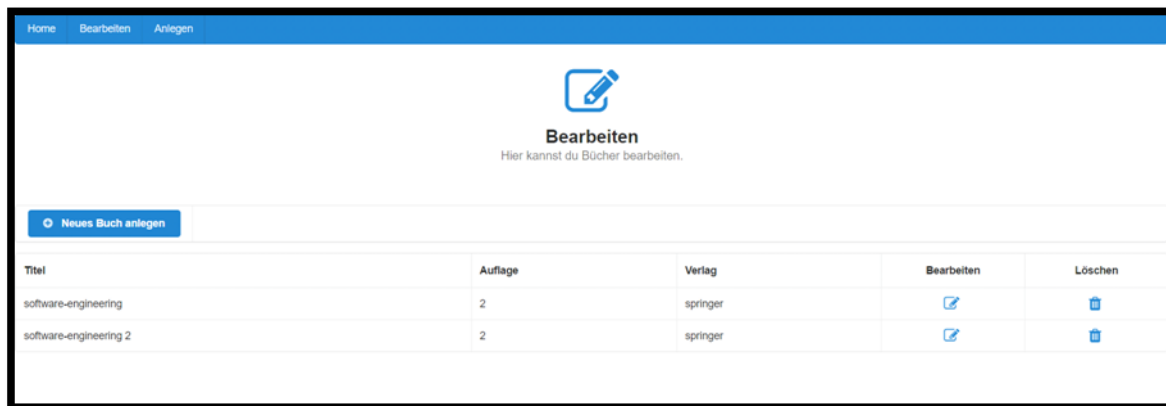


Model-View-Controller anhand Ember.JS

Wir haben unser Frontend mit dem Framework Ember.JS gebaut. Ember.JS verlangt das Muster MVC explizit. Anhand Ausschnitte aus unserem Programmcode und Screenshots unserer Webseite soll das MVC-Prinzip im Folgenden erläutert werden. Dazu wird unser Anwendungsbeispiel "Buch anlegen" verwendet. Bei diesem Anwendungsbeispiel soll dem Admin es möglich sein ein neues Buch in die Datenbank aufzunehmen. Zunächst wird die View betrachtet. Diese ist die Präsentationsschicht und ist für die Darstellung der benötigten Daten aus dem Modell und die Entgegennahme von Benutzerinteraktionen zuständig. Auf der folgenden Abbildung ist die View unserer Bearbeiten-Seite zu sehen mit den Büchern(Daten aus dem Modell), welche hier dem Admin angezeigt werden.



Im nächsten Schritt springt unser Admin auf die Seite "Anlegen". Hier kann der Admin alle benötigten Attribute für das Anlegen eines neuen Buches eingeben. Im Anschluss klickt er auf den Button "Anlegen".

Diese Benutzerinteraktion wird von der View entgegengenommen und an den Controller "anlegen.js" übergeben. Der Controller sorgt dafür, dass Benutzeraktionen wirksam werden, z. B. durch Änderung der Präsentation (z. B. Verschieben des Fensters) oder durch Weiterleiten an das Modell (z. B. Übernahme von Eingabedaten oder Auslösen von Verarbeitungen). In unserem Fall übergibt der Controller die eingegeben Attribute an das Modell weiter. Das Modell enthält die darzustellenden Daten. Hier werden also die Daten jetzt manipuliert. Im zweiten Schritt führt der Controller eine Methode aus, um auf die "Bearbeiten" Seite umzuleiten. Es findet also eine Änderung der View statt. Die View präsentiert die Daten aus dem Modell mit dem vom Admin neu angelegten Buch.

An diesem Beispiel sollte die strikte Trennung der Aufgabenverteilung der drei Komponenten des MVC-Prinzips deutlich dargestellt sein.