



BWOLF

EINE WEBBASIERTE PLATTFORM ZUR
EINSCHREIBUNG UND VERWALTUNG DES
EMPIRIEPRAKTIKUMS AN DER FSU JENA

Anforderungsanalyse

Christoph Keiner, Matthias Reuse, Ingo Schäfer, Christoph Staudt

26. November 2017

Inhaltsverzeichnis

1	Allgemeine Problemstellung	1
2	Genaue Anforderungen an das System	1
2.1	Ablauf eines Praktikumsmoduls	1
2.2	Verschiedene Sichten im Überblick	2
2.2.1	Sicht der Administratoren	2
2.2.2	Sicht der Dozenten	2
2.2.3	Sicht der Studenten	3
3	Der Verteilungsalgorithmus	4
4	Technische Details	4

1 Allgemeine Problemstellung

Im Zuge des Projektes gilt es eine webbasierte Plattform zu schaffen, die die Verwaltung des Empiriepraktikums der FSU Jena ermöglicht. Hauptaufgabe dieser Plattform ist, den verantwortlichen Mitarbeitern der FSU Jena zu ermöglichen, für ein kommendes Semester ein neues Praktikumsmodul mit allen zugehörigen Kursen anlegen zu können. Des Weiteren sollen die Studenten die Möglichkeit erhalten eine Präferenzliste zu erstellen, welche der angebotenen Kurse sie am liebsten besuchen möchten. Nach Ablauf einer vorher festzulegenden Frist sollen die Studenten gemäß ihrer Präferenzliste dann automatisch bestmöglich auf die Kurse verteilt werden.

2 Genaue Anforderungen an das System

Gemäß der allgemeinen Problemstellung aus Abschnitt 1 existieren verschiedene Sichtweisen auf die Anforderungen des Systems. Zum einen die Sicht der Verantwortlichen für Praktikum und Kurse, zum anderen die der Studenten. Erstere teilt sich wiederum auf in den Blickwinkel der Dozenten der einzelnen Kurse und der übergeordneten Verantwortlichen für das Empiriepraktikum, im Weiteren Administratoren genannt. Im Folgenden werden zunächst die Anforderungen für Studenten, Dozenten und Administratoren anhand des chronologischen Ablaufs eines Praktikumsmoduls dargestellt. Anschließend wird in Abschnitt 2.2 nochmal eine kurze Übersicht über die Anforderungen der jeweiligen Sichtweisen gegeben.

2.1 Ablauf eines Praktikumsmoduls

Der Ablauf beginnt, indem die Administratoren, nachdem sie sich in einer Login-Oberfläche angemeldet haben, ein neues Praktikumsmodul erstellen. Zu diesem Praktikumsmodul gehören neben generellen Informationen wie Name und Semester auch die besonderen Angaben, ab wann Studenten ihre Präferenzliste erstellen können und zu welchem Zeitpunkt die automatische Verteilung vorgenommen werden soll. Im Anschluss können die Dozenten, nachdem auch sie sich in einer entsprechenden Oberfläche angemeldet haben, ihre Kurse zu dem aktuellen

Praktikumsmodul hinzufügen. Dabei sollen Kurse Angaben über Titel, Dozent, Teilnehmerzahl, Ort, Zeit, Beschreibung des Kurses und Literaturliste besitzen.

2.2 Verschiedene Sichten im Überblick

2.2.1 Sicht der Administratoren

Ein Administrator hat die Möglichkeit, neue Dozenten zu erstellen, diese zu bearbeiten und zu löschen. Zusätzlich ist er dafür verantwortlich, die Ergebnisse des Verteilungsalgorithmus zu verändern und zu bestätigen.

2.2.2 Sicht der Dozenten

Ein Dozent hat die Möglichkeit, Kurse für das Praktikum zu erstellen. Für einen Kurs müssen sie folgendes angeben:

- Name
- Titel
- Dozenten
- Zeit/Raum
- Maximale Teilnehmer (5 oder 10!)
- Kurzbeschreibung
- Beschreibung
- Literatur
- E-Mail Adresse des Empirie-Praktikums-Leiter

Folgendes ist nicht für den Studenten einsehbar:

- Lehrstuhl
- Lehrauftrag
- Erstes Mal Empiriepraktikum?

Nachdem die Studenten verteilt worden sind, erhalten die Dozenten eine E-Mail mit den Studenten, die in ihrem Kurs sind. Im Verlaufe des Semesters können Dozenten ihren Kurs in zwei Kurse aufteilen.

Sollten Dozenten zu einem anderem Semester erneut Kurse anbieten, so haben sie die Möglichkeit, ihre alten Kurse einzusehen und (ohne Änderung des alten Kurses) einen dieser als Vorlage für einen neuen Kurs zu nutzen.

2.2.3 Sicht der Studenten

Drei Wochen vor Verteilung der Kurse beginnt die Einschreibungszeit. Hierfür erhalten die Studenten unabhängig von diesem System eine Benachrichtigung. Studenten sehen bei ihrem Besuch zuerst eine kurze Erläuterung zu dem Einschreibungs-System. Neben dem Text gibt es noch ein Feld zur Anmeldung bzw. Registrierung. Nachdem ein Student sich registriert hat und angemeldet ist, wird er zur Liste der derzeitigen Kurse weitergeleitet. Diese Kurse können allerdings auch ohne Anmeldung angesehen werden.

Durch eine weitere Seite, können Studenten dann jeden Kurs nach ihrer Präferenz sortieren. Dabei müssen Studenten jedem Kurs eine Präferenz zuteilen. Sollten Studenten dann ihre Präferenzenliste bestätigen, so wird ihnen an ihre hinterlegte E-Mail-Adresse ihre gewählte Präferenzenliste geschickt. Diese Liste kann beliebig verändert werden, bis die Verteilung beginnt. Mit jeder Änderung der Präferenzenliste erhalten Studenten eine neue E-Mail. Sobald die Verteilung abgeschlossen und durch einen Administrator bestätigt wurde, erhalten Studenten eine E-Mail-Benachrichtigung, in dem ihr zugeteilter Kurs drin steht. Studenten sehen nun auch für alle Kurse die Namen der Teilnehmer.

Nach der Verteilung beginnt die Tauschperiode. Bis zum Beginn der ersten Vorlesungswoche können Studenten Tauschanfragen an andere Studenten schicken. Jede dieser Tauschanfragen wird automatisch an den Empirie-Praktikums-Leiter weitergeleitet. Ist eine Tauschanfrage erfolgreich, so tauschen die Studenten ihre Kurse.

3 Der Verteilungsalgorithmus

Der Verteilungsalgorithmus verteilt alle Studenten auf die Kurse. Dabei ist es wichtig, dass die aufaddierte maximale Teilnehmeranzahl der Kurse größer ist als die Anzahl der zu verteilenden Studenten. Sollte dem nicht der Fall sein, so wird der Administrator informiert. Dieser hat dann die Möglichkeit die maximale Teilnehmeranzahl von Kursen zu erhöhen. Die Eingabe des Algorithmus sind die Gewichte der Präferenzen und (optional) die Auswahl eines Optimierungsalgorithmus.

Startet der Algorithmus, so versucht er, für jeden Studenten die größte, mögliche Präferenz zu den Kursen zu wählen. Dabei ist insbesondere wichtig, dass die Streuung der gewählten Präferenzen möglichst gering ist. Dies erfolgt beispielsweise durch eine Gewichtung der Präferenzen.

Weiterhin platziert der Algorithmus in jeden Kurs mindestens drei Studenten, damit der Kurs sinnvoll angeboten werden kann. Es werden aber nie mehr Teilnehmer einem Kurs zugeordnet als die maximale Teilnehmeranzahl des Kurses vorgibt.

Hat der Algorithmus schließlich eine passende Zuordnung von Studenten zu Kursen gefunden, so wird der Administrator zuerst benachrichtigt. Dieser kann die Ergebnisse bearbeiten, Parameter des Algorithmus neu einstellen, Kurse aktivieren/deaktivieren und ihn erneut starten. Ist der Administrator zufrieden mit der Verteilung, so bestätigt dieser das Ergebnis. Anschließend wird an die Dozenten eine E-Mail mit ihren Teilnehmern geschickt, und Studenten erhalten eine E-Mail mit ihrem Kurs.

4 Technische Details

Der Login wird über eine E-Mail/Passwort Authentifizierung realisiert. Dafür werden nur E-Mail Adressen der FSU Jena nutzbar sein. Dies bedeutet, dass E-Mails auf "@uni-jena.de"enden müssen. Man kann sich zu jeder Zeit ein- und ausloggen.

Kurse können öffentlich eingesehen werden. Kurse können im Backend einge-

sehen werden, d.h. Dozenten und Administratoren können dies tun. Zum Filtern können insbesondere folgende Kriterien genutzt werden:

- Lehrstuhl
- Finanzierung/Lehrauftrag
- Erstes Mal Empiriepraktikum?

Bei der Wahl der Präferenzen darf keine Präferenz doppelt belegt werden, d.h. jede Präferenz hat einen eindeutigen Kurs.

Der Verteilungsalgorithmus kann mit verschiedenen Optionen angesteuert werden. Optionen können u.a. die Gewichtung der Varianz, oder welche Kurse belegt werden können, sein. Weiterhin berechnet der Verteilungsalgorithmus innerhalb von 24 Stunden die Verteilung aller Studenten.

Die Website wird auf einer docker-compose Umgebung aufgesetzt. Im Backend wird OctoberCMS mit Laravel 5.5 genutzt. Dafür darf kein Browser, der älter als IE11 ist, genutzt werden. Sollte ein Nutzer mit einem älteren Browser die Seite besuchen, so wird ihm angezeigt, dass er seinen Browser aktualisieren sollte. Weiterhin muss die Website auf mobilen Endgeräten funktionieren.

Der Webserver nutzt Nginx. Die Datenbank nutzt MySQL. Der Cache-Server wird mit Redis umgesetzt.

Tests werden für die wesentlichen Bestandteile, insbesondere für den Algorithmus genutzt werden.