



SemestervErwaltungsPlan

Einleitung

Ziel ist die Implementierung eines Programms zur Verwaltung sämtlicher organisatorischer Aufgaben, die im Laufe eines Semesters zur Verwaltung des SEP anfallen. Dafür soll ein Programm entwickelt werden, in welchem die einzelnen Gruppen, Studierenden, Betreuer und sonstige organisatorische Pläne eingebettet sind, um eine reibungslose Organisation der Lehrveranstaltung zu gewährleisten.

Iteration I

In der ersten Iteration werden die grundlegenden Strukturen implementiert, auf deren Grundlage das Programm basiert. Wenn das Programm gestartet wird, soll der Betreuer die Möglichkeit haben, ein entsprechendes Semester auswählen zu können, zu welchem er eine Reihe an Details erhalten möchte, dabei können diese jederzeit editierbar sein. Außerdem lassen sich neue Semester anlegen und auch wieder löschen. Zu jedem Semester soll eine entsprechende Struktur von Klassen, den zu den Klassen gehörigen Gruppen, und den zu den Gruppen gehörigen Studierenden existieren. Des Weiteren soll das Programm nach der Schaffung der grundlegenden Struktur seine Daten atomar, konsistent, isoliert und dauerhaft speichern (ACID-Kriterien) und die Anwendung muss auf mehreren Clients gleichzeitig ausgeführt werden können, ohne die genannten Kriterien dabei zu verletzen.

Iteration II

In der zweiten Iteration soll es zu jedem Semester eine organisatorische Übersicht geben, welche den Wochenplan und den Semesterplan beinhaltet, diese beiden Pläne sollen vom Benutzer angelegt und editiert werden können, um zusätzliche Informationen anzureichern. Basierend auf dem Semesterplan soll für jede Klasse ein Anwesenheitsplan erstellt werden, in dem der Benutzer die Fehlzeiten aller Studenten minutengenau eintragen und übersichtlich einsehen kann. Zu jeder der oben genannten Entitäten (s. Iteration 1) soll der Betreuer die Möglichkeit haben, Notizen hinzufügen zu können. Notizen lassen sich in gut, mittel, schlecht und ohne Zuordnung unterscheiden, dabei unterscheiden sich diese visuell. Darüber hinaus soll dem Benutzer die Möglichkeit geboten werden, ein Benutzerkonto anlegen zu können, auf welches er mit einem zuvor selbst festgelegten Passwort zugreifen kann. Zudem sollen die Benutzerkonten und sämtliche Daten des Programms persistent gespeichert werden können.

Iteration III

In der letzten Iteration wird das Programm um Templates und frei wählbare Zusatzfeatures erweitert. Dem Benutzer soll es möglich sein, einen E-Maildienst mit seinem Benutzerkonto verbinden zu können, sodass er in dieser Iteration automatisiert E-Mails verschicken kann. Der Benutzer soll Templates für das Versenden von E-Mails anlegen können. Außerdem soll der Benutzer die Möglichkeit haben, direkt über das Programm einzelnen Studierenden oder ganzen Gruppen auf Grundlage der Templates E-Mails schreiben zu können. Zuletzt sollen die Gruppenmitglieder drei weitere Zusatzfeatures implementieren, welche zum Thema der Aufgabenstellung passen. Diese Zusatzfeatures sollen zuvor mit dem Betreuer abgesprochen werden.

Mögliche Zusatzfeatures:

- **Nachrichten versenden (Chat)**

Registrierte Nutzer können sich gegenseitig Nachrichten senden, diese werden persistent gespeichert. Außerdem kriegt der Empfänger eine visuelle Rückmeldung, falls er eine neue Nachricht erhalten hat.

- **E-Mails können mit Anhang versendet werden**

Beim Versenden von E-Mails können Dateien angehängt werden.

- **Verknüpfung mit GitLab, um Metriken auf Gruppen-, Klassen- und Personen-Ebene zu erhalten**

Metriken sollen sowohl grafisch, als auch textuell angezeigt werden können.

- **Favorisieren von Gruppen und Personen**

Es können ausgewählte Gruppen und/oder Personen favorisiert werden, welche beim Starten der Anwendung, als erstes angezeigt werden sollen.

- **Datenbestände können als PDF exportieren werden**

Alle Datenbestände eines Semesters lassen sich als PDF-Datei exportieren.

- **Zurücksetzen der Daten**

Dieses Zusatzfeature erlaubt es alle Datenbestände, die in der Anwendung gespeichert sind zurückzusetzen.

Anmerkungen

- Sämtliche Informationen für die Beziehungen zwischen den einzelnen Klassen und der Aufbau der Klassen an sich entnehmen Sie bitte dem beigefügten Entity Relationship Model.
- Das Programm ist in der Programmiersprache Java zu entwickeln.
- Wir veröffentlichen besonders gelungene Software auf unserer SEP-Webseite. Hierzu ist es unbedingt erforderlich, dass das System keine urheberrechtlich geschützten Inhalte (Bilder, Musik, etc.) enthält.