

Requirements für Sitzplatzverwaltungssystem

Stakeholder für das Projekt sind der bewertende Lektor und die Studienkollegen.

Funktionale Requirements

- REQ1: Das Programm muss dem Benutzer die Möglichkeit bieten, sich als Admin mittels Token anzumelden.
- REQ2: Das Programm muss dem Admin die Möglichkeit bieten, manuell Hörsäle zu generieren.
- REQ3: Das Programm soll dem Admin die Möglichkeit bieten, über Eingabe einer Datei mehrere Hörsäle gleichzeitig zu generieren.
- REQ4: Das Programm muss dem Admin die Möglichkeit bieten, Prüfungen mittels Eingabe der Prüfungsinformationen und Raumauslastung zu generieren.
- REQ5: Das Programm muss dem Student die Möglichkeit bieten, sich bei Prüfungen mittels Studienkennung einzutragen.
- REQ6: Das Programm soll dem Studenten die Möglichkeit bieten, seine direkten und indirekten Sitznachbarn bei einer bestimmten Prüfung mittels Eingabe Studienkennung einzusehen.
- REQ7: Das Programm soll dem Admin die Möglichkeit bieten, direkte und indirekte Sitznachbarn einer eingegebenen Studienkennung bei einer bestimmten Prüfung einzusehen.
- REQ8: Das Programm soll dem Admin die Möglichkeit bieten, bei Eingabe einer bestimmten Prüfung den Sitzplan einzusehen.
- REQ9: Bei Programmaufruf soll das Programm zuvor generierte Daten von vergangenen Programmsessions wiederherstellen können.

Qualitätsanforderungen

- Q1: Das Programm ist in englischer Sprache geschrieben.
- Q2: Das Programm kann mit fehlerbehafteten Eingaben des Benutzers umgehen.

Constraints

- C1: Das Programm muss bis zum 13.06. abgegeben werden.
- C2: Für die Entwicklung des Programmes werden nur Anwendungen verwendet, die die FH zur Verfügung stellt oder in anderen Lehrveranstaltung in Benutzung sind.

User Stories und Akzeptanzkriterien

UserStory 1: Adminzugang

Als Admin möchte ich durch Eingabe des Admin-Token den Zugang zu den für meine Rolle freigeschalteten Funktionen (Erstellung Hörsaal, Erstellung Prüfung, Ausgabe des Sitzplan für bestimmte Prüfung) bekommen.

Given: ein Admin-Account, den jeder Admin benutzt

When: Anmelden mit Adminzugangsdaten (Token)

Then: Zugriff zu Adminfunktionen freigeschalten

Given: ein Admin-Account, den jeder Admin benutzt

When: Anmelden mit falschen Zugangsdaten

Then: Meldung „kein Zugang zum Admin-Account“ -> kein Zugriff auf Adminfunktionen

UserStory 2: manuelle Generierung Hörsaal

Als Admin möchte ich durch Angabe der Rauminformationen (Raumbezeichnung, Reihen, Spalten) einen Hörsaal generieren können.

Given: Admin ist eingeloggt

When: Eingabe Rauminformationen

Then: neuer Hörsaal generiert

UserStory 3: Generierung Hörsäle durch Einlesen einer csv-Datei

Als Admin möchte ich durch Eingabe einer csv-Datei, die eine Liste an Informationen für Räume enthält, mehrere Hörsäle gleichzeitig generieren können.

Given: Admin ist eingeloggt

When: Eingabe Name der csv-Datei

Then: neue Hörsäle generiert

UserStory 4: Prüfung generieren

Als Admin möchte ich durch Angabe der Raumauslastung (100/50/25) und Prüfungsinformationen Sitzpläne für bestimmte Prüfungen generieren können.

Given: Admin ist eingeloggt, Hörsaal existiert schon

When: Eingabe Raumauslastung und Prüfungsinformationen (Bezeichnung, Hörsaal, Datum)

Then: neue Prüfung erstellt

Given: Admin ist eingeloggt, Hörsaal existiert schon

When: Eingabe Prüfungsinformationen, die sich schon vorhanden sind

Then: Meldung: „Prüfung existiert schon“ -> Prüfung wird nicht erstellt

UserStory 5: Anmeldung für Prüfung

Als Student möchte ich durch Eingabe meiner Studienkennung, einen Sitzplatz für eine bestimmte Prüfung erhalten, damit ich bei der Prüfung einen festen Platz habe (angemeldet bin).

Given: Prüfung mit freien Sitzplätzen

When: Eintragung mit Studienkennung

Then: zugewiesener Sitzplatz wird angezeigt

Given: Prüfung ohne freie Sitzplätzen

When: Eintragung mit Studienkennung

Then: Ausgabe, dass Benutzer sich nicht für Prüfung anmelden kann

UserStory 6: Sitznachbarabfrage Studenten

Als Student möchte ich durch Eingabe meiner Studienkennung für eine bestimmte Prüfung auswählen können, ob ich direkte oder indirekte Nachbarn abfragen möchte, damit mir meine direkten/indirekten Sitznachbarn für eine bestimmte Prüfung ausgegeben werden.

Given: Prüfung, Student in Prüfung eingetragen

When: Auswahl direkte/indirekte Nachbarn

Then: Ausgabe aller Studienkennungen der direkten/indirekten Nachbarn

UserStory 7: Sitznachbarabfrage Admin

Als Admin möchte ich durch Eingabe einer beliebigen Studienkennung für eine bestimmte Prüfung auswählen können, ob ich direkte oder indirekte Nachbarn abfragen möchte, damit mir die direkten/indirekten Sitznachbarn jener Studienkennung für eine bestimmte Prüfung ausgegeben werden.

Given: Prüfung, Student in Prüfung eingetragen

When: Auswahl direkte/indirekte Nachbarn

Then: Ausgabe aller Studienkennungen der direkten/indirekten Nachbarn