



Web-basierte Anwendungen

Praktikumsaufgabe 4: Ein „geführtes“ Rails-Projekt

Studiengänge AI (4140) & WI (4120)



Hintergrund

- Motto: Erst mal machen, dann verstehen lernen
 - Obwohl uns noch viele theoretische Grundlagen fehlen, sollen Sie bereits jetzt ein kleines, aber vollständiges Rails-Projekt implementieren.
 - Sie folgen dazu einer bewährten detaillierten Anleitung, die auch viele Erklärungen enthält.
 - Das dabei entstehende kleine Rails-Projekt wird anschließend nicht fortgesetzt, es dient „nur“ zum Üben.
 - Mit Ihren Erfahrungen aus diesem „Wegwerf-Projekt“ werden Sie die Aufgabenstellung für das Projekt im Januar leichter verstehen und auch die Arbeitsaufwände für die Teilaufgaben besser einschätzen können.
 - Ferner wird Sie diese Erfahrung aufnahmefähiger für die späteren Vorlesungsteile machen, welche die hier entdeckten Vorgehensweisen in einen größeren Kontext stellen und genauer erklären werden.
- Studierendenwunsch!
 - Diese Aufgabe wurde bisher erst kurz vor dem abschließenden Praktikumsprojekt gestellt. Die damaligen Studierenden wünschten sich im Nachhinein, dass diese Aufgabe früher gestellt werden sollte, weil ihnen dadurch einige Zusammenhänge früher klar geworden wären.



Vorübung: Ruby

- Grundlage: Ruby
 - Sie kommen nun in Kontakt mit Controller- und Model-Code. Dieser ist in Ruby verfasst. Für diesen Kurs genügen die Ruby-Grundkenntnisse aus dem ADS-Kurs des letzten Semesters sowie vorgegebene Beispiele zu verwenden bzw. zu modifizieren
 - Wer diese Ruby-Grundlagen nicht hat bzw. sie auffrischen möchte: Bearbeiten Sie auch das Ruby-Tutorial
- Ruby-Tutorial (optional für Studierende mit vorhandenen Ruby-Grundlagen)
 - Ausgangspunkt für die Beschäftigung mit Ruby ist die Homepage dieser Sprache: **<https://www.ruby-lang.org>**
 - Im rechten Seitenkasten finden Sie Rubrik „Der Einstieg ist einfach“
 - Aufgabe 0.1: Nachvollziehen der Schritte aus „Ruby in 20 Minuten“
 - Auch die irb-Beispiele bitte selbst abtippen und testen, nicht nur lesen
 - Datei **ri20min.rb** bitte abgeben und bei der Abnahme kurz zeigen, vor allem aber den Code erklären können.
 - Aufgabe 0.2: Lesen von „Ruby für Umsteiger“ + „Von Java zu Ruby“

Ziele

- Ziele
 - Erarbeiten Sie sich die in http://guides.rubyonrails.org/v6.1/getting_started.html beschriebenen Grundkenntnisse zur Erzeugung eines vollständigen Rails-Projekts.
 - Die für die Entwicklung von Model-Methoden benötigten Ruby-Grundkenntnisse haben Sie soeben im Tutorial erworben, tiefere Ruby-Kenntnisse sind hier nicht erforderlich.
- Vorgabe
 - Implementieren Sie eine Variante des Rails-Projekts „Blog“ aus „getting_started.html“ mit folgenden Unterschieden:
 - Verwenden Sie für die Views HAML statt HTML & erb
 - Stellen Sie das Beispiel aus der Anleitung um auf deutsche Texte
 - Bem.: Reines Abschreiben des Beispiel-Codes gibt max. ½ Punkt!
 - Tipp:
 - Wenn Sie unsicher sind, bringen Sie erst das Original-Beispiel mit HTML & erb ans Laufen und stellen dann um auf HAML (vgl. Aufg. 01) und deutsche Texte

- Der Weg
 - Denken Sie über die Aussagen aus Abschnitt 2 „*What is Rails?*“ nach
 - Sie sollten bei der Abnahme Fragen dazu beantworten können
 - Befolgen Sie die Anleitungen *ab* Abschnitt 3.2 „*Creating the Blog Application*“.

Abweichungen dabei:

- Verwenden Sie **rails new blog**
- Übernehmen Sie dann Datei „Gemfile“ aus Ihrem Projekt „wba1“
- Bauen Sie das Layout und die Views um auf HAML und HTML5
- **Ersetzen Sie alle Texte durch deutsche**
 - Neben Views auch an Controller und Models (validations) denken!
 - Auch die Fehlertexte behandeln
- **Wichtig:** Beschäftigen Sie sich intensiv mit den Erklärungen von „getting_started“ – dies ist sogar wichtiger als das reine Nachbauen der Blog-Anwendung!

* Tipps

- Views
 - Deutsche Texte bei Element „label“ z.B. wie folgt ergänzen:
`= f.label :title, 'Titel'`
 - Deutsche Texte bei Element „submit“ z.B. wie folgt:
`= f.submit 'Kommentar anlegen'`
 - Layout: Ersetzen Sie „style“ von „body“ durch CSS-Code
- Models
 - Deutsche Fehlertexte bei Validierungen erreichen Sie durch explizite Kontrolle des Meldungs-Texts mit einer Zusatzangabe. Beispiel:
`validates :name, presence: { message: "ist ein Pflichtfeld" }`
- Controllers
 - Bemerkung zu „10 Security“: Passwörter bettet man in echten Anwendungen natürlich nicht – wie hier der Einfachheit halber geschehen – direkt im Klartext in den Quellcode ein!



Bedingungen

- Abgabe
 - Wegen des Feiertags am 13.5. erst in zwei Wochen (KW 21)
 - Jeweils vor Beginn Ihrer Praktikumsgruppe in der Abgabewoche
 - Hinweis: In KW 20 beginnt bereits die Bearbeitung der Übung 05
- Art des Leistungsnachweises
 - Zu vergeben: **1 Punkt** (bis Abschnitt 7), +½ optionaler Sonderpunkt (auch 8-10)
 - **Einzel-Arbeit – keine Teams!**
 - Abgabe der Dateien:
 - Im Rails-Verzeichnis „blog“ ausführen:

```
$ rake log:clear  
$ rake tmp:clear  
$ cd ..
```
 - Sie sind nun im Elternverzeichnis von „blog“. Jetzt noch ausführen:

```
# Ordner ./blog verpacken, ohne unnötige Unterordner:  
$ tar czf 04-blog-<matnr>.tar.gz --exclude blog/tmp \  
--exclude blog/node_modules --exclude blog/log ./blog
```

Datei „04-blog-<matnr>.tar.gz“ abgeben
 - Achten Sie auf die Entfernung unnötiger Daten – Dateigröße von ca. 250 kB ist ok.
Dateien > 1 MB werden nicht angenommen!
 - Kurzes Abnahmegespräch / Online-Demo der korrekten Funktion
 - Online, per Breakout-Session.

Fortsetzungszeile

- **Zu Ruby**

- **<http://ruby-lang.org/>**

- Heimat-Seite dieser Sprache
 - Darin erarbeiten: „Ruby in 20 Minuten“ und „Ruby für Umsteiger“
 - Dokumentation der Ruby-Klassen: <https://ruby-doc.org> (insb. „Core 3.0.1“)

- **Zu Rails**

- **<http://guides.rubyonrails.org/v6.1>**

- Gute Hilfen für den Einstieg in die Entwicklung mit Rails, mit zahlreichen Beispielen
 - Hier im Fokus: „**Getting started**“
 - Weiterführend: Die Guides zu „Model“
 - Optional: „Testing Rails Applications“

- **<http://api.rubyonrails.org/v6.1>**

- Für alle, die es genau wissen wollen oder müssen