

SWT Projekt: Individueller interaktiver Studienplaner.

Zielstellung:

- Erleichterung des Semesterplanung.
- Interaktiv
- Modulare Anordnung
- Restriktionen

(mir war langweilig, hab mal grob angefangen irgendwie das ganze Ding zusammenzufassen, keine Ahnung wie viel wir tatsächlich in die Zielstellung packung sollen (und die Formulierungen sind noch whack) - Mike)

Das Ziel der Anwendung ist es, Studierende bei der Planung ihres Studiums zu unterstützen. Dabei ist es irrelevant, ob sie sich im ersten oder beliebig hohen Semester befinden.

Dazu ermöglicht das System eine modularer Anordnung der einzelnen Fächer. Man kann sich diese frei und interaktiv zusammenstellen, wie man möchte. Jedoch gibt es bestimmte Fortschrittsregelungen und Abhängigkeiten zwischen den Fächern. Sobald man Regelungen bricht, wird man gewarnt.

Für die Fortschrittsregelungen muss man seinen Plan entsprechend anpassen. Dies sieht dann so aus, dass man bestimmte Module gar nicht belegen kann, weil man andere Module noch nicht abgeschlossen hat.

Bei den Modulabhängigkeiten kann man entscheiden, ob man das Risiko eingeht, dass einem Kompetenzen für das Fach fehlen, oder ob man seinen Plan auch da verändert.

Kompetenzen werden nämlich von vielen Modulen erwartet, das heißt man sollte die entsprechenden Fächer schon erledigt haben, allerdings ist es keine Pflicht. Nach eigenem Ermessen kann man auch ohne bestandenes Modul andere Module belegen, die darauf aufbauen.

Außerdem kann man dem System ein Limit an CP pro Semester geben. Damit ist es auch möglich Studium und Arbeit zu vereinen und entsprechend seine Semester zu planen.

Admins haben die Möglichkeit per Textdatei verschiedene Studiengänge im System zu hinterlegen. Diese müssen ein bestimmtes Format haben, damit das System sie fehlerfrei einlesen und verarbeiten kann. Damit ist man nicht an einen Studiengang gebunden und kann variabel Fortschrittsregelungen, Abhängigkeiten, CP-Anzahl pro Modul u.v.m. einstellen.

Anwendungsszenarien:

Erstsemester:(Lyra)

Die Studentin Siegfriede hat gerade erst angefangen zu studieren und wurde auf den Studienplaner aufmerksam gemacht. Sie muss sich erst einmal zurechtfinden und folgt den bereitgestellten **Tooltips/Anweisungen** um sich mit dem Planer vertraut zu machen. Zu Beginn sucht sie Ihre Uni/Hochschule aus und legt ihren Studiengang fest. Nachdem dies getan ist schaut sie sich die für ihr Semester angebotenen Fächer an und deren CP-Anzahl und plant sich fröhlich das ganze Semester mit allen möglichen Fächern zu, bis sie von dem System davor gewarnt wird, dass die maximal empfohlene CP-Anzahl erreicht ist. Enttäuscht darüber, dass sie nicht alles auf einmal lernen kann, entfernt Sie die zusätzlichen Fächer aus dem zweiten Semester und legt diese schon einmal präventiv für das nächste Semester an. Damit sie auch nicht vergisst was sie alles ausgewählt hat, möchte sie das erste Semester gerne ausgedruckt an Ihrem Schreibtisch haben und konvertiert das Ganze in eine **"PDF?"-Datei** um es dann auszudrucken.

Fortschrittsregelung: (Chrys)

Fach ohne Voraussetzungen (Moik):

Der Student Friedfert möchte im kommenden Semester das Modul Computergrafik aus dem vierten Semester belegen. Allerdings fehlt ihm Programmieren 3 aus dem dritten Semester. Da Computergrafik die Kompetenz Python aus diesem Modul voraussetzt wird ihm warnend angezeigt, dass er noch diese fehlende Kompetenz hat und daher besser erst Programmieren 3 abschließen sollte. Aufgrund dessen entscheidet sich Friedfert dazu, Computergrafik in diesem Semester erstmal nicht zu machen und wählt es für ein späteres Semester aus. Damit verschwindet auch die Warnung.

Studiengangs Admin (Peblo):

Die Frau Joghurta, aus der Verwaltung des Studienganges der Sozialen Arbeit hat zur Aufgabe bekommen, den Studentenplaner für den Studiengang anzulegen. Dazu meldet sie sich in dem Planer als Studiengangs Admin für die Soziale Arbeit an. Das Programm bietet eine Möglichkeit an, eine Textdatei mit allen Modulen und deren Infos mit vorgegebenen Format hochzuladen. Nach dem Erstellen der Datei, lädt sie diese hoch, bekommt aber erstmal eine Meldung, dass eine Zeile fehlerhaft ist. Diese verbessert sie, und kann nach erneuten Hochladen überprüfen, ob alles so funktioniert wie es sollte.

Funktionale Anforderung:

- Das System MUSS Speicherung / Einlesen (vom personalisierten Plan) ermöglichen
- MUSS für verschiedene Studiengänge nutzbar sein.
- MUSS Modulare Konfiguration ermöglichen
- Einträge MUSS(en) korrigierbar sein.
- MUSS Module nach Fortschrittsregelung (wenn gegeben) kontrollieren
- MUSS fehlerhaft eingelesene Dateien erkennen können (Plan Einspeisung)

- Login / Registrierung SOLLTE möglich sein
 - System SOLLTE darauf hinweisen, wenn benötigte Voraussetzungen (Module in Abhängigkeiten) für ein gewisses Modul nicht gegeben sind
 - Abhängigkeiten SOLLTEN angezeigt werden (Fach anklicken -> dependencies werden hervorgehoben)
 - Das System SOLLTE einen CP-Zähler für bestandene Module anzeigen. Und auch jeweils für die einzelnen Semester.
 - SOLLTE Anwesenheits-Tracker für Praktika enthalten
 - SOLLTE ansprechend gestaltet sein
 - Module SOLLTE man mit geschafft oder durchgefallen(1/2x?) markieren können.
 - SOLLTE Möglichkeit des Reset anbieten
-
- Das System WIRD dir deinen Stundenplan vorschlagen oder anpassen, falls du ein geplantes Modul nicht schaffen solltest.
 - Das System WIRD Werbung für die Partner-Unternehmen der Universitäten / Hochschulen schalten (Inspiration für Praktika etc.)
 - Das System WIRD Sportplanung / Sprachkurse integrieren
 - Das System WIRD ein gruseliges Maskottchen erhalten (Karl die Klammer)
 - Das System WIRD gamifiziert! (onewayticket)

Anwendungsfälle:

- Vorbereitung: Auswahl von bestandenen und nicht bestandenen Fächern
- Kursbelegung (Auswählen von Kursen die man machen kann und will)
- Anlegung von Studiengängen (Admin) -> ERROR-Case (z.B.Datei falsch formatiert)
- Korrigieren (ändern) von Einträgen
- Abhängigkeiten anzeigen lassen
- Login/Registrierung
- Abspeichern/Herunterladen von Semester-Plänen
- Studiengang-Wechsel oder Clear-Button / Reset
-

Vorlage:

Titel:

Akteure:

Fachlicher Auslöser:

Vorbedingungen:

Standardablauf:

Alternative Abläufe / Fehlersituationen / Sonderfälle:

Nachbedingung / Ergebnis:

Nicht-funktionale Anforderungen:

Parametrisierbarkeit / Flexibilität:

Nutzungshäufigkeit / Mengengerüst:

Titel: Anlegung von Studiengängen [Lara Reitz]

Akteure: Administration

Fachlicher Auslöser:

Daten zur Bearbeitung der Semesterpläne müssen erfasst und eingespeist werden, damit Studenten ihre Pläne erstellen können

Vorbedingungen: Informationen zu den jeweiligen Studiengängen liegen vor

Standardablauf:

1. Administrator/in: Studiengang wird angelegt
2. Administrator/in: Listenfächer inkl. CP Angabe und Abhängigkeiten werden angegeben
3. System: Vollständigkeit und Plausibilität der Eingaben sicherstellen
4. System: Erzeugen eines leeren Studienplans mit der benötigten Semesterzahl und der einzelnen Listenfach-Komponenten, zum Befüllen des selbigen
5. Administrator/in: Anlage des Studiengangs inkl. der Fächer bestätigen
6. System: Bereitstellung des Studienganges für die Nutzer

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 3a. System erkennt fehlerhafte oder unvollständige Eingabe und erstellt keine entsprechenden Komponenten
 - 3b1. Administrator/in verwirft den angelegten Studiengang
 - 3b2. Administrator/in macht ggf. Anpassungen
- 5a. Administrator/in lehnt Anlage ab
 - 5b1. Administrator/in verwirft den angelegten Studiengang
 - 5b2. Administrator/in macht ggf. Anpassungen

Nachbedingung / Ergebnis:

System geht in Bereitstellungsmodus über

Nicht-funktionale Anforderungen:

geringer Speicherverbrauch, Performance

Parametrisierbarkeit / Flexibilität:

- Löschen, Bearbeiten im Nachhinein möglich
- Anlegen von Vorlagen möglich (Semesteranzahl etc) [optional]

Nutzungshäufigkeit / Mengengerüst:

- Einmalig bei Einrichtung
- Bei Änderungen in der Prüfungsordnung oder Wegfallen des Studiengangs

Titel: Login/Registrierung [Lara Reitz]

Akteure: Student/in

Fachlicher Auslöser: Student möchte sich einen Studienplan erstellen

Vorbedingungen: Studiengang ist in Datenbank eingespeist

Standardablauf:

1. Student/in: Registrierung im System wird angestoßen
2. System: Zeigt Registrierungsmaske an
3. Student/in: Füllt die Maske mit den entsprechenden Daten und bestätigt die Registrierung
4. System: Speichert die Daten des Studenten in Datenbank ab [und sendet Bestätigungsmail an die angegebene Mailadresse - OPTIONAL]
5. System: Fordert Nutzer dazu auf sich einzuloggen
6. Student/in meldet sich auf der Login-Seite mit den entsprechenden Daten an
7. System: gleicht eingegebene Daten mit Datenbank ab. Bei Erfolg: Abruf des bereits erstellten oder Neuerstellung eines Studienplanes

Alternative Abläufe / Fehlersituationen / Sonderfälle:

- 3a. Student bricht die Registrierung ab
 - 3a1. System schließt die Registrierungsmaske und geht zurück zur Startseite (nicht eingeloggt)
- 4a. System: Validation der Daten fehlerhaft.
 - 4a1. System gibt Hilfestellung/Tipps zu den fehlerhaften Spalten an
 - 4a2. Student/in korrigiert ggf. die Eingaben
 - 4a3. weiter bei 3
- 5a. System loggt den Nutzer nach Registrierung automatisch ein
 - 5a1. Weiter bei 7
- 7a System: Validation der Login Daten fehlgeschlagen
 - 7a1. System weist auf fehlerhafte Eingabe hin und fordert auf zum wiederholten Login-Versuch
 - 7a2. [System ermöglicht Passwort-Zurücksetzung per Mail - OPTIONAL]

Nachbedingung / Ergebnis:

System stellt Studienplan bereit und geht in "Idle" Zustand über

Nicht-funktionale Anforderungen:

Effizienter Suchalgorithmus für Login, Sinnvolle Speichermethode für Registrierung um schnelles Suchen beim Login zu ermöglichen

Parametrisierbarkeit / Flexibilität:

Daten Caching zum eingeloggt bleiben wenn gewünscht

Nutzungshäufigkeit / Mengengerüst

Bei jeder Nutzung (je nach Nutzerverhalten von 1 mal Täglich bis 1 Mal pro Semester)

Vorbereitung:
Kursbelegung:
Korrekturen von Einträgen:
Abhängigkeiten anzeigen lassen:
Speichern & Herunterladen von Semester-Plänen:
Studiengang-Wechsel und/oder Clear-/Reset-Button

