### Autopublish und Insecure

Wenn mit Meteor eine neue Anwendung erstellt wird, werden zwei Packages automatisch mit installiert: *autopublish* und *insecure*. Diese machen die anfängliche Entwicklung einer Anwendung unter Meteor schneller und einfacher, sind jedoch nur für die Prototyping-Phase geeignet. Projekte wie THMcards, die über die Phase hinaus gehen und in Produktion geschaltet werden, müssen die Packages *autopublish* und *insecure* entfernen.

Durch *autopublish* werden alle Collections und alle Daten automatisch im Client veröffentlicht. Das hat nicht nur Konsequenzen für die Sicherheit, sondern auch für die Leistungsfähigkeit der Anwendung, da es durch die Veröffentlichung aller Daten zu einer Überlastung des Servers kommen kann.

Mit dem Package *insecure* haben alle Clients der Anwendung uneingeschränkter Zugriff auf die Datenbank. Dadurch ist es möglich, über die Funktionen `insert`, `update` und `remove` Einträge der Datenbank über die Konsole des Webrowsers zu bearbeiten. Um die Sicherheit zu erhöhen, sollten diese Funktionen nur vom Server aus aufgerufen werden können, indem das *insecure*-Package aus dem Projekt entfernt wird.

#### Check und SimpleSchema

Mithilfe von *check* und *SimpleSchema* wird THMcards um weitere Sicherheitsmaßnahmen erweitert. Diese Packages erlauben es, Daten, die vom Client an den Server oder in die Datenbank gesendet werden, zu validieren und damit NoSQL-Injections zu verhindern.

Daten, die in THMcards vom Client zum Server gesendet werden, sichert das *check*-Package ab. Es wird sichergestellt, dass nur die gewünschten Daten an

die entsprechenden Methoden im Server gesendet werden. Der Funktionsaufruf `check()` validiert dazu die empfangenen Daten auf ihren Datentyp und verhindert, dass die Methode bei einem falschen Datentyp vollständig ausgeführt wird. Demnach können unerwünschte Daten nicht in die Datenbank gelangen.

Mit *SimpleSchema* werden für die MongoDB Collections von THMcards Datenbankschemata festgelegt. Datenbankschemata definieren Regeln, die festlegen, auf welche Weise Daten in der Datenbank gespeichert und bearbeitet werden dürfen [Glo15]. Damit wird eine gleichbleibende Struktur der Collections garantiert, sodass keine Überprüfung der Daten, die aus der Datenbank entnommen werden, stattfinden muss. Der Versuch, ein Dokument in eine Collection zu speichern, das nicht dem vordefinierten Datenbankschemata entspricht, wird durch das Package unterbunden.

### Browser Policy

Das Package *Browser Policy* verhindert den unbefugten Zugriff auf THMcards durch Dritte, indem das Sicherheitskonzept Content Security Policy (CSP) in das Projekt eingebunden wird. Die durch das Sicherheitskonzept implementierten Regeln erlauben es, Attacken wie Cross-Site-Scripting oder Code Injection abzuwehren und verhindern, dass die Anwendung geframed wird.

Durch das *Browser Policy*-Package lässt sich definieren, welche Inhalte von fremden Webseiten in THMcards geladen werden können und wer dazu befugt ist, THMcards zu framen (z.B. durch `<iframe></iframe>`). Dazu wird in THMcards zunächst grundlegend festgelegt, welche Aktionen in der Anwendung blockiert werden sollen und welche erlaubt sind:

```
1 BrowserPolicy.framing.disallow();
2 BrowserPolicy.content.disallowInlineScripts();
3 BrowserPolicy.content.disallowEval();
4 BrowserPolicy.content.allowInlineStyles();
5 BrowserPolicy.content.allowFontDataUrl();
```

Listing 7.1: Durch Browser Policy blockierte Aktionen in THMcards

Mit diesen Einstellungen werden folgende Regeln für THMcards festgelegt:

- Framen der Anwendung verhindern.
- Inline-JavaScript, also das Setzen von `<script>`-Tags in der Anwendung, verhindern.

- Die Nutzung der `eval()`-Methode verhindern.
- Inline-Styles wie `<div style="background: white;">` werden in der Anwendung zugelassen.
- Fonts dürfen über Data-URL geladen werden.

Anschließend werden Ausnahmen definiert, bei denen externe Skripte geladen werden dürfen. Folgende Anweisung erlaubt es, Skripte von Google Analytics über eine HTTPS-Verbindung zu laden:

```
1 BrowserPolicy.content.allowOriginForAll(  
2     'https://*.google-analytics.com');
```

Listing 7.2: Eine durch Browser Policy definierte Ausnahme in THMcards

### 7.2.2. Datenbank

Die Datenbank ist der empfindlichste Teil von THMcards. Hier werden alle sensiblen Daten wie die Lerninhalte und Benutzer des Systems gespeichert. Es gilt zu verhindern, dass diese Daten, unabhängig davon, ob sie personenbezogen sind oder nicht, von Dritten manipuliert, gelöscht oder entwendet werden können. Um die Sicherheit dieser gespeicherten Daten zu gewährleisten, sind bei der Programmierung mit Meteor einige Richtlinien zu beachten, wenn eine Anwendung entwickelt werden soll. Welche dieser Richtlinien unter THMcards eingepflegt sind, soll nun aufgezeigt werden.

#### Publish und Subscribe

Nachdem das Package *autopublish* aus THMcards entfernt worden ist, muss nun explizit definiert werden, welche Daten im Server veröffentlicht und an den Client gesendet werden. Wie bereits in Kapitel 6.3.1 erwähnt wurde, erlaubt das *Publish/Subscribe*-Verfahren über **publish** Ergebnisse einer Datenbankabfrage auf dem Server zu veröffentlichen und über **subscribe** im Client zu abonnieren. Das Besondere hierbei ist, dass dadurch Collections nur die Daten bereit halten, die vorher festgelegt worden sind. Das soll Abbildung 7.7 nochmal verdeutlichen:

Durch **publish** werden die Dokumente 1-4 vom Server geladen, auf die Resultate 2 und 4 gefiltert und anschließend zum Client gesendet. In Bezug auf die Sicherheit spielt dieses Verfahren eine wichtige Rolle, da die grundsätzliche Veröffentlichung aller Daten im Client verhindert werden sollte. Werden beispielsweise



Abbildung 7.7.: Veröffentlichung von Karten

alle Benutzer-Dokumente einer Collection veröffentlicht, könnte jeder Benutzer von THMcards auf die Daten eines anderen Benutzers des Systems zugreifen. Mit `publish` lässt sich explizit bestimmen, welche Daten in welcher Situation an den Client gesendet werden.

In THMcards wird `publish` verwendet, um Dokumente, abhängig von der aktuell angemeldeten Benutzer-ID, im Client anzuzeigen:

```
1 Meteor.publish("privateUserData", function () {  
2   ...  
3   return Meteor.users.find({_id: this.userId});  
4   ...  
5 });
```

Listing 7.3: Filtern der Dokumente anhand der Benutzer-ID in THMcards

Dokumente aus den Collections werden in THMcards nur dann veröffentlicht, wenn diese eigene Lerninhalte oder personenbezogene Daten des Benutzers sind. Darüber hinaus werden mit einer zweiten `publish`-Anweisung mit dem Filter

`visible: true` öffentliche Kartensätze und öffentliche Benutzerprofile für andere Benutzer von THMcards zugänglich gemacht. Meldet sich der Benutzer von THMcards ab, werden keine Dokumente an den Benutzer zurückgegeben.

### Overpublishing

Das nächste Sicherheitsproblem bei der Veröffentlichung von Daten aus der Datenbank ist das *Overpublishing*. Ähnlich wie bei dem Beispiel mit der Benutzer-ID muss konkret bestimmt werden, welche Daten an den Client gesendet werden. Zum Beispiel: In THMcards soll nur der Name des Benutzers angezeigt werden. Entsprechend wird folgende Anweisung erstellt:

```
1 Meteor.publish("userData", function (id) {  
2   return Meteor.users.find({_id: this.userId});  
3 });
```

Listing 7.4: Fehlerhafte Veröffentlichung des Benutzernamen

Mit dieser Anweisung werden neben dem Benutzernamen alle Daten der Benutzer-Collection zurückgegeben. Dies lässt sich verhindern, indem in der `find()`-Anweisung nur die Felder angegeben werden, die im Client veröffentlicht werden sollen:

```
1 Meteor.publish("userData", function (id) {  
2   return Meteor.users.find(  
3     {_id: this.userId},  
4     {fields: {'profile.name': 1, ...}}  
5   );  
6 });
```

Listing 7.5: Veröffentlichung des Benutzernamen in THMcards

Durch die Option `fields` in Kombination mit dem Wert "1" (1 steht für `true`) wird nur das festgelegte Feld für den Benutzernamen im Client von THMcards veröffentlicht. Wird der Wert "1" durch "0" ersetzt, werden spezifische Felder der Datenbank aus der Veröffentlichung entfernt.

### Vermeiden von `allow` und `deny`

Meteor bietet Anwendungen wie THMcards zwei Möglichkeiten Daten in Collections einzufügen, zu bearbeiten oder zu löschen: *Methods* und *allow/deny*. Durch *Method Calls* lassen sich Funktionen auf dem Server definieren und auf dem Client aufrufen. *Allow* und *deny* wiederum ermöglichen den direkten Zugriff auf die

Collections über den Client. Wird im Client eine `insert`, `update` oder `remove` Operation auf eine Collection aufgerufen, führt der Server die Callbacks aus, um zu überprüfen, ob die Operation erlaubt (`allow`) oder verweigert (`deny`) werden soll.

Die Entwickler von Meteor raten strengstens davon ab, *allow/deny*-Regeln in Anwendungen zu verwenden. Begründet wird dies dadurch, dass es für Entwickler schwierig ist, diese Regeln zu verstehen und richtig zu implementieren. Im Artikel *Allow & Deny Security Challenge* wurde bereits demonstriert, dass es sehr schnell zu einer fehlerhaften Umsetzung kommen kann. [Metc]

Aus diesem Grund wird in THMcards komplett auf *allow/deny*-Regeln verzichtet und ausschließlich mit *Method Calls* gearbeitet. Das bedeutet, dass Aufrufe vom Client wie `Cards.insert()` standardmäßig verhindert und damit unbefugte Zugriffe über den Client auf die Datenbank unterbunden werden.

### Zugriff auf die Meteor.users-Collection verwehren

Neben dem Verfahren auf *allow/deny*-Regeln in THMcards zu verzichten, sollte auch der clientseitige Zugriff auf die *Meteor.users*-Collection unterbunden werden. Im Gegensatz zu den selbst implementierten Collections ist die *Meteor.users*-Collection standardmäßig beschreibbar, selbst nachdem das *insecure*-Package entfernt wurde.

Um die Daten der *Meteor.users*-Collection und die damit verbundenen personenbezogenen Daten der Benutzer unter THMcards zu sichern, wird sichergestellt, dass kein unbefugter Zugriff auf die Daten über den Client erfolgen kann. Ausnahmsweise muss die Collection mit den sensibelsten Daten in der Anwendung mit den *allow/deny*-Regeln vor solchen Zugriffen geschützt werden:

```
1 Meteor.users.allow({
2   insert() { return false; },
3   update() { return false; },
4   remove() { return false; }});
5
6 Meteor.users.deny({
7   insert() { return true; },
8   update() { return true; },
9   remove() { return true; }});
```

Listing 7.6: allow/deny-Regeln in der Meteor.users-Collection in THMcards

Dadurch wird genau wie die eigenen Collections auch die *Meteor.users*-Collection vor clientseitigen Zugriffen in THMcards geschützt.

### 7.2.3. Rollen

Bei Anwendungen wie THMcards, die aus einem Front-End- und Back-End-Bereich bestehen, ist die Verwendung eines geeigneten Rollensystems unabdingbar. Durch die Einführung von Rollen lassen sich verschiedene Benutzertypen in THMcards erstellen und kontrollieren, auf welche Bereiche der Anwendung diese Zugriff haben. Dazu wird in THMcards das von Meteor angebotene Package *alan-ning:roles* verwendet, das die Möglichkeit bereitstellt, Rollen zu definieren und damit bestimmte Funktionalitäten der Anwendung einzuschränken. Das Package bietet damit eine einfache Lösung, um dem Datenschutzgrundsatz der *Zugangskontrolle* in THMcards nachzukommen. Folgende Rollen sind in Bezug auf die Datensicherheit in THMcards definiert:

#### Standard

Der Standard-User repräsentiert den normalen Benutzer des Systems. Dieser besitzt keine besonderen Zugriffsrechte und kann auf alle grundlegenden Funktionen von THMcards zugreifen, wie es auch in der ersten Version möglich war. Dazu gehören die Aktionen im Front-End wie das Erstellen von privaten und öffentlichen Kartensätzen sowie das Lernen von eigenen und als *Free* gekennzeichneten Kartensätzen.

#### Admin

Der Admin-User ist der Super-Administrator in THMcards und besitzt uneingeschränkte Zugriffsrechte im System. Dieser hat Zugriff auf alle Funktionen und kann neben dem Front-End auf den Back-End-Bereich von THMcards zugreifen, um Inhalte wie Lernkartensätze, Lernkarten und Benutzer zu verwalten. Darüber hinaus besitzt ein Admin-User die Rechte Editor-User zu bestimmen, Benutzer von THMcards zu blockieren und im System integrierte Benachrichtigungen an die Benutzer zu versenden.

### Editor

Ein Editor-User besitzt ähnliche Zugriffsrechte wie der Admin-User. Auch er hat Zugang zu allen Funktionen und kann das Front-End und Back-End von THMcards nutzen, um Inhalte zu verwalten. Im Gegensatz zum Admin-User kann dieser jedoch keine weiteren Editor-User im System bestimmen. Um die Rollenhierarchie zu wahren, ist es ihm außerdem nicht möglich, den Admin-User aus THMcards zu entfernen.

### Blocked

Verstößt ein Benutzer gegen die Allgemeinen Geschäftsbedingungen von THMcards, erhält dieser von einem Admin- oder Editor-User die Rolle *Blocked* und wird damit von dem System blockiert. Dieser kann weder Funktionen von THMcards nutzen, noch auf die Daten von THMcards zugreifen. Einem blockierten Benutzer in THMcards wird statt dem eigentlichen Inhalt eine Fehlerseite mit einer entsprechenden Begründung angezeigt (siehe Abbildung 7.8).

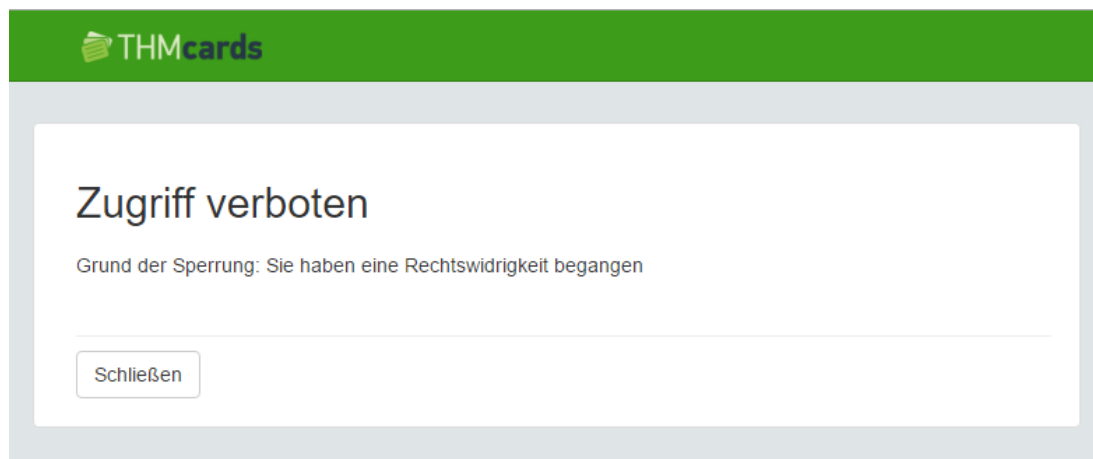


Abbildung 7.8.: Fehlerseite in THMcards

### FirstLogin

Meldet sich ein Benutzer zum ersten Mal in THMcards an, wird diesem die Rolle *FirstLogin* zugewiesen. Dadurch wird der Benutzer aufgefordert, die Allgemeinen Geschäftsbedingungen und Datenschutzerklärung von THMcards zu akzeptieren, bevor dieser die Funktionen des Systems nutzen kann. Werden die Bedingungen

vom Benutzer verweigert, kann dieser THMcards nicht verwenden.

Neben den in dieser Masterarbeit vorgestellten Rollen werden in THMcards weitere Rollen in Bezug auf das Geschäftsmodell definiert. Eine genaue Beschreibung ist in der Masterarbeit *Erfolgreiche Monetarisierung einer Open-Source Webanwendung durch die Integration eines innovativen Geschäftsmodells* von Marius Trautrimis zu finden [Tra16].

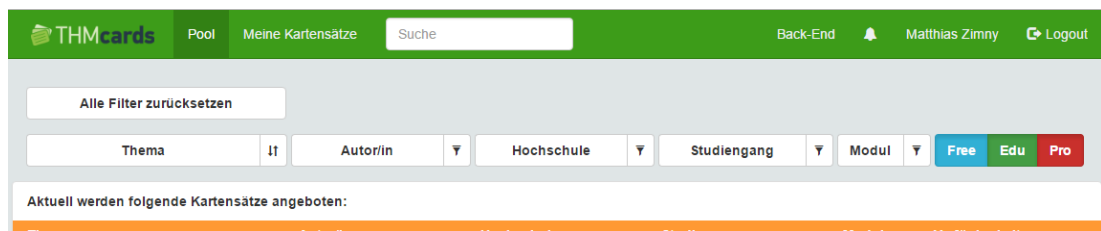


Abbildung 7.9.: Menüleiste eines Admin-User in THMcards

Mithilfe des Packages *alanning:roles* lässt sich in THMcards durch den Template-Helper `isInRole` bestimmen, welche Teile der Webseite für Benutzer in einer bestimmten Rolle geladen oder welche Funktionen zur Verfügung gestellt werden sollen. Ist beispielsweise in THMcards ein Benutzer als Admin-User oder Editor-User angemeldet, wird ihm in der Menüleiste die Schaltfläche *Back-End* angezeigt (siehe Abbildung 7.9). Das wird mit folgendem Codeschnipsel realisiert:

```

1 {{#if isInRole 'admin, editor'}}
2   <li id="navbar-adminpanel">
3     <a href="{{pathFor route='admin_dashboard'}}"
4       id="adminpanel">
5       {{_ "navbar-collapse.backend"}} //Back-End
6     </a>
7   </li>
8 {{/if}}
```

Listing 7.7: Einsatz vom `isInRole`-Helper in THMcards

Darüber hinaus können durch den Einsatz von Rollen Server-Methoden um zusätzliche Sicherheitsmaßnahmen erweitert werden. In THMcards wird damit der Zugriff auf bestimmte Methoden im Server für jeweilige Benutzer verwehrt. Server-Methoden, die nur von einem Admin- oder Editor-User aufgerufen werden sollen, werden in THMcards entsprechend mit folgender if-Anweisung geschützt:

```
1 if (!Roles.userIsInRole(this.userId, ['admin', 'editor'])) {  
2   throw new Meteor.Error("not-authorized");  
3 }
```

Listing 7.8: Absicherung der Server-Methoden durch Rollen in THMcards

### 7.2.4. Settings-Datei

In THMcards werden Informationen wie geheime API-Keys oder der Admin-User des Systems festgelegt. Solche Informationen sollten unter keinen Umständen direkt im Code gespeichert werden, da die Gefahr besteht, dass ein Benutzer diese auslesen und sich zu eigen machen kann, was ein Sicherheitsrisiko darstellt. Die *settings*-Datei erlaubt es, diese Informationen vom Code zu trennen und in eine Sammlung von **public**- und **private**-Werten in der Anwendung im *JSON*-Format zu speichern. Werte, die unter **public** gespeichert werden, sind im Client und im Server von THMcards zugänglich, Werte, die nur vom Server aus aufgerufen werden sollen, sind in **private** abgelegt.

Der Einsatz der Datei soll nochmals in einem Beispiel verdeutlicht werden. Der Admin-User wird unter THMcars wie folgt in der *settings*-Datei definiert:

```
1 "private": {  
2   "admin": {  
3     "name": "Administrator"  
4   }  
5 }
```

Listing 7.9: Festlegung des Admin-Users in THMcards

Im Server-Code von THMcards kann nun auf den Wert über die Variable `Meteor.settings.admin.name` zugegriffen werden.

## 8. Abschluss

In diesem Kapitel wird zunächst eine Zusammenfassung der gesamten Arbeit gegeben. Abschließend werden die gewonnenen Erkenntnisse der Arbeit in einem Fazit bewertet und anhand eines Ausblicks mögliche Verbesserungen der Lernkartenplattform THMcards bezüglich des Datenschutzes und der Datensicherheit aufgeführt.

### 8.1. Zusammenfassung

In dieser Masterarbeit wurde eine Konzeption und Implementierung geeigneter Datenschutz- und Datensicherheitsmaßnahmen in E-Learning Systemen erarbeitet, deren Ziel es ist, aufzuzeigen, welche datenschutzrechtlichen Aspekte durch ein solches System einzuhalten und welche Maßnahmen erforderlich sind, um den Schutz der Daten zu gewährleisten. Im Rahmen dieser Arbeit konnten diese durch die Implementierung in die Lernkartenplattform THMcards, basierend auf den aktuellen Technologien, optimiert werden.

In Kapitel 2 wurden dem Leser die Komponenten eines E-Learning Systems, die aus den Funktionsbereichen des eigentlichen Systems und den dazugehörigen technischen Bausteinen bestehen, näher gebracht. Durch die aufgezeigten Komponenten konnten mögliche Schwachstellen eines E-Learning Systems veranschaulicht werden, die den Prozess unterstützen sollten, geeignete Datenschutz- und Datensicherheitsmaßnahmen für ein solches System auszuarbeiten.

Anhand der Grundlage der Komponenten wurden in Kapitel 3 die vorgeschriebenen Gesetze und rechtlichen Rahmenbedingungen aus der Perspektive des Bundesdatenschutzgesetzes, die bei der Entwicklung von E-Learning Systemen einzuhalten sind, näher gebracht. Hierzu wurde dem Leser der Begriff der personenbezogenen Daten erläutert und deren Verwaltungsmöglichkeiten in einem E-Learning System veranschaulicht. Anschließend wurden die Grundsätze des Datenschutzes, die bei der Verwendung von personenbezogenen Daten einzuhal-

ten sind, festgelegt. Die Einhaltung dieser Grundsätze stellt sicher, dass ein E-Learning System alle nötigen rechtlichen Maßnahmen umsetzt, um personenbezogene Daten nutzen zu dürfen. Darüber hinaus wurden durch weitere datenschutzrechtliche Aspekte des Telemediengesetzes Informationspflichten aufgezeigt, die durch das BDSG nicht abgedeckt worden sind. Darunter fällt die Pflicht, ein Impressum zu führen oder Benutzer auf den Einsatz von Cookies hinzuweisen.

Im Anschluss wurde in Kapitel 4 gezeigt, welche Haftung Online-Dienste übernehmen, wenn rechtswidrige Inhalte eingestellt werden. Dazu wurden die verschiedenen Unterteilungen von Internetdiensteanbietern des Telemediengesetzes aufgezeigt, die den Umfang der Haftung bestimmen. Außerdem war es wichtig zu klären, mit welchen Rechtsverletzungen Internetdiensteanbieter eines E-Learning Systems konfrontiert werden können, für die sich der Provider verantworten muss. Anschließend wurde für Projekte wie THMcards, die ihren Speicherplatz für fremde Inhalte zur Verfügung stellen, die Verantwortlichkeiten eines Host-Providers bei der Kenntnisnahme eines rechtswidrigen Inhaltes definiert.

Kapitel 5 deckte die Datensicherheitsmaßnahmen ab, die notwendig sind, um die vorgestellten Komponenten eines E-Learning Systems zu schützen. Hierzu wurden die Datenschutzziele der Datensicherheit betrachtet, um dem Leser Angriffe, die möglicherweise auf Daten oder Informationen eines E-Learning Systems erfolgen, besser veranschaulichen zu können. Es gilt das System vor solchen beabsichtigten Angriffen zu schützen, indem die permanente Sicherheit durch die Einhaltung der Schutzziele, durch die kontinuierliche Überwachung und Weiterentwicklung der technischen und organisatorischen Datensicherheitsmaßnahmen garantiert wird. Des Weiteren wurde für den Aufbau eines effektiven Schutzes verschiedene Angriffsmethoden auf ein E-Learning System aufgezeigt. Eine Vorbereitung auf die Angriffsmethoden ermöglicht es, gezielt Sicherheitslücken in einem System zu schließen und damit Hackern einen Angriff zu erschweren.

Kapitel 6 wurde die Lernkartenplattform THMcards vorgestellt und somit die nötige Grundlage geschaffen, die ausgearbeiteten Datenschutz- und Datensicherheitsmaßnahmen in die Plattform zu implementieren. Dazu wurde auf die Re-Implementierung von THMcards eingegangen, um dem Leser die Wahl des JavaScript-Frameworks zu verdeutlichen und einen groben Einblick in die Änderungen in der Codestruktur von THMcards zu verschaffen. Anschließend wurden die technischen Aspekte von THMcards erläutert.

In Kapitel 7 wurde die Implementierung der vorgestellten Datenschutz- und Datensicherheitsmaßnahmen in THMcards behandelt. Hierzu wurde in Bezug auf die Datenschutzmaßnahmen zunächst eine Übersicht über die Informationspflichten, das Einwilligungsverfahren und Urheberrechte, die in das Projekt eingearbeitet wurden, gegeben. Des Weiteren ermöglicht das implementierte Back-End Lerninhalte, Benutzer und Zugriffsrechte von THMcards zu verwalten und damit den Verantwortlichkeiten eines Host-Providers nachzukommen und effektiv Rechtswidrigkeiten entgegenzuwirken. Um in THMcards die Sicherheit der Daten zu gewährleisten, wurden von dem JavaScript-Framework Meteor vorgegebene Maßnahmen eingepflegt. THMcards wird durch geeignete Packages, durch die Absicherung der Datenbank und durch ein rollenbasiertes Berechtigungskonzept in der Datensicherheit verbessert.

## 8.2. Fazit und Ausblick

E-Learning Systeme finden immer häufiger Einsatz, um den Lernprozess durch digital gespeicherte Lerninhalte oder softwareunterstützte Lernumgebungen zu unterstützen. Benutzer profitieren hierbei von der starken Flexibilität bei der Wahl von Lernorten und Lernzeiten im Vergleich zum herkömmlichen Lernen. Es entsteht die Möglichkeit, mehr Studenten und Schüler effektiver mit weniger Aufwand mit Lerninhalten zu fördern. Dadurch werden inzwischen an nahezu allen Hochschulen für die Durchführung des Lernens internetbasierte Systeme, die das E-Learning unterstützen, eingesetzt. Hierbei werden bei der Benutzung eines E-Learning Systems Daten, wie personenbezogene Informationen, Lernfortschritte, erreichte Zertifikate oder Lerninhalte, gesammelt, um einen reibungslosen Ablauf und die technische Überwachung des Systems zu ermöglichen.

Im Rahmen dieser Masterarbeit ist deutlich geworden, welche Problematik durch die Menge der gesammelten Daten entstehen kann. Es sind nicht nur die Anforderungen an die Datensicherheit, die bei der Verarbeitung von Daten zu berücksichtigen sind. Auch datenschutzrechtliche Aspekte müssen strengstens beachtet werden, vor allem, wenn die zu verarbeitenden Daten personenbezogen sind. Durch die vorgestellten Datenschutz- und Datensicherheitsmaßnahmen wurden jedoch Vorkehrungen getroffen, diesen Problemen entgegenzuwirken. Diese bieten bei der Entwicklung eines E-Learning Systems ein Grundgerüst, dass

die Sicherheit von Daten gewährleistet und dabei die rechtlichen Aspekte des Datenschutzes bei der Nutzung und Verarbeitung von Daten beachtet.

Die ausgearbeiteten Maßnahmen wurden erfolgreich unter Berücksichtigung der Meteor-Richtlinien in die Lernkartenplattform THMcards implementiert. Dadurch wurde die Plattform vor allem in der Datensicherheit optimiert. Angreifer haben nicht mehr die Möglichkeit auf die Datenbank zuzugreifen, um Lerninhalte zu manipulieren oder zu entwenden, personenbezogene Daten von Benutzern zu stehlen oder Zugriffsrechte zu manipulieren. Folglich wurde die Vertrauenswürdigkeit von THMcards erhöht, sodass Benutzer mit gutem Gewissen die Plattform nutzen können, ohne den Verlust ihrer Lerninhalte oder personenbezogenen Daten zu befürchten. Auch die Verantwortlichkeiten eines Host-Providers wurden berücksichtigt, sodass durch das implementierte Back-End und ein integriertes Benachrichtigungssystem effizient auf Rechtswidrigkeiten, sei es eine Persönlichkeits- oder Urheberrechtsverletzung in Lerninhalten oder der Verstoß gegen die AGB, in THMcards reagiert werden kann. Maßnahmen wie die Einwilligungspflicht der AGB und Datenschutzerklärung, das Führen eines Impressum oder die Festlegung von Urheberrechten decken die datenschutzrechtlichen Aspekte von THMcards ab, die in einem E-Learning System eingehalten werden müssen.

Jedoch ist es mit der einmaligen Einarbeitung der Sicherheitsmaßnahmen in THMcards nicht getan. Die permanente Sicherheit der Plattform kann nur gewährleistet werden, wenn eine kontinuierliche Überwachung und Weiterentwicklung der Datensicherheitsmaßnahmen stattfindet. Dazu soll für die Weiterentwicklung von THMcards der vorgestellte PDCA-Zyklus genutzt werden, um die Schutzziele des Projektes sicherzustellen.

Ein Ansatzpunkt, den Schutz der gespeicherten Daten in der Datenbank zu gewährleisten, ist die erneute Überprüfung der Servermethoden, um sicherzustellen, dass nur die vorgesehenen Benutzer Zugriffsrechte auf gewünschte Servermethoden haben. Des Weiteren kann die Validierung durch das *check*-Package verbessert werden. Neben der Überprüfung des Datentypes kann `check()` eingehende Daten auch durch selbst geschriebene Funktionen validieren. Dadurch kann genauer spezifiziert werden, welche Daten in der Server-Methode verarbeitet werden dürfen.

Durch das Package *Browser-Policy* wurde das Sicherheitskonzept Content Security Policy in THMcards eingebunden, um Attacken wie Cross-Site-Scripting

oder Code Injection zu verhindern. Dazu wurde grundlegend festgelegt, welche Aktionen in der Anwendung blockiert werden sollen und anschließend Ausnahmen definiert. Sollen aus Lernzwecken weitere Webseiten zugelassen werden, um externe Skripte in THMcards zu laden, müssen diese vorher als Ausnahme in der `browser-policy.js` festgelegt werden.

Aufgrund der begrenzten Zeit konnten die Allgemeinen Geschäftsbedingungen, die Datenschutzerklärung und das Verzeichnissverzeichnis nicht auf die rechtliche Richtigkeit durch einen Anwalt oder Datenschutzbeauftragten kontrolliert werden. Eine rechtssichere Formulierung kann somit nicht garantiert werden, sodass die Nutzung der formulierten Texte in der Produktionsphase dringend abzurufen ist, um Abmahnungen zu vermeiden. Sie dienen lediglich als Grundlage, um im weiteren Verlauf des Projektes Arbeitszeit sparen zu können. Aus diesem Grund ist zu empfehlen, die verfassten Texte durch die professionelle Unterstützung eines Anwalts oder Datenschutzbeauftragten zu überprüfen.

Mit der Einarbeitung der Datenschutz- und Datensicherheitsmaßnahmen wurde ein wichtiger Meilenstein in der Entwicklung von THMcards gelegt. Ein abgesichertes E-Learning System erhöht nicht nur das Vertrauen der Benutzer des Systems, sondern lockt auch neue an. Besteht die Möglichkeit, sich zwischen mehreren E-Learning Systemen zu entscheiden, fällt die Wahl logischerweise auf ein solches, das die Sicherheit der Daten verspricht und die Daten vertraulich unter den rechtlichen Aspekten des Datenschutzes behandelt. THMcards kann somit an Popularität gewinnen und dadurch das Interesse eines größeren Benutzerkreises wecken. Darüber hinaus wurde die Lernqualität erhöht. Angriffe, die das System in der Verfügbarkeit einschränken oder durch die Entwendung von Lernmaterial Lernprozesse unterbinden können, werden durch die eingearbeiteten Datensicherheitsmaßnahmen reduziert oder sogar verhindert.

# Literaturverzeichnis

- [2BI] 2BIT: *SPA (Single Page Application) – der perfekte Begleiter für klassische Softwareanwendungen*. <http://www.2bit.ch/single-page-application/>, Abruf: 20.09.2016
- [act] *Muster: Verfahrensverzeichnis*. activeMind AG. <https://www.activemind.de/datenschutz/dokumente/verfahrensverzeichnis/>, Abruf: 14.11.2016
- [Bäu00] BÄUMLER, Helmut: *E-Privacy - Datenschutz im Internet*. Springer Vieweg, 2000
- [BDSa] *Bundesdatenschutzgesetz (BDSG) § 1 Zweck und Anwendungsbereich des Gesetzes*. Bundesministerium der Justiz und für Verbraucherschutz. [https://www.gesetze-im-internet.de/bdsg\\_1990/\\_1.html](https://www.gesetze-im-internet.de/bdsg_1990/_1.html), Abruf: 22.11.2016
- [BDSb] *Bundesdatenschutzgesetz (BDSG) § 3 Weitere Begriffsbestimmungen*. Bundesministerium der Justiz und für Verbraucherschutz. [https://www.gesetze-im-internet.de/bdsg\\_1990/\\_3.html](https://www.gesetze-im-internet.de/bdsg_1990/_3.html), Abruf: 22.11.2016
- [Bet] *Haftung für Hyperlinks*. Bettinger Rechtsanwälte - Patentanwälte. <http://www.bettinger.de/infothek/it-und-medienrecht/internetrecht/internetrecht-a-z/haftung/haftung-fuer-hyperlinks/index.html>, Abruf: 22.11.2016
- [bit16] *Datenschutz, IT-Sicherheit & Datensicherheit - ein Wegweiser*. bitbase group. <http://bitbasegroup.com/aktuelles/datenschutz-it-sicherheit-datensicherheit>. Version: 2016, Abruf: 21.11.2016
- [bpb13] *Lizenzen: Klassiker und Alternativen*. bpb: Bundeszentrale für politische Bildung, 2013
- [Bre] *Einführung Telemediengesetz Teil 1 Telemedien*. Brennecke & Partner. <https://www.brennecke.pro/87349/Einfuehrung-Telemediengesetz-Teil-1-Telemedien->, Abruf: 21.11.2016
- [Brü15] BRÜNTJE, Marlon: *Single-Page-Application - Individuelle Web-Anwendung für Unternehmen*. <http://www.flyacts.com/blog/single-page-application-individuelle-web-anwendung-fuer-unternehmen/>. Version: 2015, Abruf: 20.09.2016

- [Bsia] *Aktive Inhalte.* BSI. [https://www.bsi.bund.de/DE/Themen/StandardsKriterien/ISi-Reihe/ISi-Web-Server/aktive\\_inhalte/aktive\\_inhalte\\_node.html](https://www.bsi.bund.de/DE/Themen/StandardsKriterien/ISi-Reihe/ISi-Web-Server/aktive_inhalte/aktive_inhalte_node.html), Abruf: 18.11.2016
- [Bsiib] *Aktive Inhalte.* BSI. [https://www.bsi.bund.de/DE/Themen/StandardsKriterien/ISi-Reihe/ISi-Web-Server/aktive\\_inhalte/definitionen/definitionen\\_node.html](https://www.bsi.bund.de/DE/Themen/StandardsKriterien/ISi-Reihe/ISi-Web-Server/aktive_inhalte/definitionen/definitionen_node.html), Abruf: 18.11.2016
- [Bsi11a] *DNS-Flooding - Denial-of-Service.* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/g/g05/g05151.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/g/g05/g05151.html). Version: 2011, Abruf: 09.11.2016
- [Bsi11b] *Verhinderung von Diensten (Denial of Service).* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/g/g00/g00040.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/g/g00/g00040.html). Version: 2011, Abruf: 09.11.2016
- [Bsi13a] *Cross-Site Scripting (XSS).* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/g/g05/g05170.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/g/g05/g05170.html). Version: 2013, Abruf: 09.11.2016
- [Bsi13b] *Datenschutz.* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/baust/b01/b01005.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/baust/b01/b01005.html). Version: 2013, Abruf: 21.11.2016
- [Bsi13c] *Festlegung von technisch-organisatorischen Maßnahmen entsprechend dem Stand der Technik bei der Verarbeitung personenbezogener Daten.* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/m/m02/m02505.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/m/m02/m02505.html). Version: 2013, Abruf: 18.11.2016
- [Bsi13d] *Verhinderung von Cross-Site Request Forgery (CSRF, XSRF, Session Riding).* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/m/m04/m04403.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/m/m04/m04403.html). Version: 2013, Abruf: 09.11.2016
- [Bsi16] *Verhinderung der Blockade von Ressourcen (DoS) bei Webanwendungen und Web-Services.* BSI. [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/m/m04/m04405.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/m/m04/m04405.html). Version: 2016, Abruf: 09.11.2016
- [bso] *BSON Description.* <http://bsonspec.org/>, Abruf: 19.09.2016
- [Buc03] BUCKSTEEG, Andreas: *Methoden und Techniken zur Sicherstellung des vertrauenswürdigen Managements von Benutzerprofildaten in vernetzten Anwendungen*, Technische Universität München, Diplomarbeit, 2003

- [Cre] CREMER, Gino: *Einsteiger-Tipps – Unterschied zwischen statischen und dynamischen CMS – Websites*. <https://pixelbar.be/blog/einsteiger-tipps-unterschied-statisch-dynamisch-cms/>, Abruf: 18.11.2016
- [DAS15] *Haftung des Providers*. D.A.S. Rechtsschutz der ERGO. <https://www.das.de/de/rechtsportal/internetrecht/rufschadigung/haftung.aspx>. Version: 2015, Abruf: 22.11.2016
- [Data] <http://www.luo-darmstadt.de/wiki2/lib/exe/fetch.php?media=db:datenbanksystem.png>
- [Datb] *Datenbank – Was ist eine Datenbank?* Datenbanken verstehen. <http://www.datenbanken-verstehen.de/datenbank-grundlagen/datenbank/>, Abruf: 18.11.2016
- [Datc] *Datenbanksystem*. Datenbanken verstehen. <http://www.datenbanken-verstehen.de/lexikon/datenbanksystem/>, Abruf: 18.11.2016
- [Datd] *Die elektronische Einwilligung – Gleicher Name, anderes Kind!* Datenschutzbeauftragter Info. <https://www.datenschutzbeauftragter-info.de/die-elektronische-einwilligung-gleicher-name-anderes-kind/>, Abruf: 21.11.2016
- [Date] *Informationssicherheit und Datenschutz*. Datenschutzbeauftragter Info. <https://www.datenschutzbeauftragter-info.de/fachbeitraege/informationssicherheit-und-datenschutz/>, Abruf: 14.11.2016
- [Datf] *Verfahrensverzeichnis*. Datenschutzbeauftragter Info. <https://www.datenschutzbeauftragter-info.de/fachbeitraege/verfahrensverzeichnis/>, Abruf: 14.11.2016
- [Dat14] *Begriff und Geschichte des Datenschutzes*. Datenschutzbeauftragter Info. <https://www.datenschutzbeauftragter-info.de/begriff-und-geschichte-des-datenschutzes/>. Version: 2014, Abruf: 21.11.2016
- [Dat15a] *Anonymisierung und Pseudonymisierung von Kundendaten*. Datenschutz-Praxis. <https://www.datenschutz-praxis.de/fachartikel/anonymisierung-und-pseudonymisierung-von-kundendaten/>. Version: 2015, Abruf: 21.11.2016
- [Dat15b] *Personenbezogene Daten*. Datenschutz-Praxis. <https://www.datenschutz-praxis.de/fachartikel/personenbezogene-daten/>. Version: 2015, Abruf: 21.11.2016

- [Dat16] *Unterschied zw. IT-Sicherheit, Datensicherheit, Datenschutz & Informationssicherheit.* Datenschutzbeauftragter Info. <https://www.datenschutzbeauftragter-info.de/unterschiede-zwischen-datenschutz-datensicherheit-informationssicherheit-oder-it-sicherheit/>. Version: 2016, Abruf: 21.11.2016
- [DE16] DB-ENGINES: *DB-Engines Ranking - Trend der Document Stores Popularität.* [http://db-engines.com/de/ranking\\_trend/document+store](http://db-engines.com/de/ranking_trend/document+store). Version: 2016, Abruf: 29.09.2016
- [DFN15] *Leitfaden über den Umgang mit Social Media.* Forschungsstelle Recht im DFN, 2015
- [DHF07] DR. HANNES FEDERRATH, Prof. Dr. Andreas P.: *Datenschutz und Datensicherheit 6. Auflage.* Springer, 2007
- [DPN] *Personenbezogene Daten.* DPN - Datenschutz & Informationssicherheit. <http://www.dpn-datenschutz.de/datenschutz/personenbezogene-daten/>, Abruf: 21.11.2016
- [Eil15] EILERS, Carsten: *Cross-Site Scripting im Überblick, Teil 4: DOM-basiertes XSS.* <http://www.ceilers-news.de/serendipity/633-Cross-Site-Scripting-im-UEberblick,-Teil-4-DOM-basiertes-XSS.html>. Version: 2015, Abruf: 09.11.2016
- [Ele] *DoS - Denial of Service.* Elektronik Kompendium. <http://www.elektronik-kompendium.de/sites/net/1412091.htm>, Abruf: 09.11.2016
- [Gag15] GAGERN, Wolfram von: *Bundesdatenschutzgesetz: Wann Sie personenbezogene Daten erheben, speichern und nutzen dürfen.* <https://www.wirtschaftswissen.de/unternehmensgruendung-und-fuehrung/datenschutz/kundendatenschutz/bundesdatenschutzgesetz-wann-sie-personenbezogene-daten-erheben-speichern-und-nutzen-duerfen/>. Version: 2015, Abruf: 21.11.2016
- [Glo] *Cross Site Scripting.* Glossar HS-Augsburg. [https://glossar.hs-augsburg.de/Cross\\_Site\\_Scripting](https://glossar.hs-augsburg.de/Cross_Site_Scripting), Abruf: 09.11.2016
- [Glo15] GLOVER, Ryan: *Using the Collection2 Package.* <https://themeteorchef.com/snippets/using-the-collection2-package/>. Version: 2015, Abruf: 08.11.2016
- [Gra11] GRAF, Thomas: *Webhosting.* <http://pixelfolk.net/webhosting>. Version: 2011, Abruf: 18.11.2016

- [Gru] *Allgemeines Persönlichkeitsrecht. Grundrechtsschutz - Ihre Grundrechte in Deutschland und Europa.* <https://www.grundrechtsschutz.de/allgemein/allgemeines-personlichkeitsrecht-260>, Abruf: 22.11.2016
  
- [Grü14] GRÜNER, Sebastian: *Node.js-Fork für mehr Freiheiten.* Golem. <http://www.golem.de/news/io-js-node-js-fork-fuer-mehr-freiheiten-1412-110994.html>. Version: 2014, Abruf: 21.09.2016
  
- [Hec16] HECKER, Peter: *JavaScript goes Enterprise - Node.js-Anwendungen mit Visual Studio und den Node.js-Tools entwickeln.* SlideShare. <http://de.slideshare.net/phecker65/javascript-goes-enterprise-nodejsanwendungen-mit-visual-studio-und-den-nodejstools-entwickeln>. Version: 2016, Abruf: 29.09.2016
  
- [Hoe16] HOEREN, Prof. Dr. T.: *Internetrecht.* Universität Münster, 2016
  
- [Hub16] HUBER, Daniel: *Verwendung von Cookies nur noch bei ausdrücklicher Einwilligung der Nutzer?* <https://www.it-recht-kanzlei.de/cookies-einwilligung-datenschutzerklaerung.html>. Version: 2016, Abruf: 21.11.2016
  
- [iRi09] *Fremde Inhalte auf eigenen Seiten.* iRights info, 2009
  
- [ItR] *Haftung im Internet.* IT-Rechtsinfo.de. <http://www.it-rechtsinfo.de/haftung-im-internet/>, Abruf: 22.11.2016
  
- [Jae11] *BGH konkretisiert die Verantwortlichkeit von Host Providern für persönlichkeitsrechtsverletzende Blog-Einträge (BGH, Urteil vom 25.10. 2011, Az.: VI ZR 93/10).* Jaeschke Rechtsanwalt Gießen. <https://www.ipjaeschke.de/rechtsanwalt-giessen/markenschutz-markenrecht-nachrichten/163-bgh-konkretisiert-die-verantwortlichkeit-von-hostprovidern-fuer-persoendlichkeitsrechtsverletzende-blog-eintraege-bgh-urteil-vom-2510-2011-az-vi-zr-9310.html>. Version: 2011, Abruf: 22.11.2016
  
- [JH] JAN HANSEN, Nadine H.: *Datenschutz beim E-Learning - Zum Verhältnis von Kontrolle und Vertrauen in der Informationsgesellschaft*
  
- [Jur] *Telemediengesetz.* Jura Forum. <http://www.juraforum.de/lexikon/telemediengesetz>, Abruf: 21.11.2016
  
- [jus] *Haftung für Links auf fremde Inhalte.* just law - Rechtsanwälte. <http://www.internetrecht.justlaw.de/haftung-links.htm>, Abruf: 22.11.2016

- [Kam14] KAMMER, Jan C.: *Implementierung von Spaced Repetition Algorithmen zur effektiven Abfrage von Lernkaten innerhalb der eLearning Plattform THMcards*, Technische Hochschule Mittelhessen, Diplomarbeit, 2014
- [Kap07] KAPPES, Prof. Dr. M.: *Netzwerk- und Datensicherheit*. Springer, 2007
- [Kat07] KATZLINGER, Elisabeth: Big Brother beim Lernen: Privatsphäre und Datenschutz in Lernplattformen. (2007)
- [Kei] KEIL, Oliver: *Grundsätze des Datenschutzrechts*
- [Kla02] KLAU, Peter: *Hacker, Cracker, Datenräuber - Datenschutz selbst realisieren, akute Gefahren erkennen, jetzt Abhilfe schaffen*. Springer Vieweg, 2002
- [Kna13] KNAPP, Daniel: *Implementierung von Spielmechaniken zur Steigerung der Lernmotivation von Studierenden am Beispiel der Lernkarten Plattform THMcards*, Technische Hochschule Mittelhessen, Diplomarbeit, 2013
- [Kre07] KREUTZER, Till: *Rechtsfragen bei E-Learning – Ein Praxisleitfaden*. 2007
- [LDI] *Welche Rechtsvorschriften regeln den Datenschutz?* LDI - Landesbeauftragte für Datenschutz und Informationsfreiheit Nordrhein-Westfalen. [https://www.ldi.nrw.de/mainmenu\\_Datenschutz/Inhalt/FAQ/Rechtsvorschriften.php](https://www.ldi.nrw.de/mainmenu_Datenschutz/Inhalt/FAQ/Rechtsvorschriften.php), Abruf: 21.11.2016
- [Lie13] LIENAU, Matthias: *Meteor: Einblick in die Full-Stack-JavaScript-Plattform für das Echtzeit-Web*. <http://t3n.de/magazin/meteor-full-stack-javascript-plattform-fuer-echtzeit-web-234154/>. Version: 2013, Abruf: 26.09.2016
- [Mar16] MARTIN, Nick: *Tuning Meteor Mongo Livedata for Scalability*. Meteor. <http://info.meteor.com/blog/tuning-meteor-mongo-livedata-for-scalability>. Version: 2016, Abruf: 30.09.2016
- [Meta] *Atmosphere vs. npm*. Meteor. <https://guide.meteor.com/atmosphere-vs-npm.html>, Abruf: 30.09.2016
- [Metb] *Defining a schema*. Meteor. <https://guide.meteor.com/collections.html#schemas>, Abruf: 30.09.2016
- [Metc] *Security - How to secure your Meteor app*. Meteor. <https://guide.meteor.com/security.html>, Abruf: 09.11.2016
- [Metd] METEOR: *Introducing Meteor API Docs*. <http://docs.meteor.com/#what-is-meteor>, Abruf: 26.09.2016

- [Mon13] *MongoDB – Die dokumentorientierte JSON-Datenbank*. NodeCode. <http://nodecode.de/mongodb>. Version: 2013, Abruf: 30.09.2016
- [Mon15] MONGODB: *Meteor: Build iOS and Android Apps that are a Delight to Use*. <https://www.mongodb.com/blog/post/meteor-build-ios-and-android-apps-are-delight-use>. Version: 2015, Abruf: 26.09.2016
- [Mül10a] MÜLLER, David: *Angriffe auf Webanwendungen – Teil 1: XSS (+Beispielangriff)*. <https://d-mueller.de/blog/angriffe-auf-webanwendungen-teil-1-xss-beispielangriff/>. Version: 2010, Abruf: 09.11.2016
- [Mül10b] MÜLLER, David: *Angriffe auf Webanwendungen – Teil 2: Session-Hijacking und Session-Fixation*. <https://d-mueller.de/blog/angriffe-auf-webanwendungen-teil-2-session-highjacking-und-session-fixation/>. Version: 2010, Abruf: 09.11.2016
- [Nag08] NAGEL, Arndt J.: *Providerhaftung nach dem Telemediengesetz*. <http://www.it-recht-kanzlei.de/providerhaftung-stoererhaftung.html>. Version: 2008, Abruf: 22.11.2016
- [New16] NEWMAN, Ben: *Announcing Meteor 1.4.1: Node 4.5.0, Faster Package Downloads, + More*. Meteor Blog. <http://info.meteor.com/blog/announcing-meteor-1.4.1>. Version: 2016, Abruf: 21.09.2016
- [NK13] NIKO KÖBLER, Heiko S.: *Einstieg in die Entwicklung von Web-Apps mit Meteor*. <http://www.heise.de/developer/artikel/Einstieg-in-die-Entwicklung-von-Web-Apps-mit-Meteor-1949891.html>. Version: 2013, Abruf: 26.09.2016
- [Nod] [https://d262ilb51hltx0.cloudfront.net/max/800/1\\*Te1HNFdDpgCn9ImQivtwRA.jpeg](https://d262ilb51hltx0.cloudfront.net/max/800/1*Te1HNFdDpgCn9ImQivtwRA.jpeg)
- [Nod15] *Node v4.0.0 (Current)*. <https://nodejs.org/en/blog/release/v4.0.0/>. Version: 2015, Abruf: 21.09.2016
- [Ott] OTT, Dr. S.: *Die Impressumspflicht nach § 5 TMG / § 55 RStV*. <http://linksandlaw.info/Impressumspflicht-Notwendige-Angaben.html>, Abruf: 21.11.2016
- [Owa] *SQL and NoSQL Injection*. OWASP NodeGoat Tutorial. [https://ckarande.gitbooks.io/owasp-nodegoat-tutorial/content/tutorial/a1\\_-\\_sql\\_and\\_nosql\\_injection.html](https://ckarande.gitbooks.io/owasp-nodegoat-tutorial/content/tutorial/a1_-_sql_and_nosql_injection.html), Abruf: 09.11.2016
- [Owe15] OWENS, Josh: *Why Meteor?* <http://whymeteor.com/>. Version: 2015, Abruf: 26.09.2016

- [PDC] [https://beratungniedersachsen.files.wordpress.com/2012/06/deming\\_kreis.gif](https://beratungniedersachsen.files.wordpress.com/2012/06/deming_kreis.gif)
- [Pro] *Personenbezogene Daten gemäß § 3 BDSG*. procado Consulting, IT- & Medienservice GmbH. <https://www.procado.de/datenschutz-lexikon/895/Personenbezogene%20Daten%20gem%C3%A4%C3%9F%20C2%A7%203%20BDSG.html>, Abruf: 21.11.2016
- [QC13] QUIBLEDEY-CIRKEL, Prof. Dr. K.: Leitners Lernkartei revisited: Gegen das Aufschieben und Vergessen im Studium. (2013). <https://dl.dropboxusercontent.com/u/87040050/Fellowship/Leitners%20Lernkartei%20revisited-%20Gegen%20das%20Aufschieben%20und%20Vergessen%20im%20Studium.pdf>
- [Ros15] ROST, Jennifer: *Das Rätselraten um die Cookie-Richtlinie*. <http://shopbetreiber-blog.de/2015/10/02/cookie-richtlinie/>. Version: 2015, Abruf: 21.11.2016
- [Rou13] ROUSE, Margaret: *Denial of Service (DoS)*. <http://www.searchsecurity.de/definition/Denial-of-Service-DoS>. Version: 2013, Abruf: 09.11.2016
- [Rou14] ROUSE, Margaret: *Datenbank-Management-System (DBMS)*. <http://www.searchenterprisesoftware.de/definition/Datenbank-Managementssystem-DBMS>. Version: 2014, Abruf: 18.11.2016
- [Sch] SCHAMBACH, Arno: *IT Outsourcing vs. Inhouse IT oder Cloud Computing vs. On-Premise*. <http://www.softselect.de/wissenspool/it-outsourcing-inhouse-it-cloud-computing-on-premise>, Abruf: 18.11.2016
- [Sch11a] SCHUTT, Timo: *Haftung für Links*. <http://www.itrecht-portal.de/haftung-fuer-links/>. Version: 2011, Abruf: 22.11.2016
- [Sch11b] SCHWENKE, Thomas: *Kann man wegen fehlender AGB abgemahnt werden?* <http://rechtsanwalt-schwenke.de/kann-man-wegen-fehlender-agb-abgemahnt-werden/>. Version: 2011, Abruf: 08.11.2016
- [Ser] *Outsourced Hosting Support vs. In-House Hosting Support*. Server Sitters. <https://serversitters.com/outsourced-hosting-support-vs-in-house-hosting-support.html>, Abruf: 18.11.2016
- [Sie] SIEBERT, Sören: *6 Fragen zu: Impressum für Webseiten*. <https://www.e-recht24.de/artikel/datenschutz/209.html>, Abruf: 21.11.2016

- [Sie16a] SIEBERT, Sören: *AGB für Online Shops: So werden Sie nicht abgemahnt.* <https://www.e-recht24.de/artikel/ecommerce/14.html>. Version: 2016, Abruf: 08.11.2016
- [Sie16b] SIEBERT, Sören: *Neues Gesetz: Fehlende Datenschutzerklärung auf Webseiten kann ab sofort abgemahnt werden.* <https://www.e-recht24.de/artikel/datenschutz/10066-abmahnung-datenschutzerklaerung-auf-webseiten.html>. Version: 2016, Abruf: 08.11.2016
- [sof] *Technische und organisatorische Maßnahmen gemäß Anlage zu §9 BDSG.* Softgarden. <https://www.softgarden.de/technische-und-organisatorische-masnahmen-gemas-anlage-zu-%C2%A79-bdsg/>, Abruf: 18.11.2016
- [Spr13a] SPRINGER, Sebastian: *Node.js - Das umfassende Handbuch.* Galileo Computing, 2013
- [Spr13b] SPRINGER, Sebastian: *Node.js: Das JavaScript-Framework im Überblick.* t3n. <http://t3n.de/magazin/serverseitige-javascript-entwicklung-nodejs-einsatz-231152/>. Version: 2013, Abruf: 30.09.2016
- [Sti14] STICHTENOTH, Olaf: *Single Page Applications: die Lösung für alle Probleme?* <https://blog.secu-ring.de/software/single-page-applications-loesung-fuer-probleme/>. Version: 2014, Abruf: 20.09.2016
- [Sut12] SUTER, Rico: *MongoDB - An introduction and performance analysis.* Hochschule für Technik Rapperswil. <http://wiki.hsr.ch/Datenbanken/files/MongoDB.pdf>. Version: 2012, Abruf: 29.09.2016
- [Tra16] TRAUTRIMS, Marius: *Erfolgreiche Monetarisierung einer Open-Source Webanwendung durch die Integration eines innovativen Geschäftsmodells,* Technische Hochschule Mittelhessen, Diplomarbeit, 2016
- [Tur14] TURNBULL, David: *7 Reasons to Develop Your Next Web App with Meteor.* <https://www.sitepoint.com/7-reasons-develop-next-web-app-meteor/>. Version: 2014, Abruf: 26.09.2016
- [Tut15] TUTORIALS, Bootstrap: *Meteor JS: Was ist das und wie kann ich es nutzen?* <http://bootstrapaholic.de/tutorials/meteor-js-tutorial/>. Version: 2015, Abruf: 26.09.2016
- [Uni] *Autorenwerkzeuge.* Universität Hamburg. <https://www.uni-hamburg.de/elearning/werkzeuge/autorenwerkzeuge.html>, Abruf: 18.11.2016

- [Wag05] WAGNER, Kerstin: *Konzeption und Entwicklung eines Lernmoduls für das Bildungsportal Sachsen gemäß des E-Learning Standards SCORM 1.2*, Hochschule für Technik und Wirtschaft Dresden, Diplomarbeit, 2005
- [Weba] *Cross-Site Request Forgery*. Webmasterpro. <http://www.webmasterpro.de/server/article/sicherheit-cross-site-request-forgery.html>, Abruf: 09.11.2016
- [Webb] *Cross-Site Scripting*. Webmasterpro. <http://www.webmasterpro.de/server/article/sicherheit-cross-site-scripting.html>, Abruf: 09.11.2016
- [Webc] *E-Payment - Online Bezahlssysteme*. Webmasterpro. <http://www.webmasterpro.de/management/article/geld-verdienen-e-paymant-online-bezahlssysteme.html>, Abruf: 18.11.2016
- [Webd] *Session Fixation - Im Namen der ehrlichen Nutzer*. Webmasterpro. <http://www.webmasterpro.de/server/article/sicherheit-session-fixation-im-namen-der-erlichen-nutzern.html>, Abruf: 09.11.2016
- [Webe] *Session Hijacking - Im Namen der ehrlichen Nutzer*. Webmasterpro. <http://www.webmasterpro.de/server/article/sicherheit-session-hijacking-im-namen-der-erlichen-nutzern.html>, Abruf: 09.11.2016
- [Wie] WIERZ, Norman: *Impressum: Pflichtangaben*. [http://www.impressumservice.de/?page\\_id=109](http://www.impressumservice.de/?page_id=109), Abruf: 21.11.2016
- [Wika] *Datenbank*. Wikipedia. <https://de.wikipedia.org/wiki/Datenbank>, Abruf: 18.11.2016
- [Wikb] *Denial of Service*. Wikipedia. [https://de.wikipedia.org/wiki/Denial\\_of\\_Service](https://de.wikipedia.org/wiki/Denial_of_Service), Abruf: 09.11.2016
- [Wikc] *Internetdiensteanbieter*. Wikipedia. <https://de.wikipedia.org/wiki/Internetdiensteanbieter>, Abruf: 22.11.2016
- [Wikd] *Lernplattform*. Wikipedia. [https://de.wikipedia.org/wiki/Lernplattform#Anforderung\\_aus\\_Sicht\\_der\\_Betreiber.2C\\_Entwickler\\_und\\_Administratoren](https://de.wikipedia.org/wiki/Lernplattform#Anforderung_aus_Sicht_der_Betreiber.2C_Entwickler_und_Administratoren), Abruf: 18.11.2016
- [Wike] *MongoDB*. Wikipedia. [https://de.wikipedia.org/wiki/MongoDB#cite\\_note-mongowebseite-2](https://de.wikipedia.org/wiki/MongoDB#cite_note-mongowebseite-2), Abruf: 30.09.2016
- [Wikf] *Node.js*. Wikipedia. <https://de.wikipedia.org/wiki/Node.js>, Abruf: 30.09.2016

- [Wikg] *Persönlichkeitsrecht (Deutschland)*. Wikipedia. [https://de.wikipedia.org/wiki/Pers%C3%B6nlichkeitsrecht\\_\(Deutschland\)](https://de.wikipedia.org/wiki/Pers%C3%B6nlichkeitsrecht_(Deutschland)), Abruf: 22.11.2016
- [Wikh] *SQL-Injection*. Wikipedia. <https://de.wikipedia.org/wiki/SQL-Injection>, Abruf: 09.11.2016
- [Wiki] *Technische und organisatorische Maßnahmen*. Datenschutz-Wiki. [https://www.bfdi.bund.de/bfdi\\_wiki/index.php/Technische\\_und\\_organisatorische\\_Ma%C3%9Fnahmen](https://www.bfdi.bund.de/bfdi_wiki/index.php/Technische_und_organisatorische_Ma%C3%9Fnahmen), Abruf: 18.11.2016
- [Wikj] *Verfahrensverzeichnis*. Wikipedia. <https://de.wikipedia.org/wiki/Verfahrensverzeichnis>, Abruf: 14.11.2016
- [Wikkk] *Verfügbarkeit*. Wikipedia. <https://de.wikipedia.org/wiki/Verf%C3%BCgbarkeit>, Abruf: 21.11.2016
- [Zen] *Datenschutz für Hochschulen*. Zendas - Zentrale Datenschutzstelle der baden-württembergischen Universitäten. [https://www.zendas.de/recht/allgemein/datenschutz\\_hochschule.html](https://www.zendas.de/recht/allgemein/datenschutz_hochschule.html), Abruf: 21.11.2016

# A. Anhang

## A.1. Impressum

### Dienstanbieter

Der Betrieb der Webapp arsnova.cards ist eine kostenlose Dienstleistung der TransMIT-Gesellschaft für Technologietransfer mbH, Projektbereich für mobile Anwendungen.

Kerkraeder Straße 3

D-35394 Gießen

c/o THM – Technische Hochschule Mittelhessen

Fachbereich MNI – Mathematik, Naturwissenschaften und Informatik

Prof. Dr. Klaus Quibeldey-Cirkel

Lehrstuhl für Softwaretechnik

Wiesenstraße 14

D-35390 Gießen

Telefon: +49 641 3090

Telefax: +49 641 3092901

E-Mail: klaus.quibeldey-cirkel@transmit.de

Sitz der Gesellschaft: Gießen

Website: <https://www.transmit.de>

Rechtsform: GmbH

Amtsgericht: Gießen HRB 3036

USt-IdNr.: DE 188 685 037

## **A.2. Datenschutzerklärung**

### **Grundsatz**

Ihre personenbezogenen Daten werden vertraulich behandelt. Sämtliche Handlungen sind an die gesetzlichen Grundlagen des Hessischen Datenschutzgesetzes (HDSG), dem Telemediengesetz (TMG) sowie dieser Datenschutzerklärung gebunden. Dies wird von der TransMIT GmH als Betreiber von arsnova.cards garantiert.

### **Erheben und Nutzen von Daten**

Personenbezogene Daten werden nur dann erhoben, verarbeitet und genutzt, soweit sie für die Begründung, inhaltliche Ausgestaltung und Abwicklung Ihrer Bestellung erforderlich sind (Bestandsdaten). Personenbezogene Daten über die Inanspruchnahme von THMcards (Nutzungsdaten) werden nur erhoben, verarbeitet und genutzt, soweit dies erforderlich ist, um dem Benutzer die Inanspruchnahme des Systems zu ermöglichen oder abzurechnen.

### **Anwendungsdaten**

THMcards sammelt Inhalte und sonstige Informationen, die der Benutzer bereitstellt, wenn er die Plattform nutzt. Dazu gehört auch der Anmeldename sowie Anmeldeservice bei der Registrierung, das Erstellen und Veröffentlichen von Lerninhalten und das Versenden von Nachrichten bzw. die Meldungen an die Administratoren. Dies können Informationen über die von Benutzern bereitgestellten Inhalte sein oder solche, die in ihnen enthalten sind, z.B. das Datum, an dem ein Lerninhalt erstellt wurde. Des Weiteren sammelt THMcards Informationen über das Bereitstellen von Lerninhalten, das Lernverhalten im Rahmen von Lernphasen sowie die Bewertungen.

### **Zugriffsdaten**

Personenbezogene Daten werden nur im technisch notwendige Umfang von THMcards erhoben. Außer Anmeldename und Anmeldeservice des Benutzers sowie Datum und Uhrzeit der Registrierung werden keine personenbezogenen Daten gespeichert.

## Webserver-Logdateien

Es werden automatisch Informationen in Log Files erhoben und gespeichert, die Ihr Browser automatisch an uns übermittelt. Dies sind:

- Browsertyp/ Browserversion
- verwendetes Betriebssystem
- Referrer URL
- Hostname des zugreifenden Rechners
- Uhrzeit der Serveranfrage

Diese Daten sind nicht bestimmten Personen zuordenbar. Eine Zusammenführung dieser Daten mit anderen Datenquellen wird nicht vorgenommen.

## Cookies

Die Internetseiten verwenden teilweise so genannte Cookies. Diese dienen dazu, unser Angebot nutzerfreundlicher, effektiver und sicherer zu gestalten. Cookies sind kleine Textdateien, die auf Ihrem Rechner abgelegt werden und die Ihr Browser speichert. Die meisten der von uns verwendeten Cookies sind so genannte „Session-Cookies“. Sie werden nach Ende Ihres Besuchs automatisch gelöscht. Cookies beeinträchtigt Ihren Rechner nicht und enthalten keine Viren.

Sie können Ihren Browser so einstellen, dass Sie über das Setzen von Cookies informiert werden und diese nur im Einzelfall erlauben, die Annahme von Cookies für bestimmte Fälle oder generell ausschließen sowie das automatische Löschen der Cookies beim Schließen des Browsers aktivieren. Durch die Deaktivierung der Cookies kann es zu Einschränkungen der Funktionalität der Plattform kommen.

## Sicherheit

Es wurden umfangreiche technische und betriebliche Schutzvorkehrungen getroffen, um Ihre Daten vor zufälligen oder vorsätzlichen Manipulationen, Verlust, Zerstörung oder dem Zugriff unberechtigter Personen zu schützen. Die Sicherheitsverfahren werden regelmäßig überprüft und dem technologischen Fortschritt angepasst. Datenübertragungen im Internet (z.B. bei der Kommunikation per E-Mail) können jedoch Sicherheitslücken aufweisen, sodass ein lückenloser Schutz der Daten vor dem Zugriff durch Dritte nicht möglich ist.

## **Datenübermittlungen an Dritte**

Eine Übermittlung Ihrer Daten an Dritte findet nicht statt, es sei denn, die Betreiber von THMcards sind gesetzlich dazu verpflichtet. Soweit externe Dienstleister mit Ihren personenbezogenen Daten in Kontakt kommen, wird durch rechtliche, technische und organisatorische Maßnahmen sowie durch regelmäßige Kontrollen sichergestellt, dass diese die Vorschriften der Datenschutzgesetze einhalten.

## **Links zu Webseiten anderer Anbieter**

THMcards kann Links zu Webseiten anderer Anbieter enthalten, auf die sich diese Datenschutzerklärung nicht bezieht. Soweit mit der Nutzung der Internetseiten anderer Anbieter die Erhebung, Verarbeitung oder Nutzung personenbezogener Daten verbunden ist, beachten Sie bitte die Datenschutzhinweise der jeweiligen Anbieter.

## **Einbindung von Diensten und Inhalten Dritter**

Es kann vorkommen, dass angebotene Lerninhalte von THMcards Inhalte Dritter, wie zum Beispiel Videos von YouTube und Vimeo oder Grafiken von anderen Webseiten, eingebunden werden. Dies setzt voraus, dass die Anbieter dieser Inhalte die IP-Adresse der Benutzer/innen wahrnehmen. Ohne die IP-Adresse können die Inhalte nicht an den Browser gesendet werden.

## **Zahlungsinformationen**

Wird THMcards für Einkäufe oder finanzielle Transaktionen genutzt (beispielsweise wenn ein Kartensatz gekauft wird oder ein Abonnement abgeschlossen wird), werden von THMcards keine Zahlungsinformationen gespeichert. Alle Transaktionen werden über den Dienst Braintree abgewickelt. Dort werden alle Konto- und Authentifizierungsinformationen und Angaben zur Abrechnung gesammelt.

## **Übrige Datenerfassung**

Weitergehende personenbezogene Daten werden nur erfasst, wenn Sie diese Angaben freiwillig, etwa im Rahmen einer Dozenten-Anfrage, angeben. Soweit Sie uns personenbezogene Daten zur Verfügung gestellt haben, werden diese nur zur Beantwortung Ihrer Anfragen, zur Abwicklung mit Ihnen geschlossener Verträge, zur Auswertung der Lernphasen und für die technische Administration.

## **Widerspruchsrecht und Auskunftsrecht**

Sie haben jederzeit das Recht auf unentgeltliche Auskunft über Ihre gespeicherten personenbezogenen Daten, deren Herkunft und Empfänger, den Zweck der Datenverarbeitung sowie ein Recht auf Berichtigung, Sperrung oder Löschung dieser Daten.

Bei Unklarheiten oder Anregungen wenden Sie sich an folgende E-Mail-Adresse oder an die im Impressum angegebenen Kontaktdaten:

`klaus.quibeldey-cirkel@transmit.de`

Eine Übersicht über die in THMcards gespeicherten Daten finden Sie im Verfahrensverzeichnis unter:

<https://git.thm.de/arsnova/flashcards/wikis/verfahrensverzeichnis>

## **Änderungen der Datenschutzerklärung**

Wir behalten uns das Recht vor, diese Datenschutzerklärung jederzeit unter Beachtung der geltenden Datenschutzvorschriften zu ändern. Derzeitiger Stand ist der 17.10.2016.

## A.3. Allgemeine Geschäftsbedingungen

### Geltung der Allgemeinen Geschäftsbedingungen

- Die Nutzung von THMcards wird durch eine rechtliche Vereinbarung zwischen dem Benutzer und THMcards geregelt. Die Webapp THMcards ist eine kostenlose Dienstleistung der TransMIT-Gesellschaft für Technologietransfer mbH, Projektbereich für mobile Anwendungen mit der Anschrift: Kerkrader Straße 3, 35394 Gießen, Deutschland.
- Alle Studierende der Technischen Hochschule Mittelhessen THMcards dürfen THMcards für ihre Zwecke verwenden, solange dabei nicht gegen deutsches oder EU-Recht verstoßen wird.
- THMcards ist ein kostenloser Webservice (Software as a Service). Die Software ist Open Source und steht unter der GNU General Public License Version 3.
- Neben dem eigentlich Lernkartendienst bietet THMcards optional die Möglichkeit, kostenpflichtige Lerninhalte von anderen Benutzern der Plattform zu nutzen. Die folgenden Geschäftsbedingungen sollen das Rechtsverhältnis zwischen dem Benutzer, THMcards und den Autoren, die kostenpflichtige Lerninhalte anbieten, regeln.
- Um THMcards nutzen zu können, muss die Vertragsbedingung angenommen werden. Sollte der Benutzer die Vertragsbedingung nicht annehmen, wird dieser Dienst nicht freigeschaltet.
- Der Vertrag wird vom Benutzer angenommen, wenn dieser bei der ersten Anmeldung in das System die allgemeinen Geschäftsbedingungen mit der Schaltfläche „Akzeptieren“ einwilligt.
- Mit der Annahme der Vertragsbindung wird akzeptiert, dass THMcards bei wichtigen Änderungen, Ankündigungen oder beim Starten einer Lernphase E-Mails oder im System integrierte Nachrichten an den Benutzer sendet.
- Wenn bei THMcards Kartensätze gekauft werden (optional), kommt zwischen dem Benutzer und THMcards ein rechtsgültiger Vertrag zustande. Nach der Bezahlung können die ausgewählten Kartensätze in der Pool-Ansicht vom Benutzer eingesehen und gelernt werden.

## **Regeln bei der Nutzung von THMcards**

- Bei der Benutzung von THMcards versichert der Benutzer, dass
  - alle von ihm angegebenen Informationen im System wahrheitsgetreu und richtig sind.
  - gegen keine geltenden Gesetze verstoßen werden.
  - er für alles was unter dem Nutzernamen geschieht, verantwortlich ist.
- Bei der Benutzung sind keine rassistischen, hetzerischen oder beleidigenden Äußerungen in jeglicher hier möglichen Form erlaubt.
- THMcards behält sich das Recht vor, Benutzer, die sich nicht an die Regeln halten, zu sperren.

## **Verlässlichkeit und Verfügbarkeit von THMcards**

- Eine ununterbrochene Verfügbarkeit von THMcards ist technisch nicht möglich. Wartungs- und Updatemaßnahmen des Dienstes sowie unvorhersehbare Störungen, wie z.B. Ausfälle beim Internet-, Server- oder Stromprovider, können zu vorübergehenden Behinderungen der Erreichbarkeit des Dienstes führen. THMcards ist bemüht, den Online-Dienst zuverlässig und störungsfrei wie möglich zu betreiben soweit dies technisch möglich ist.
- Jeder Benutzer ist bei einem Ausfall des Dienstes für seine eigenen Inhalte selbst verantwortlich. Hierfür sollte regelmäßig die Export-Funktion des jeweiligen Kartensatzes genutzt werden, um eigene Inhalte sicherzustellen, wenn THMcards ausfallen sollte.

## **Passwörter und Sicherheit des Benutzerkontos**

- Jeder Benutzer ist alleine für die Geheimhaltung seines Passwortes des Benutzerkontos verantwortlich.
- Sollte ein Benutzer darauf aufmerksam werden, dass das Benutzerkonto unbefugt durch einen Dritten benutzt wird, ist er dazu verpflichtet, THMcards umgehend zu benachrichtigen.

## Datenschutz

- THMcards achtet auf die Privatsphäre der Benutzer. Sämtliche datenschutzrechtliche Bestimmungen werden eingehalten. Persönliche Daten wie E-Mail-Adresse oder Anmeldedaten werden nicht an Dritte weitergegeben, außer in Fällen einer gesetzlichen Verpflichtung.
- Gespeicherte Lernstatistiken und Batches eines Benutzers bleiben ebenfalls geschützt, solange der Benutzer die Möglichkeit zur Einsicht nicht explizit anderen Benutzern von THMcards erlaubt hat (Öffentlichmachung des Benutzerprofils). Dies geschieht implizit, sollte ein Benutzer die Rolle eines „Pro-Users“ oder „Lecturer-Users“ erlangen.
- Bei der Löschung eines Benutzerkontos von THMcards werden alle persönlichen Daten und privaten Kartensätze gelöscht. Das Benutzerkonto kann unter den Einstellungen des Profils gelöscht werden.

## Inhalte auf THMcards

- Jeder Benutzer von THMcards ist für alle Inhalte, die er eingibt und speichert (z.B. Texte, Bilder), selbst verantwortlich.
- Jeder Benutzer bestätigt gegenüber THMcards, dass er über sämtliche Rechte, Vollmachten und Befugnisse verfügt, um seine Inhalte zu veröffentlichen.
- Jeder Benutzer erkennt an, dass er allein für alle Inhalte, die er während der Nutzung von THMcards erstellt, überträgt oder darstellt und die Auswirkungen seiner Handlungen (einschließlich vorhandener Verluste oder Schäden, die THMcards oder Dritten hierbei entstehen können) verantwortlich ist.
- Sollte ein Benutzer Inhalte erstellen, die gegen geltendes Recht verstoßen, behält sich THMcards das Recht vor, diese Inhalte zu löschen.
- Lerninhalte, die von Benutzern selbstständig erarbeitetet und öffentlich zur Verfügung gestellt wurden, dürfen gemäß der Creative Commons Licence weiterverwendet werden.
- Bei nicht veröffentlichten Lerninhalten, die nicht das Urheberrecht Dritter verletzt, liegt das Urheberrecht beim Benutzer.

- Jedem Benutzer ist bekannt, dass veröffentlichte Lerninhalte von anderen Benutzern von THMcards weiterverwendet werden können. Da die veröffentlichten Inhalte nach Löschung des Benutzers weiterhin bestehen bleiben, erklärt sich jeder Benutzer damit einverstanden, dass die veröffentlichten Inhalte weiterhin von anderen Benutzern verwendet werden dürfen und anderen zur Verfügung gestellt werden können.

## **THMcards FREE und THMcards PRO**

- THMcards ist in der Free-Version kostenlos nutzbar, jedoch können nicht alle Lerninhalte der Plattform eingesehen werden.
- Die Pro-Version von THMcards ist erhältlich, sobald der Benutzer den zum Zeitpunkt des jeweiligen Vertragsabschlusses gültigen Preis für eine Leistungsbeschreibung zahlt.
- Die Gebühren für die Pro-Version und der kostenpflichtigen Lerninhalte ergeben sich aus dem zum Zeitpunkt des jeweiligen Vertragsabschlusses gültigen Preis- und Leistungsbeschreibung bzw. dem Preis der käuflichen Lerninhalte und enthalten die gesetzliche Mehrwertsteuer.
- Die Laufzeit der Pro-Version wird bis zur Kündigung automatisch jeden Monat erneuert.
- Entscheidet sich ein Benutzer von THMcards nach Ablauf der Pro-Version für die Free-Version, erklärt sich dieser damit einverstanden, dass er THMcards erst nutzen kann, wenn er die Leistungsbeschreibung der Free-Version erfüllt.
- THMcards bedient sich Diensten von Dritten, um Einkäufe oder finanzielle Transaktionen abzuwickeln. Diese werden im Bestellprozess deutlich bezeichnet.
- Die Vertragsbedingungen bleiben wirksam, bis sie von dem Benutzer oder von THMcards gekündigt werden.
- Die Kündigung der Free-Version von THMcards erfolgt durch die Löschung des Accounts. Dies ist unter den Einstellungen des Profils möglich.

- Die Pro-Version kann vom Benutzer jederzeit gekündigt werden und hat bis zum Ablauf des monatlichen Abrechnungszeitraums weiterhin die Möglichkeit, THMcards zu nutzen.
- THMcards gewährt keine Rückerstattungen oder Gutschriften für nicht vollständig genutzte Monate oder nicht genutzte Lerninhalte.
- THMcards kann den Vertrag mit dem Benutzer jederzeit kündigen, falls:
  - Der Benutzer gegen Bestimmungen der Vertragsbedingungen verstößt.
  - THMcards vom Anbieter nicht mehr betrieben wird.

## Widerrufsrecht

- Der Benutzer hat das Recht, seine Vertragserklärung innerhalb von 14 Tagen ohne Angaben von Gründen zu widerrufen. Die Frist beginnt ab dem Tag des Vertragsabschlusses.
- Um den Widerruf auszuüben, muss eine eindeutige Erklärung über den Entschluss, den Vertrag zu widerrufen, auf einem dauerhaften Datenträger (z.B. ein Brief, Telefax, E-Mail) gesendet werden. Die Erklärung ist an folgende Adresse zu richten: (Technische Hochschule Mittelhessen, Wiesenstraße 14, 35390 Gießen, Deutschland, E-Mail: `klaus.quibeldey-cirkel@transmit.de`)
- Zur Wahrung der Widerrufsfrist genügt die rechtzeitige Absendung der Mitteilung über die Ausübung des Widerrufsrechts vor Ablauf der Widerrufsfrist.
- Bei einem wirksamen Widerruf sind die beiderseits empfangenen Leistungen zurück zu gewähren, soweit es möglich ist (Löschung des Benutzers).
- Die Erstattung empfangener Leistungen muss innerhalb von 14 Tagen erfolgen. Die Frist beginnt beim Benutzer mit der Absendung der Widerrufserklärung, für uns mit dem Empfang. Für die Rückerstattung von Zahlungen verwenden wir dasselbe Zahlungsmittel, das bei der Ursprünglichen Transaktion eingesetzt wurde, es sei denn, es wurde ausdrücklich etwas anderes vereinbart. Es werden in keinem Fall bei der Rückzahlung weitere Entgelte berechnet.

## Gewährleistung

- THMcards übernimmt keine Verantwortung für die von Benutzern bereitgestellten Lerninhalte und macht sich diese auch nicht zu eigen.
- THMcards bietet lediglich den technischen Dienst zur Erstellung, Speicherung und des Austausches der erstellten Lerninhalte.
- Eine ständige Überprüfung aller eingestellter Inhalte auf mögliche Rechtsverletzungen ist nicht realisierbar und findet daher grundsätzlich nicht statt.
- THMcards reagiert nach berechtigtem Hinweis auf Rechtsverstöße und behält sich vor, straf- oder haftungsrechtliche Inhalte unverzüglich zu löschen.
- Die Korrektheit eingestellter Lerninhalte wird von THMcards nicht gewährleistet. Alle Lerninhalte werden von den Benutzern des Systems erstellt und/oder verändert. Eine Überprüfung auf Korrektheit ist seitens THMcards hierbei nicht möglich.

## Änderungen der Vertragsbedingungen

- THMcards behält sich das Recht vor, Änderungen an diesen allgemeinen Geschäftsbedingungen vorzunehmen.
- Die Benutzer von THMcards werden über E-Mail oder durch eine Nachricht im System über Änderungen der allgemeinen Geschäftsbedingungen informiert. Die Änderungen werden wirksam, wenn der Benutzer nicht innerhalb 14 Tage nach Zugang der Information den Änderungen widerspricht. In der E-Mail oder Nachricht wird gesondert auf Widerspruchsmöglichkeit und deren Frist hingewiesen.
- Im Übrigen gelten die gesetzlichen Gewährleistungsvorschriften.
- THMcards behält sich das Recht vor, bei Widerspruch der geänderten Geschäftsbedingungen das Nutzungsverhältnis zu kündigen und zu beenden. Der Benutzer hat das Recht, das Vertragsverhältnis innerhalb von 4 Wochen nach Änderung der allgemeinen Geschäftsbedingungen bis zum Ende des nächsten Monats zu kündigen.

## **Sonstige Bedingungen**

- Für Lern- und Prüfungserfolg gibt es seitens THMcards keine Gewährleistung.
- Verstößt ein Benutzer gegen diese allgemeinen Geschäftsbedingungen, behält sich THMcards das Recht vor, von diesem Benutzer eingestellte Inhalte zu löschen. Bei mehrmaligen Verstoß des Benutzers wird das entsprechende Benutzerkonto gesperrt.

Stand: 17.10.2016

## A.4. Verfahrensverzeichnis

"Nach dem ab 1.6.1999 geltenden §6 Abs.1 HDSG hat jede der in §3 Abs.1 HDSG genannten öffentlichen Stellen, die im Rahmen eines automatisierten Verfahrens personenbezogene Daten speichert oder gemäß §4 HDSG im Auftrag speichern lässt, sicherzustellen, dass der für den Einsatz des Verfahrens Zuständige ein für den behördlichen Datenschutzbeauftragten bestimmtes Verzeichnis nach dem angefügten ersten Muster erstellt." Vgl. *Hessischer Datenschutzbeauftragter*.

Sofern nicht anders angegeben, beziehen sich alle Paragraphen auf das Hessische Datenschutzgesetz (abgekürzt mit HDSG).

### Name und Anschrift der datenverarbeitenden Stelle

Technische Hochschule Mittelhessen  
Wiesenstraße 14  
35390 Gießen  
Germany

### Zweckbestimmung und Rechtsgrundlage der Datenverarbeitung

Die erhobenen Daten dienen der Identifizierung eines Benutzers im System. Sie dienen einer interaktiven Kommunikation zwischen den Benutzern in THMcards, um einen Wissenszuwachs speziell bei den Lernenden zu erzielen (vgl. Definition: *Lernplattform*). Die Rechtsgrundlage ist das zweifellose Einverständnis der Betroffenen nach §7 Abs.1 (3)).

### Art der gespeicherten Daten

Die Art der gespeicherten Daten kann unter <https://git.thm.de/arsnova/flashcards/wikis/verfahrensverzeichnis> eingesehen werden.

## **Kreis der Betroffenen**

Alle Portalmitglieder. I.d.R. sind dies

- Studierende
- Dozenten (Professoren, Gast-Dozenten, Lehrbeauftragte, etc.)
- Mitarbeiter/innen
- Ehemalige Studierende

## **Art regelmäßig übermittelter Daten, deren Empfänger sowie Art und Herkunft regelmäßig empfangener Daten**

THMcards benutzt den Payment Service Provider Braintree, um alle Einkäufe oder finanzielle Transaktionen (beispielsweise wenn ein Kartensatz gekauft wird oder ein Abonnement abgeschlossen wird) abzuwickeln. Dort werden alle Konto- und Authentifizierungsinformationen sowie Angaben zur Abrechnung gesammelt.

Es werden keine Daten von Dritten regelmäßig empfangen.

## **Zugriffsberechtigten Personen oder Personengruppen**

Auf personenbezogene Daten haben nur die Personen Zugriff, denen der Betroffene ein Leserecht eingeräumt hat oder Dozenten, sofern eine Lernphase gestartet wurde. Ansonsten können die Administratoren die personenbezogenen Daten einsehen. Sie unterliegen dem Datengeheimnis nach § 9.

## **Technische und organisatorische Maßnahmen nach §10**

### **Zutrittskontrolle**

Die Serverräume liegen unter der Verantwortung der IT-Services (ITS) der TH Mittelhessen. Sie sind durch elektronische Schlösser gesichert, wobei nur ein spezifizierter Personenkreis eine Zugangsberechtigung besitzt.

### **Benutzerkontrolle**

Die Benutzer authentifizieren sich über den zentralen Authentifizierungsserver CAS, wenn sie sich in THMcards einloggen möchten.

### **Zugriffskontrolle**

THMcards stellt durch folgende Maßnahmen sicher, dass keine fremden personenbezogenen Daten einsehbar sind:

- Rollenkonzept
- Definieren von Sichtbarkeitsbereichen

### **Datenverarbeitungskontrolle**

Die Kommunikation zwischen Client und Server wird durch das Kommunikationsprotokoll HTTPS (Hypertext Transfer Protocol Secure) gesichert. Zugriff auf Betriebssystemebene wird durch das Public-Key-Verschlüsselungs-Verfahren RSA (Rivest, Shamir und Adleman) geschützt.

### **Verantwortlichkeitskontrolle**

Wird in den Datenbank-Logdateien erfasst und gesichert.

### **Auftragskontrolle**

In THMcards werden keine Daten im Auftrag eines Auftraggebers verarbeitet.

### **Dokumentationskontrolle**

THMcards wird in einem Wiki dokumentiert. Darüber hinaus steht eine Ausführliche Dokumentation durch die Abschlussarbeiten *Implementierung von Spaced Repetition Algorithmen zur effektiven Abfrage von Lernkaten innerhalb der eLearning Plattform THMcards* von Jan Christopher Kammer und *Implementierung von Spielmechaniken zur Steigerung der Lernmotivation von Studierenden am Beispiel der Lernkarten Plattform THMcards* von Daniel Knapp zur Verfügung. [Kam14] [Kna13]

### **Organisationskontrolle**

Alle THMcards-Entwickler inkl. den Administratoren werden zu Beginn ihrer Tätigkeit zu datenschutzrechtlichen Fragen geschult.

## **Fristen für die Löschung nach §19 Abs. 3**

Eine Löschung von Daten findet weder statt, noch ist sie geplant.

Der Benutzer kann alle von ihm erstellten Daten ohne Zeitverzögerung löschen. Einzige Ausnahme: Daten, die als "Free", "Edu" oder "Pro" Lerninhalt veröffentlicht werden können nicht mehr gelöscht werden.

### **Beabsichtigte Datenübermittlung an Drittstaaten nach §17 Abs. 2**

Eine Datenübermittlung an Drittstaaten findet weder statt, noch ist sie geplant.

### **Begründetes Ergebnis der Untersuchung nach §7 Abs. 6 Satz 3**

Da dieses Verfahren bereits bei Erstellung dieses Verfahrensverzeichnisses lief, ist eine Vorabkontrolle nicht möglich. Aufgrund der in diesem Verfahrensverzeichnis beschriebenen Maßnahmen kann davon ausgegangen werden, dass der Schutz der in § 1 Abs. 1 Nr. 1 beschriebenen Rechte gegeben ist.

# Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Fulda, den 22. November 2016

---

MATTHIAS ZIMNY

