

Homework 1

Aufgabe 1 a)

1. Dozenten

Wie: Dozenten sind Benutzer des Systems, die es verwenden, um Gruppen anzulegen und Details wie Zeitfenster oder Gruppengröße, zu definieren.

Warum: Sind an einem effizienten und gerechten Zuweisungsprozess interessiert. Sie sind an einfacherer Gruppenverwaltung interessiert.

2. Studenten

Wie: Sie sind die Endnutzer des Systems, die die Gruppen auswählen und Informationen abrufen. Einzelne Studenten werden auch direkt als Tester oder Entwickler im Projekt involviert sein.

Warum: Studenten haben ein Interesse daran flexibel und fair Übungsgruppen zugeteilt, zu bekommen. Die Studenten werden eine benutzerfreundliche Plattform erwarten.

3. Lehrstuhl für Software und Systems Engineering

Wie: Organisator der Entwicklung. Bereitsteller von Ressourcen und Entwicklern. Überwacht den Entwicklungsprozess.

Warum: Der Lehrstuhl, will erstens eine erfolgreiche Entwicklung und zweitens ein praktisches, stabiles und nutzerfreundliches System bereitstellen.

4. Software-Entwickler

Wie: Entwickeln und Testen das System.

Warum: Sie wollen, dass System Anforderungsgerecht entwickeln. Sie können dadurch Java und Softwareentwicklungsfähigkeiten und möglicherweise auch eine Vergütung erhalten.

5. IT-Sicherheitsbeauftragte

Wie: Überwachen Sicherheitsmaßnahmen. Prüft Umsetzung der Authentifizierung und des Datenschutzes.

Warum: Sicherheit und Datenschutz sicherstellen, vor allem die richtige Authentifizierung über das Shibboleth-System.

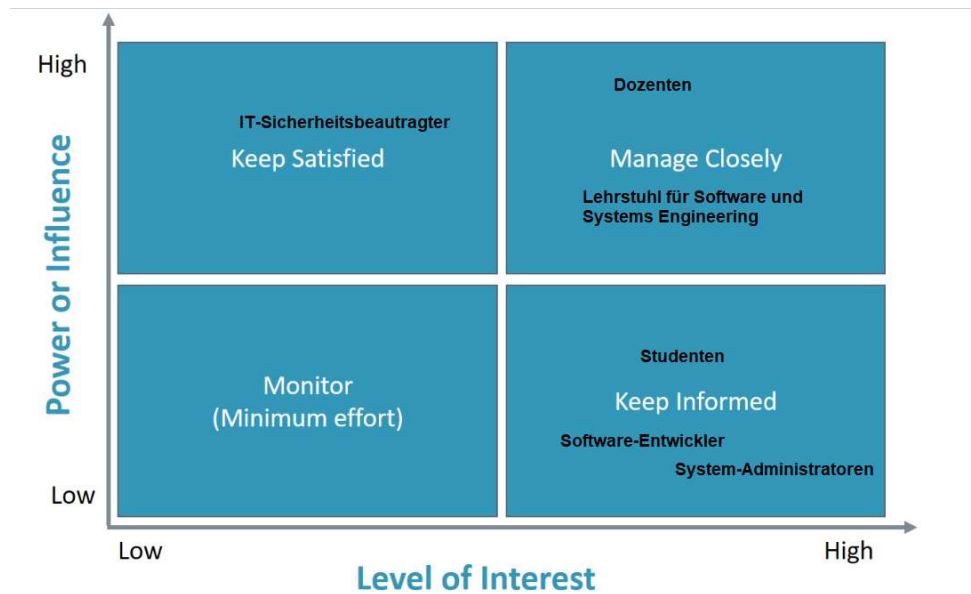
6. Systemadministratoren

Wie: Einrichten der Plattform, Überwachung des laufenden Betriebes, Eingriff bei technischen Problemen und Zuständigkeit für Zugangs- und Datensicherheitsmaßnahmen.

Warum: Sie sind zuständig für die technische Wartung und Support. Sie stellen sicher, dass das System zuverlässig läuft.

Aufgabe 1 b)

1. **High Power, High Interest (Eng Engagieren):** Dozenten und der Lehrstuhl für Software und Systems Engineering
2. **High Power, Low Interest (Zufriedenstellen):** IT-Sicherheitsbeauftragte
3. **Low Power, High Interest (Informiert Halten):** Studenten, Software-Entwickler, Systemadministratoren
4. **Low Power, Low Interest (Beobachten):** ----



Aufgabe 2

A)

1. Dozenten sollen Übungsgruppen für ihre Kurse anlegen können.
2. Dozenten sollen Details, wie Gruppengröße und Veranstaltungszeiten festlegen können.
3. Studenten sollen sich einloggen können um verfügbare Gruppen einsehen, zu können.
4. Studenten sollen durch Benachrichtigungen informiert werden, welcher Gruppe, sie zugeordnet wurden.
5. Das System soll die Studenten fair und unter der Berücksichtigung ihrer Verfügbarkeit und anderen Kurskonflikten auf die Gruppen verteilen.
6. Studenten sollen die Zeiten notieren, während denen sie nicht verfügbar sind.

B)

1. Das System sollte in der Lage sein mit Tausenden Nutzern umzugehen. Skalierbarkeit.
2. Das System soll benutzerfreundlich sein. Usability.
3. Das System muss sicher sein und den Zugang über Shibboleth kontrollieren. Sicherheit.

C)

Das System soll in Java entwickelt sein.

D)

Das Entwicklungsbudget liegt bei 70.000 Euro.

E)

Zugriff aufs System soll durch Shibboleth-Daten kontrolliert werden.

Aufgabe 3

Dozenten sollen Übungsgruppen für ihre Kurse anlegen können.

1. Precision: Teilweise erfüllt. Was genau heißt erstellen (Benennen, Details festlegen, usw.)?
2. Consistency: Erfüllt. Widerspricht keinem anderen Requirements.
3. Verifiability: Erfüllt. Test: Können Dozenten Gruppen erstellen oder nicht?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Dozenten können für ihre Kurse Übungsgruppen erstellen, indem sie den Gruppennamen, die Sitzungszeiten und die maximale Gruppengröße angeben.“

Dozenten sollen Details, wie Gruppengröße und Veranstaltungszeiten festlegen können.

1. Precision: Teilweise erfüllt. Welche Details umfasst „festlegen“.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt. Test: Können Dozenten Details, wie Gruppengröße und Veranstaltungszeiten festlegen?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Dozenten können für jede Übungsgruppe Sitzungszeiten (Tag und Uhrzeit) und eine maximale Teilnehmeranzahl festlegen.“

Studenten sollen sich einloggen können um verfügbare Gruppen einsehen, zu können.

1. Precision: Teilweise erfüllt. Welche Authentifizierung umfasst das „einloggen“
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt, wenn Authentifizierungsmethoden klar spezifiziert werden. Test: Können sich Studenten einloggen oder nicht?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Studenten können sich über ihre Universitätsanmeldedaten (Shibboleth) in das System einloggen, um verfügbare Übungssitzungen für ihre Kurse einzusehen.“

Studenten sollen durch Benachrichtigungen informiert werden, welcher Gruppe, sie zugeordnet wurden.

1. Precision: Teilweise erfüllt. Unklar, wie die Studenten benachrichtigt werden.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt, wenn spezifiziert wird, wie Studenten die Benachrichtigungen erhalten. Test: Erhalten Studenten Benachrichtigungen, die sie informieren welcher Gruppe sie zugeordnet wurden?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Studenten erhalten eine System-Benachrichtigung und eine E-Mail, in der sie über ihre Übungsgruppenzuweisung informiert werden.“

Das System soll die Studenten fair und unter der Berücksichtigung ihrer Verfügbarkeit und anderen Kurskonflikten auf die Gruppen verteilen.

1. Precision: Nicht vollständig erfüllt. Es fehlen konkrete Verteilungsregeln und das Wort fair ist nicht eindeutig.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Nicht erfüllt. Fairness ist ohne weitere Definition nicht klar meßbar.
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Das System verteilt Studierende unter Berücksichtigung ihrer Verfügbarkeiten und bestehender Kurskonflikte auf Übungsgruppen, mit dem Ziel, die Anzahl nicht zugewiesener Studierender zu minimieren.“

Studenten sollen die Zeiten notieren, während denen sie nicht verfügbar sind.

1. Precision: Teilweise erfüllt. Unklar, wie die Zeiten notiert werden.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Teilweise erfüllt, da die Art der Eingabe nicht spezifiziert ist.
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Studenten können in einer Kalenderansicht Zeitfenster markieren, in denen sie nicht verfügbar sind.“

Das System sollte in der Lage sein mit Tausenden Nutzern umzugehen. Skalierbarkeit.

1. Precision: Teilweise erfüllt. Tausende Nutzer ist als Formulierung nicht klar genug.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt, wenn die Systemlast weiter spezifiziert wird. Test: kann das System unter der last von „tausenden Studenten“ regulär funktionieren.
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Das System muss skalierbar sein und mindestens 10.000 Studenten gleichzeitig unterstützen können.“

Das System soll benutzerfreundlich sein. Usability.

1. Precision: Nicht erfüllt. Benutzerfreundlich ist als Begriff zu vage.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Nicht erfüllt. Benutzerfreundlichkeit ist nicht direkt meßbar.
6. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
4. Improvement: „Das System muss benutzerfreundlich sein, indem es eine intuitive Navigation, eine klare Benutzeroberfläche und eine einfache Bedienbarkeit bereitstellt.“

Das System muss sicher sein und den Zugang über Shibboleth kontrollieren. Sicherheit.

1. Precision: Erfüllt. Keine mehrdeutigen Formulierungen.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt. Test: Kontrolliert das System den Zugriff über Shibboleth?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement: „Das System muss sicher sein und ausschließlich über Universitätsanmeldedaten (Shibboleth) zugänglich sein, um unautorisierten Zugriff zu verhindern.“

Das System soll in Java entwickelt sein.

1. Precision: Erfüllt. Keine mehrdeutigen Formulierungen.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt. Test: Ist das System in Java entwickelt?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement:-----

Das Entwicklungsbudget liegt bei 70.000 Euro.

1. Precision: Erfüllt. Keine mehrdeutigen Formulierungen.
2. Consistency: Erfüllt. Widerspricht keinen anderen Requirements.
3. Verifiability: Erfüllt. Test: Wurde das Budget vonüberschritten?
4. Validity: Erfüllt. Ist vereinbar mit den Zielen des Projekts.
5. Improvement:-----

Aufgabe 4

- **Titel:** Anmeldung für Übungsgruppen mit Konfliktlösung
- **Actors:** Student, Dozent und das System
- **Preconditions:** Studenten ist in einem oder mehreren Kursen mit Übungsgruppen und hat Zugriff auf das EGD System.
- **Postconditions:** Der Student ist den Übungsgruppen zugewiesen oder informiert über alternative Lösungsmöglichkeiten bei Konflikten.
- **Triggers:** Semesterbeginn, Abschluss der Kursregistrierung, Benachrichtigung über Anmeldezeitraum, Änderung der Verfügbarkeit.
- **Main Succes Scenario:**
 1. Der Student loggt sich mit den Universitätszugangsdaten ins EGD-System ein.
 2. Der Student sieht die Liste der verfügbaren Übungsgruppen für jeden Kurs, in dem er eingeschrieben ist.
 3. Der Student, trägt die Zeiträume ein in denen er nicht verfügbar ist
 4. Das System analysiert die Verfügbarkeit des Studenten und versucht, ihn einer passenden Übungsgruppe für jeden Kurs zuzuweisen.
 5. Wenn kein Zeitkonflikt vorliegt, weist das System den Studenten automatisch den passenden Übungsgruppen für alle eingeschriebenen Kurse zu.
 6. Der Student erhält eine Bestätigung seiner Übungsgruppenzuweisungen.
- **Alternative Paths:**
 - 4a1: Das System findet einen Konflikt.
 - 4a2: Das System benachrichtigt den Studenten über die Existenz des Konflikts und gibt an, für welche Kurse keine passende Übungsgruppe gefunden wurde.
 - 4a3: Der Student kann Dozenten kontaktieren, um manuell eine Lösung zu finden