Web Engineering, 2. Semester  
Projektmanagment, 2. Semester

Duale Hochschule Baden-Württemberg  
TINF13B2

Betreuer:

Jörn Eisenbiegler  
Simone Freudenmann

Studenten:

David Ehlen (5460297)  
Julien Hadley Jack (4739854)  
Sebastian Dernbach (9586963)

**Flashcard Website**

Contents

[Funktionen 2](#_Toc392683416)

[Testen 5](#_Toc392683417)

[Installation 5](#_Toc392683418)

[Struktur der Seiten/Layout 6](#_Toc392683419)

[Umsetzung 7](#_Toc392683420)

[• HTML: 7](#_Toc392683421)

[• Struktur der Website 7](#_Toc392683422)

[• Login&Registrierungsformular 7](#_Toc392683423)

[• CSS: 7](#_Toc392683424)

[• Darstellung der Website 7](#_Toc392683425)

[• Javascript/JQuery 7](#_Toc392683426)

[• Karteikarten Animation 7](#_Toc392683427)

[• Client Logik 7](#_Toc392683428)

[• Ajax Request zu den JSP’s 7](#_Toc392683429)

[• JSP 7](#_Toc392683430)

[• Weiterleitungen 7](#_Toc392683431)

[• Benutzersessions 7](#_Toc392683432)

[• Aufruf von Java Beans 7](#_Toc392683433)

[• Java Beans 7](#_Toc392683434)

[• Datenbankadapter 7](#_Toc392683435)

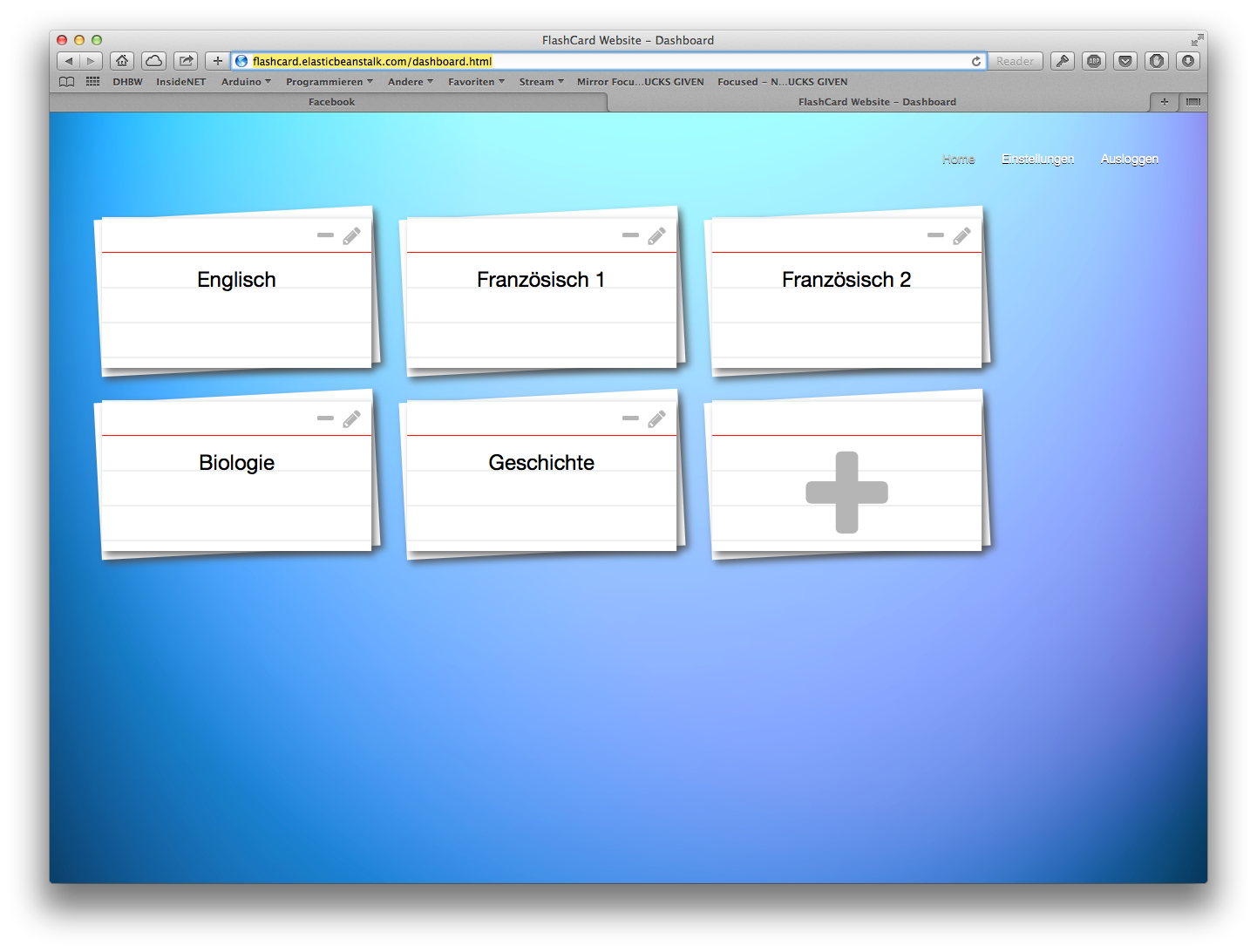
[• Benutzer 7](#_Toc392683436)

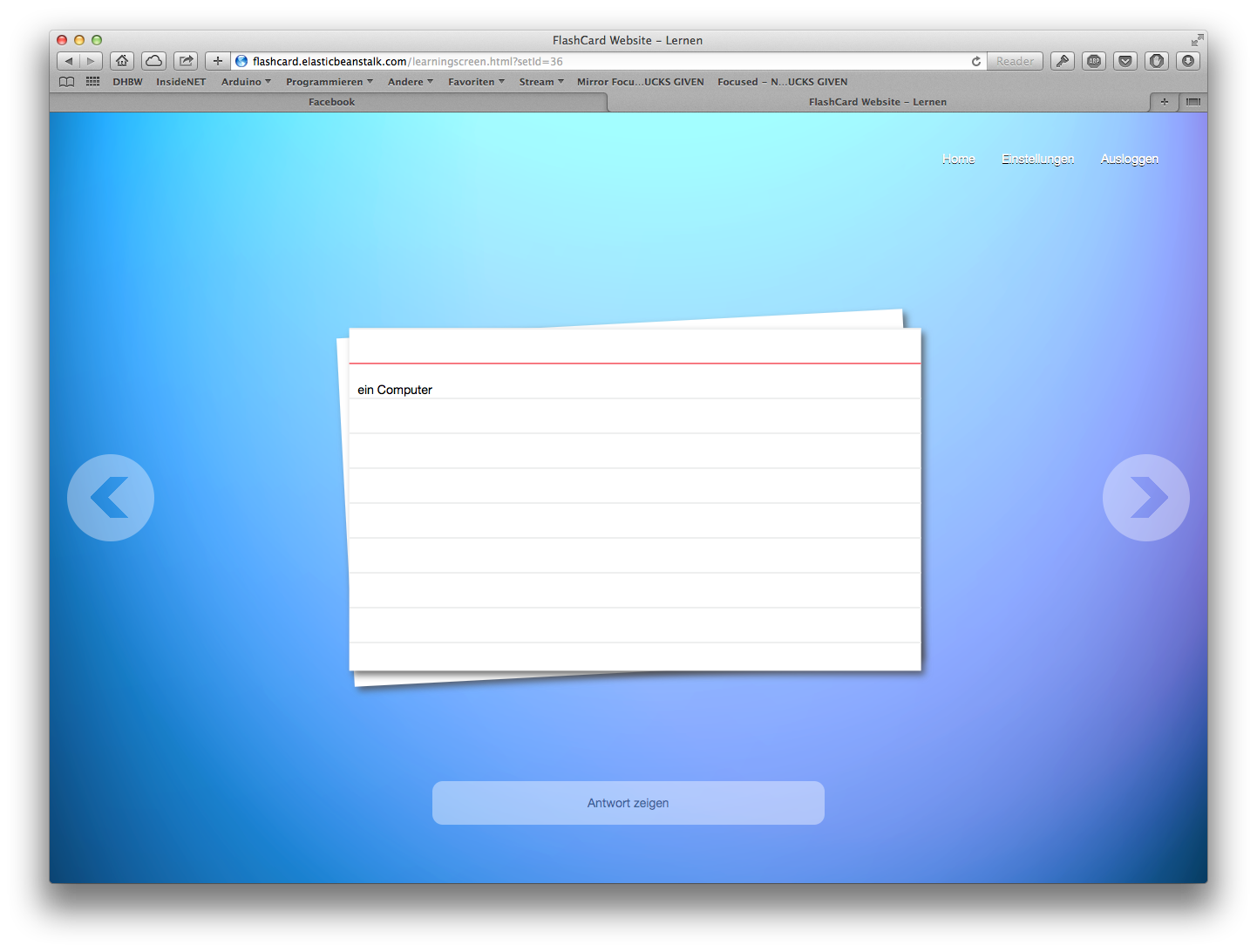
# Funktionen

Studyflash realisiert ein Online-Lernsystem, um Karteikarten elektronisch zu erstellen, verwalten und zu lernen.

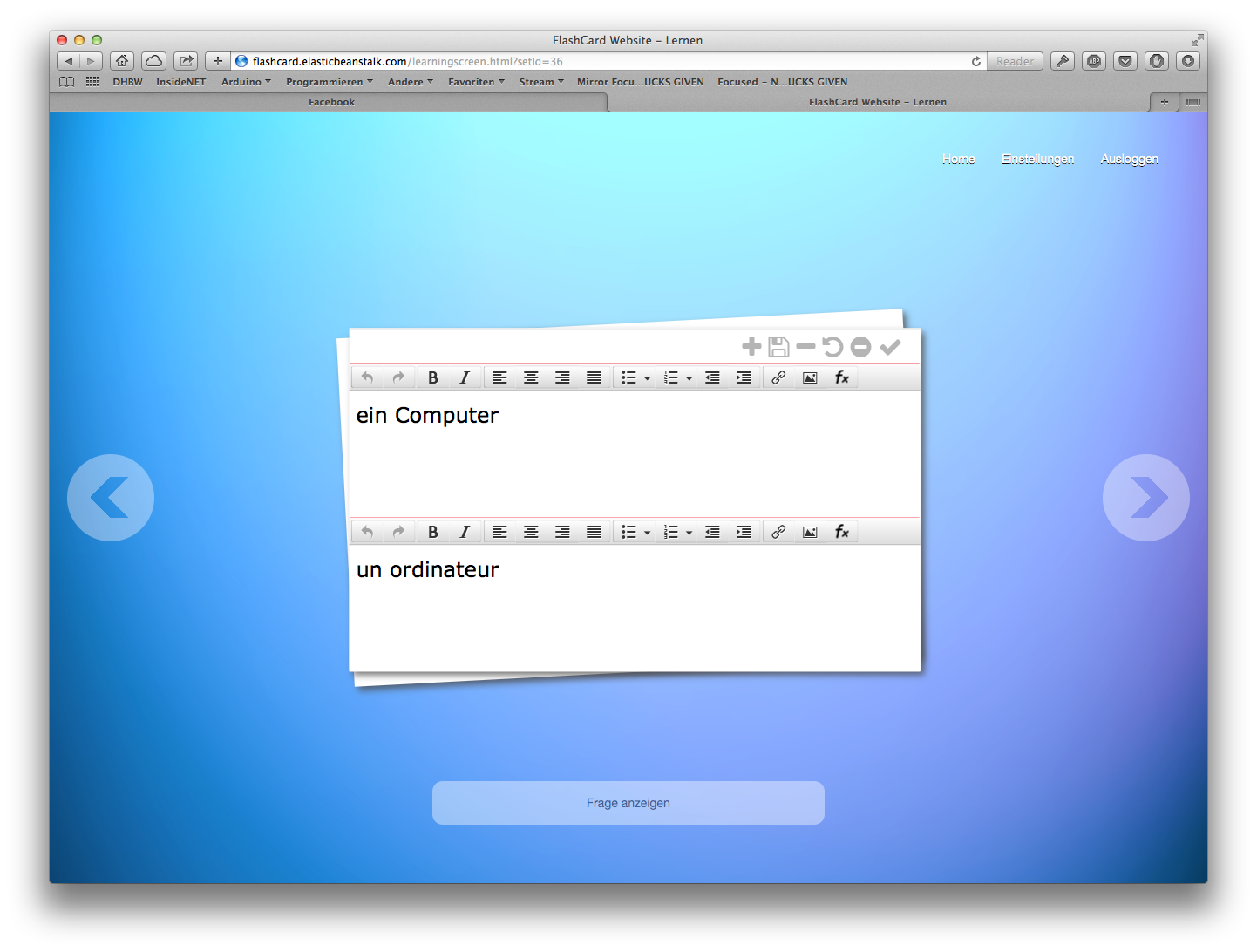


Im Folgenden wird eine einfache Anleitung gegeben, wie das System genutzt werden kann. Zunächst muss ein Benutzer registriert werden. Hierbei reicht es eine E-Mail und ein Passwort anzugeben. Falls man Probleme hat, kann man die ausgiebigen Hilfeseiten zu Rate fragen. Jene bieten dem Benutzer eine Übersicht des Angebotes und helfen dem User bei der erstmaligen Anmeldung. Sollte Interesse über das Entwicklungsteam bestehen, kann man Genaueres im Impressum nachgelesen werden. Beide Links können im Menü gefunden werden.

Nach erfolgreichen Anmeldung wird man auf die Übersichtsseite weitergeleitet, wo sie ihre Sets sehen, löschen(-) und hinzufügen(+), sowie deren Namen editieren können (Stift-Symbol).



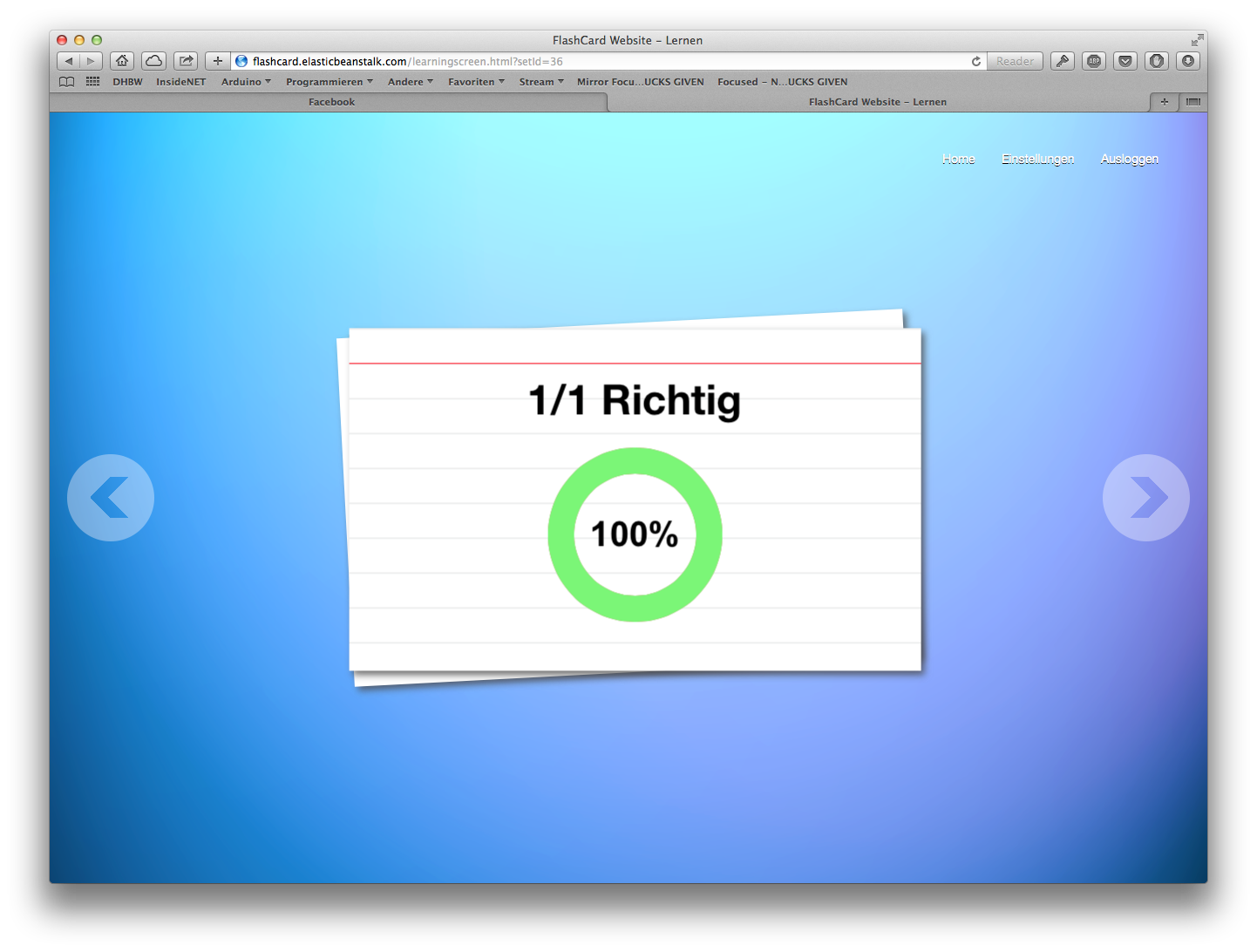
Klicken sie auf eines der Sets landen sie im Lernmodus, in dem sie durch die Karteikarten browsen können. Wollen sie die Antwort einer Karte sehen drücken sie die Taste "a" oder den Button "Antwort anzeigen". Die von ihnen angegeben Antwort erscheint und sie können in der rechten oberen Ecke der Karteikarte angeben, ob sie die Antwort wussten oder nicht.



Außerdem bietet sich noch weitere Möglichkeiten, wie zum Beispiel eine neue Karteikarte zum Set hinzuzufügen (+), die bestehende in einem voll funktionalen Editor zu bearbeiten (Stift Symbol), die Karteikarte zu löschen (-) oder eventuelle Änderungen rückgängig zu machen (Pfeil Symbol), sowie ihre Änderungen zu speichern (Disketten Symbol).

Das komplette Lernsystem ist auch über die Tastatur zu steuern, alle Tastenkürzel finden sie hier aufgelistet:

* a: Antwort anzeigen
* r: Richtig markieren
* f: Falsch markieren
* Linke Pfeiltaste: Nach links gehen
* Rechte Pfeiltaste: Nach rechts gehen
* e: Editieren



Sind sie am Ende eines Sets angekommen, gibt das System ihnen aufgrund ihrer angegebenen falschen/richtigen Antworten eine Statistik aus.

# Testen

Das Endprodukt kann auf http://flashcard.elasticbeanstalk.com erreicht und getestet werden. Für diesen Zweck haben wir einen Test-Benutzer mit dem Benutzernamen „[test@test.de](mailto:test@test.de)“ und dem Password „test“ erstellt.

# Installation

Das Projekt wird als gezippter Workspace geschickt. Dieser kann in Eclipse importiert werden und anschließen lokal auf einem Tomcat deployed werden. Unser Projekt greift dabei auf eine MySQL Datenbank zu, um das Login-System, die Speicherung von Karteikarten etc zu realisieren. Damit das Projekt bei der Abgabe auch von einem lokalen Tomcat ausgeführt werden kann, haben wir die Verbindung zur Datenbank dahingehend angepasst. Ist jedoch gewünscht die einzelnen Funktionen wie das Hinzufügen eines Sets beispielsweise in der Datenbank zu überprüfen oder unsere Datenbankstruktur näher unter die Lupe zu nehmen, ist dies durch die beiliegende SQL Datei möglich. Jene Datei kann in eine lokale MySQL Installation importiert werden.

# Struktur der Seiten/Layout

Das grundsätzliche System beruht auf einer HTML-Struktur in Verbindung mit JavaScript, die via Ajax Request an eine JSP-Datei Informationen sendet, bzw. diese von der JSP erhält. Die JSP leitet die Daten jeweils an eine JavaBean weiter, welche mit der Datenbank zum lesen bzw. schreiben der Daten verbunden ist.

So bietet das Dashboard eine Übersicht über alle Karteikarten Sets und die Möglichkeit neue hinzuzufügen, vorhandene zu editieren oder zu löschen. Bei jeder dieser Aktionen wird das oben genannte Szenario genutzt. Lediglich bei der Aktion ein neues Set hinzuzufügen wird der HTML Code eines neuen Sets eingefügt und zur Eintragung der Daten auf die Editierfunktion zurückgegriffen. Diese unterscheidet zwischen bereits vorhandenen Sets und neuen Sets und führt die jeweilige Ajax Anfrage aus.

Durch den Klick auf ein Karteikarten Set gelangt man in den Lernmodus, auch hier werden die Karten dynamisch durch Ajax geladen. Der Bildschirm bietet ebenfalls wieder die Möglichkeiten Karteikarten hinzuzufügen, zu bearbeiten oder zu löschen. Die Verarbeitung dieser Aktionen funktioniert genauso wie beim Dashboard. Zusätzlich bietet die Seite die Möglichkeit eine Karte als richtig oder falsch beantwortet zu makieren. Hierbei wird im Hintergrund via Javascript die Anzahl der korrekten Antworten hoch oder herunter gezählt. Der Lernmodus lässt eine Navigation durch Pfeiltasten oder durch die Tastatur zu, hierbei wird lediglich die Position der Karteikarten verändert und via jQuery animiert.

# Umsetzung

HTML (Struktur der Website) und :

* Loginfomular
* Registrierungsformular

CSS (Darstellung der Website):

* Hintergrund
* Karteikarte

Javascript / jQuery:

* Karteikarten-Animation
* Validierung des Login-Formulars
* Validierung des Registrierung-Formulars
* Ajax Request zu den JSPs (z.B.: Erstellung eines Sets)

Java (Beans):

* Benutzer registrieren, einlogen, löschen,…
* Karteikarten(sets) hinzufügen, modifizieren, löschen
* Datenbankadapter

JSP:

* Weiterleitungen
* •Benutzersessions
* •Aufruf von Java Beans