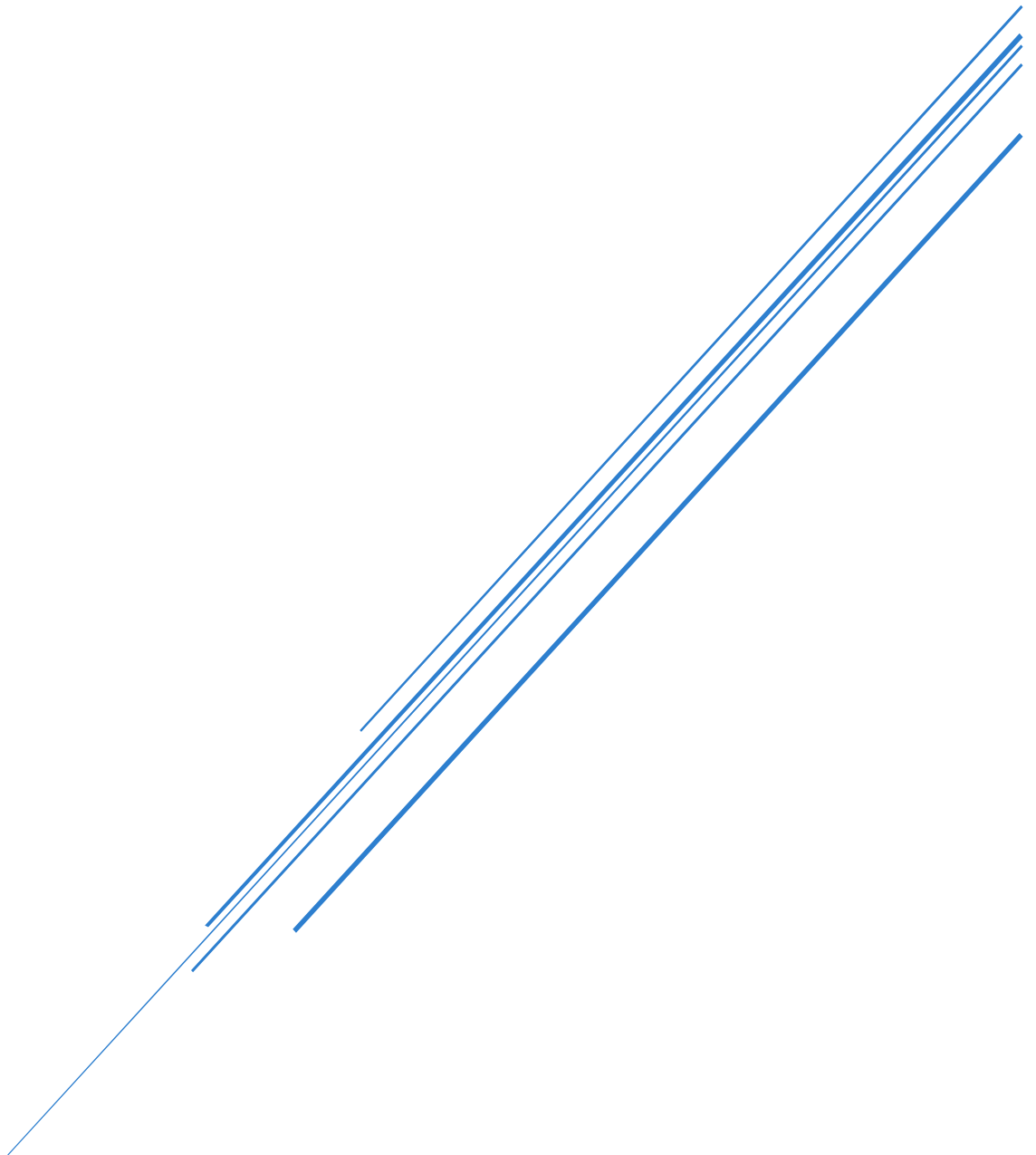


HOMEWORK 1

Softwaretechnik



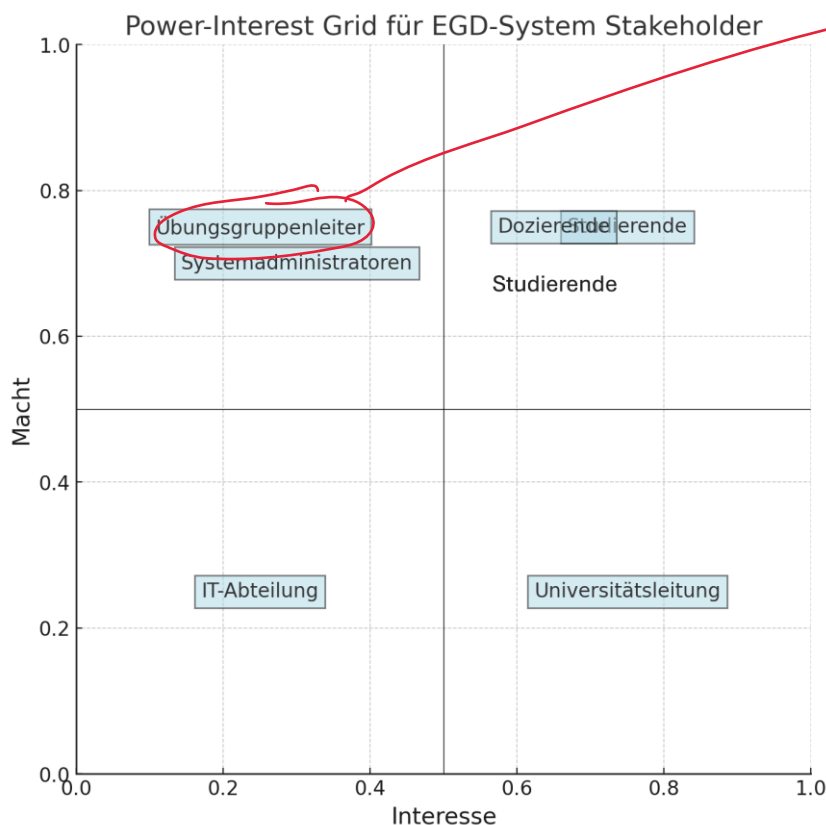
Ruben Charara
7406410

Exercise 1: Stakeholders

a) Identifikation und Einfluss der Stakeholder

1. Studierende: Hauptnutzer des Systems. Sie sind direkt betroffen, da sie basierend auf ihren Verfügbarkeiten automatisch Gruppen zugeteilt bekommen.
2. Dozierende: Verwenden das System, um Gruppen und Zeitpläne zu erstellen. Sie profitieren durch vereinfachte Verwaltung und fairere Verteilung der Studierenden.
3. Systemadministratoren: Sie betreuen das System und sichern den Zugang sowie Datenschutz.
4. Übungsgruppenleiter: Sie erhalten durch das System eine gerechte Zuteilung der Studierenden und können effektivere Gruppen planen.
5. IT-Abteilung: Verantwortlich für die technische Infrastruktur und Wartung, besonders bei einer Erweiterung für eine größere Anzahl an Studierenden.
6. Universitätsleitung: Interessiert an Effizienz, Kosteneinsparung und studentischer Zufriedenheit. Sie können das System universitätsweit ausweiten.

b) Power-Interest Grid



Klingt eher nach hohem Interesse. Aber viel Einfluss haben Übungsleiter eigentlich nicht.

-0.25

3.75

Exercise 2: Requirements

a) Funktionale Anforderungen

1. Das System erlaubt Studierenden die Eingabe ihrer Verfügbarkeiten.
2. Dozierende können Kurse und Gruppensitzungen erstellen.
3. Studierende können sich für mehrere Kurse pro Semester anmelden.
4. Das System verteilt Studierende basierend auf ihren Verfügbarkeiten auf Gruppen.
5. Bei Zuteilungsproblemen informiert das System die betroffenen Studierenden.
6. Benachrichtigungen über Gruppenzuweisungen werden automatisch an die Studierenden verschickt.

/3

b) Qualitätsanforderungen

1. Usability: Benutzerfreundliche Oberfläche für einfache Nutzung.
2. Sicherheit: Authentifizierung durch Shibboleth zur Absicherung persönlicher Daten.
3. Skalierbarkeit: System muss bei steigender Nutzung skalierbar sein.

/1.5

c) Einschränkung

- Das System muss in Java entwickelt werden.

/0.5

d) Projektanforderung

- Einhaltung des Budgets von 70.000 Euro.

/0.5

e) Prozessanforderung

- Erste Testversionen sollen zum Beginn des Wintersemesters 2025/26 verfügbar sein.

Auch ein Projekt log.

/0

Stattdessen: Muss in Java implementiert werden.

//5.5

Exercise 3: Requirements Validation

- Funktionale Anforderungen: Die Anforderungen sind präzise, aber nicht alle sind verifizierbar oder konsistent genug. Zum Beispiel könnte die Zuweisung auf Basis der Verfügbarkeit durch detaillierte Kriterien für "faire Verteilung" verbessert werden.
- Qualitätsanforderungen: Sind im Wesentlichen erfüllbar, könnten jedoch präziser spezifiziert sein. Beispielsweise könnte Usability spezifische Metriken zur Messung der Benutzerzufriedenheit enthalten.
- Verbesserung: Eine detaillierte Spezifikation der Zuteilungsalgorithmen und Erfolgsmetriken könnte die Verifizierbarkeit und Konsistenz erhöhen.

/

Zu knapp.
Mindestens
hätte auf
jedes
Requirement
eingegangen
werden
sollen.

Exercise 4: Use Case

Use Case: Kurs- und Übungsgruppenzuteilung für Studierende

- Akteur: Studierende
- Vorbedingungen: Studierende sind im System registriert und haben sich in ihre Kurse eingeschrieben.
- Ablauf:
 1. Der Studierende meldet sich im System an.
 2. Der Studierende gibt seine Verfügbarkeiten ein.
 3. Das System überprüft und vergleicht die Verfügbarkeiten mit verfügbaren Kurszeiten.
 4. Das System weist den Studierenden passenden Kursgruppen zu.
 5. Der Studierende erhält eine Benachrichtigung über seine Zuweisung.
 6. Bei Konflikten wird eine alternative Zuweisung versucht oder eine manuelle Abstimmung vorgeschlagen.

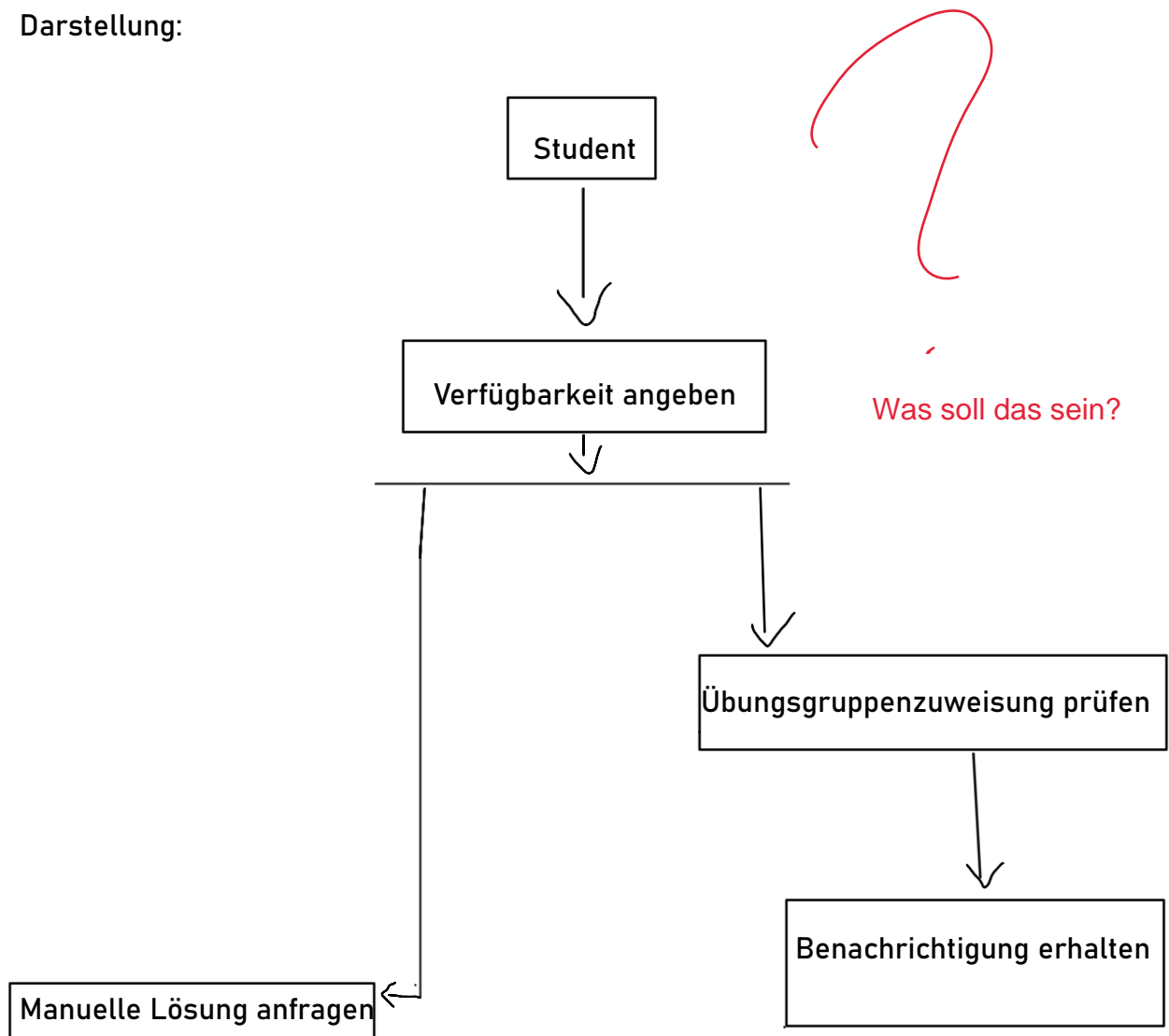
Alternativabläufe?

— 0.5

4.5

14.75/20

Darstellung:



Beziehungen:

1. Verfügbarkeiten angeben → Übungsgruppenzuweisung prüfen: <<include>>
2. Übungsgruppenzuweisung prüfen → Manuelle Lösung anfragen: <<extend>>