

Einsatzplan

Inhaltsverzeichnis

- 1. Einleitung
 - 1.1 Einleitung
 - 1.2 Rahmenbedingungen
 - 2. Anforderungen
 - 2.1. Allgemeine Anforderungen
 - 2.2. funktionale Anforderungen
 - 2.3. Nichtfunktionale Anforderungen
 - 3. Use Case (Anwendungsfälle)
 - 3.1 Use Case Diagramm
 - 3.2 Optional weitere Use Case Diagramme
 - 4. Architektur
 - 5. Geschäftsprozesse
 - 1. *use case Generierung eines Einsatzplan*
 - 2. *use case Abwesenheitsmeldung einer Lehrperson*
 - 6. Datenmodell
 - 7. GUI-Design
 - 8. Klassendiagramm
-

1. Einleitung

1.1 Einleitung

In diesem Projekt geht es um die Entwicklung einer Software Lösung, die Stundenpläne wöchentlich, automatisch erzeugt und an der akademischen Gemeinschaft von "Wissen für Alle - die Wissensakademie" vergeben muss.

1.2 Rahmenbedingungen

Vorher wurde die Planung manuell gemacht, aber da das Geschäft sich stark entwickelt hat. Braucht die Administration eine bessere Art und Weise, die Planung von Kurse :

- zu erstellen,
- zu modifizieren
- und an Studierende und Dozenten von verschiedene Standorten zu teilen.

2. Anforderungen

2.1. Allgemeine Anforderungen

- Dozenten dürfen maximal 16 Stunden pro Woche eingesetzt sein.
- Studenten müssen nachschauen ob eine Vorlesung stattfindet oder nicht
- Bei verschiedenen Standorten benötigen Dozenten 2 Stunden anreise Zeit
- Jede Woche soll ein neuer Stundenplan erstellt werden

2.2 Funktionale Anforderungen

Das System muss allgemein diese Funktionalitäten beinhalten :

1. Jeder Benutzer muss sich einloggen, bevor auf seinem Bereich zuzugreifen
2. Im Allgemeinem Bereich sollen alle Studenten ihren Stundenplan ihres Semesters und Fachbereich einsehen können
3. Ein Studierende muss in der Lage sein :
 - Seinen Stundenplan einzusehen
4. Ein Dozent muss in der Lage sein
 - seinen Stundenplan einzusehen
 - Seine Abwesenheit zu melden
5. Ein Admin muss in der Lage sein :
 - Sich alle Abwesenheiten anzugucken
 - Abwesenheiten bestätigen oder ablehnen
 - Einen Studienplan zu generieren
 - Einen Stundenplan einzusehen
 - Nach einer Generierung oder einer Veränderung eines Stundenplans eine Email zur Signalisierung der Verfügbarkeit an alle Studierende und Dozenten Schicken
6. Ein Stundenplan muss Name des Lehrers beinhalten, des Kurses und des Raums.

2.3. Nichtfunktionale Anforderungen

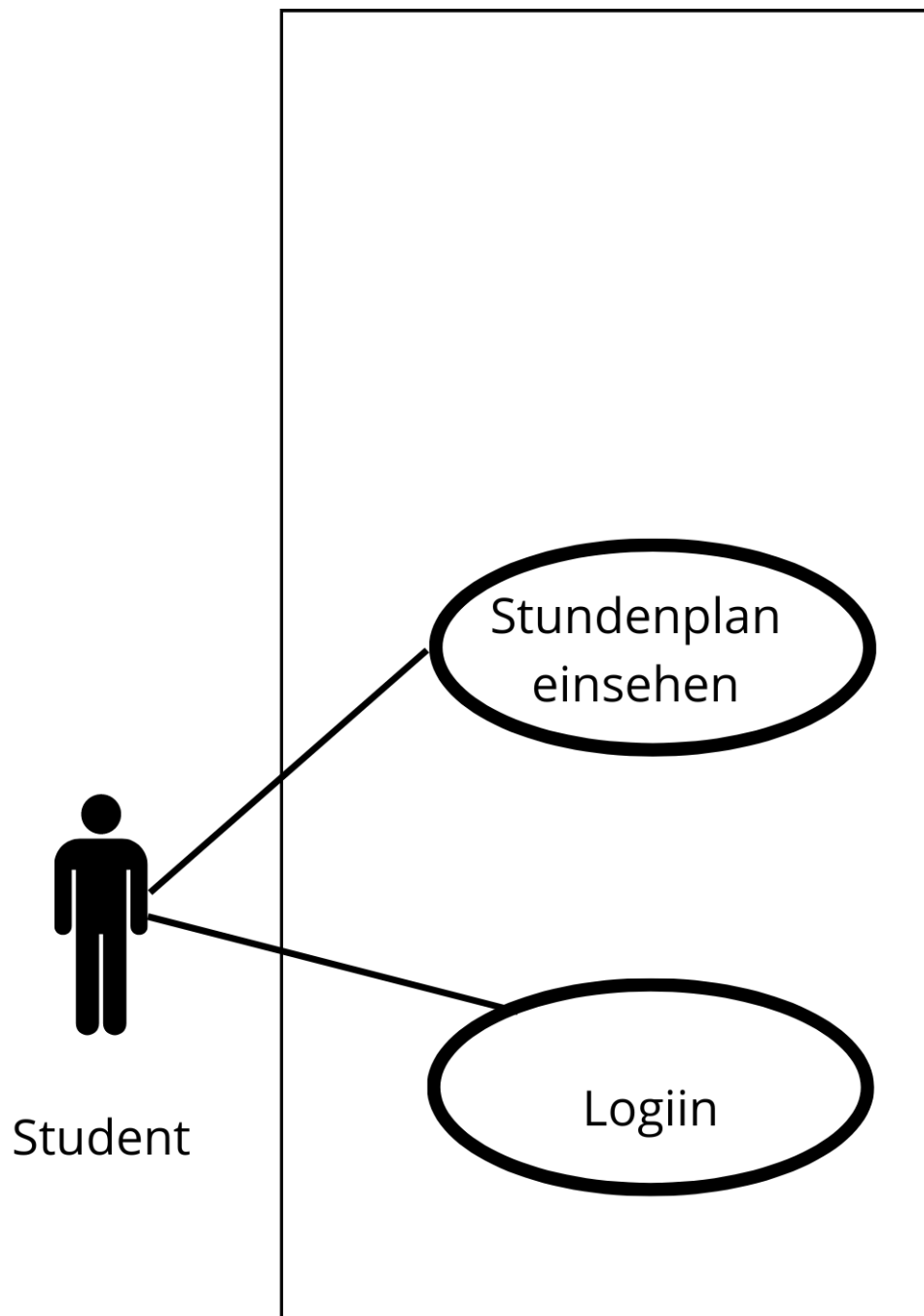
Diese Qualitäten muss das System erfüllen :

- Benutzerfreundlich
- Intuitiv
- Sicher(Ein Studierende oder Dozent darf nie auf den Adminbereich zugreifen)
- Lesbar
- Auf alle Geräte-typen zugreifbar
- System soll verfügbar sein außer bei Updates

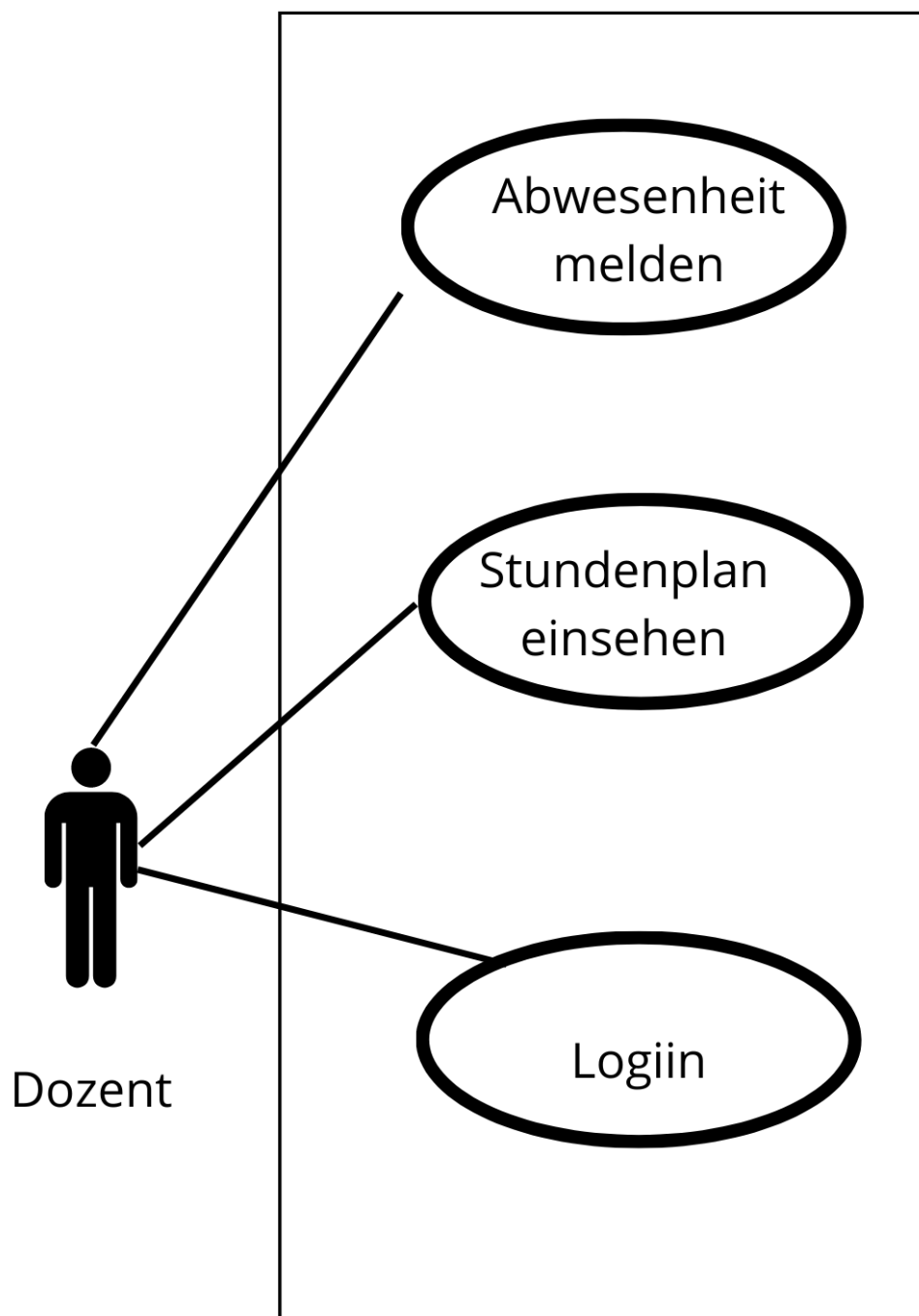
3. Use Case

3.1 Use Case Diagramm

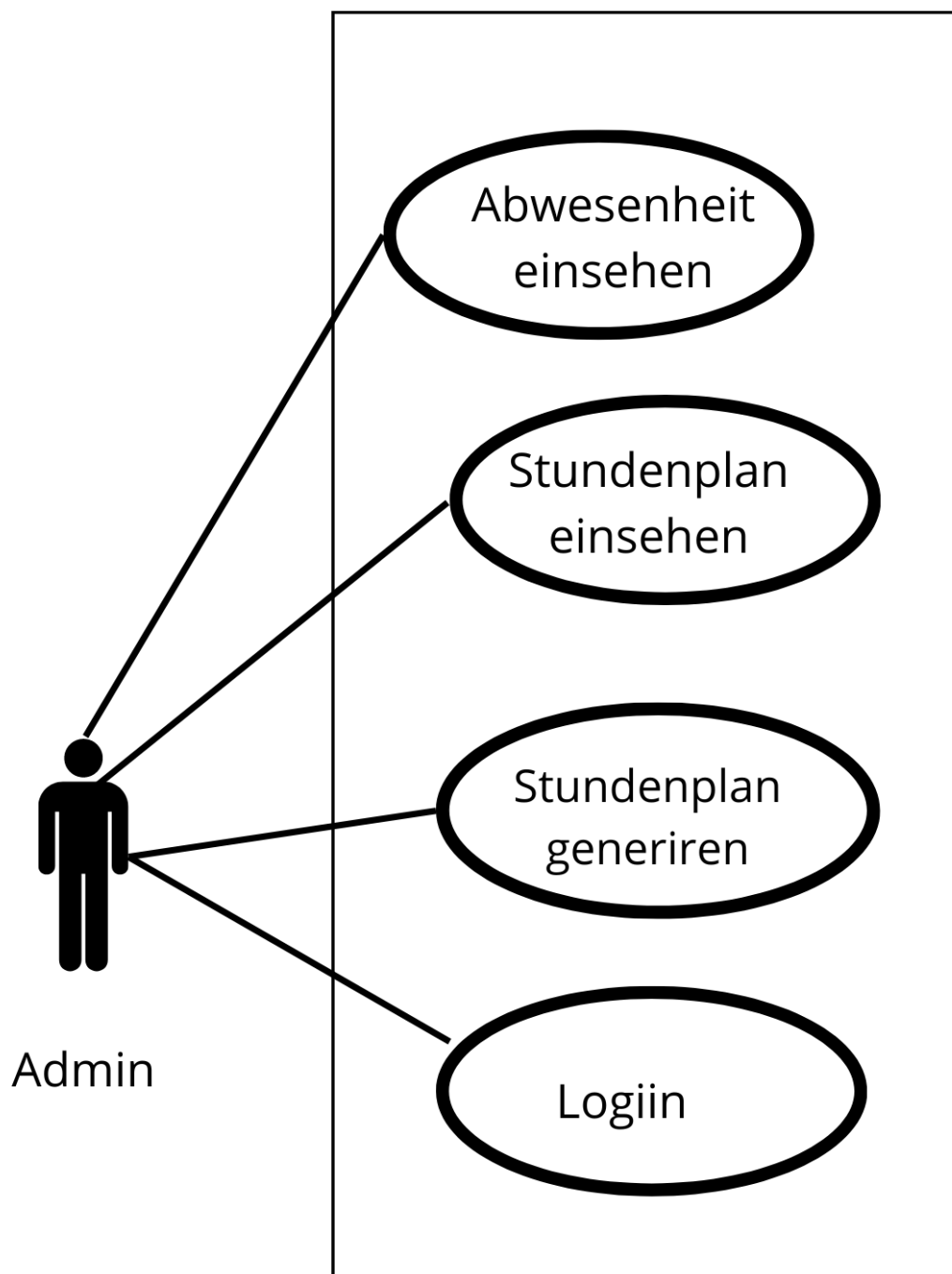
Für einen Studierende



Für einen Dozent

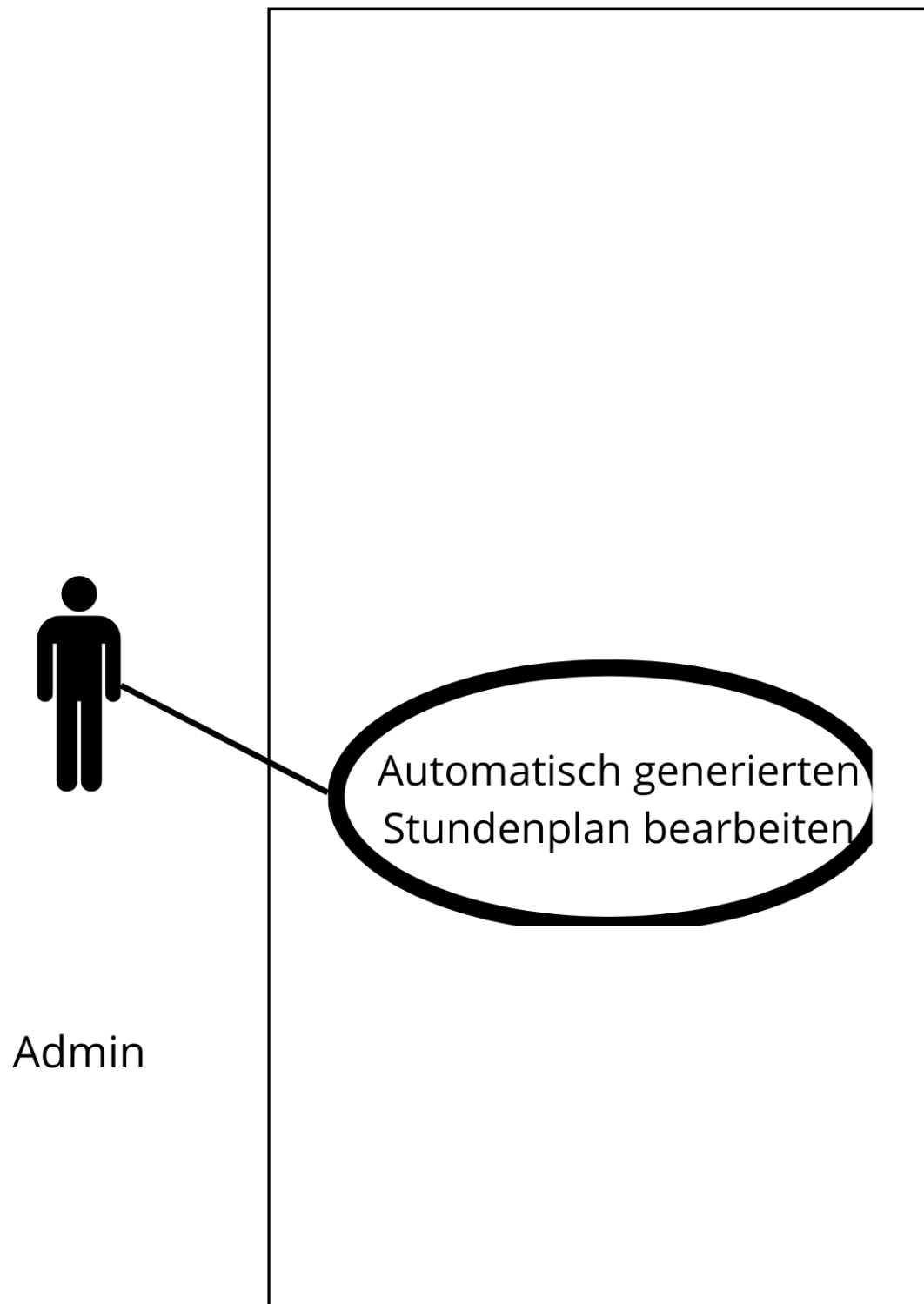


Für einen Admin



3.2 Optional weitere Use Case Diagramme

Für einen Admin



4. Architektur

Wir entscheiden uns für eine Web-App, da man auf alle Geräte und von überall zugreifen kann. Die App wird diese Technologien benutzen:

- Backend: Mit Python, weil es einfach zu lernen ist, HTTP Anfragen gut unterstützt, was sehr wichtig für eine Web-App ist, und es gibt. große Entwickler Community.
- Frontend: Mit HTML, CSS und JS, weil alle Browser das kann, große Entwickler Community die es benutzt, Programmierer haben, damit Erfahrung.
- Datenbank: Mit MySQL, strukturiert, gut für das Suchen, große Entwickler Community

5. Geschäftsprozesse

- Beschreibung Laufzeit des Systems

5.1. *use case Generierung eines Einsatzplan*

Ein Admin loggt sich im System ein und generiert einen Einsatzplan.
Nach der generierung soll eine Benachritigung per Mail an Studierende geschickt werden .

5.2. *use case Abweseheintmeldung einer Lehrperson*

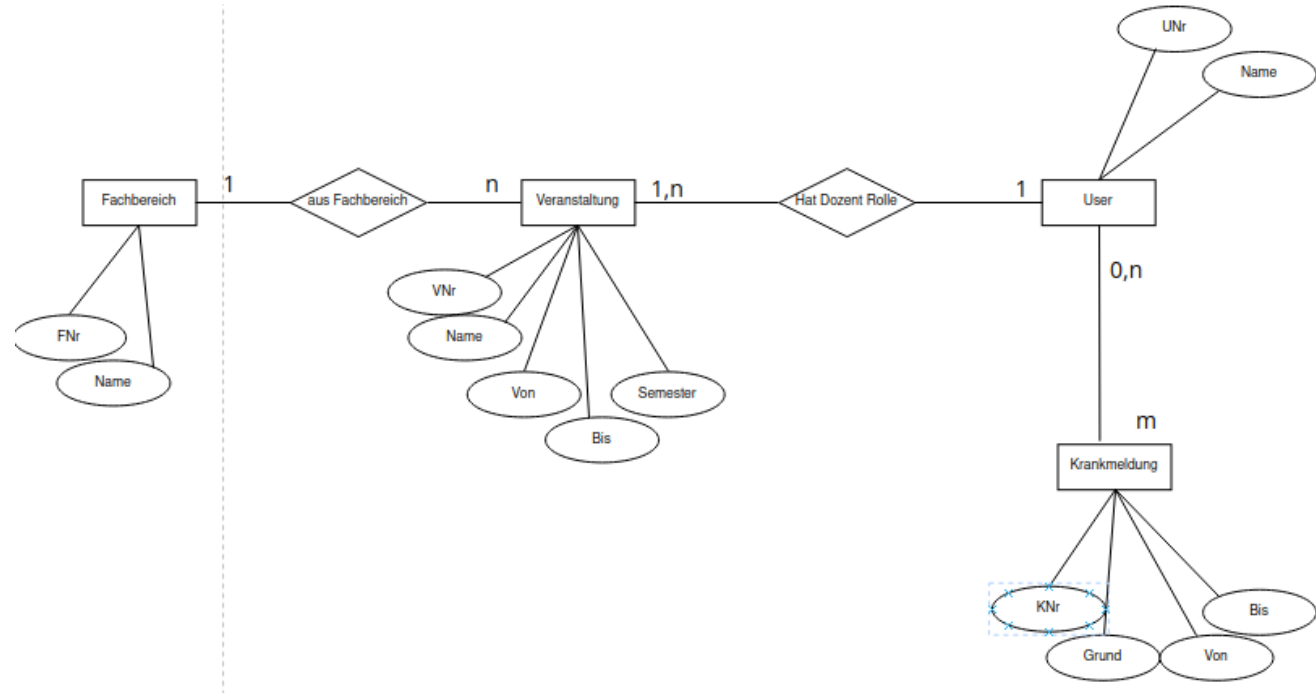
Eine Lehrperson loggt sich in seinen Bereich ein, füllt das Formular(Name, Grunc, Datum) aus, und schickt es.
Ein Admin listet alle Abwesenheiten und hat die Möglichkeit zu akzeptieren oder abzulehnen.
Eine Annahme löst eine neue generierung des Einsatzplan und das Schicken eine Mail an alle Dozenten und Studierende.

5.3. *use case Abwesenheitsmeldungen eisehen*

Nach dem Login kann ein Admin alle Meldungen sehen, sie bestätigen oder ablehnen.

6. Datenmodell

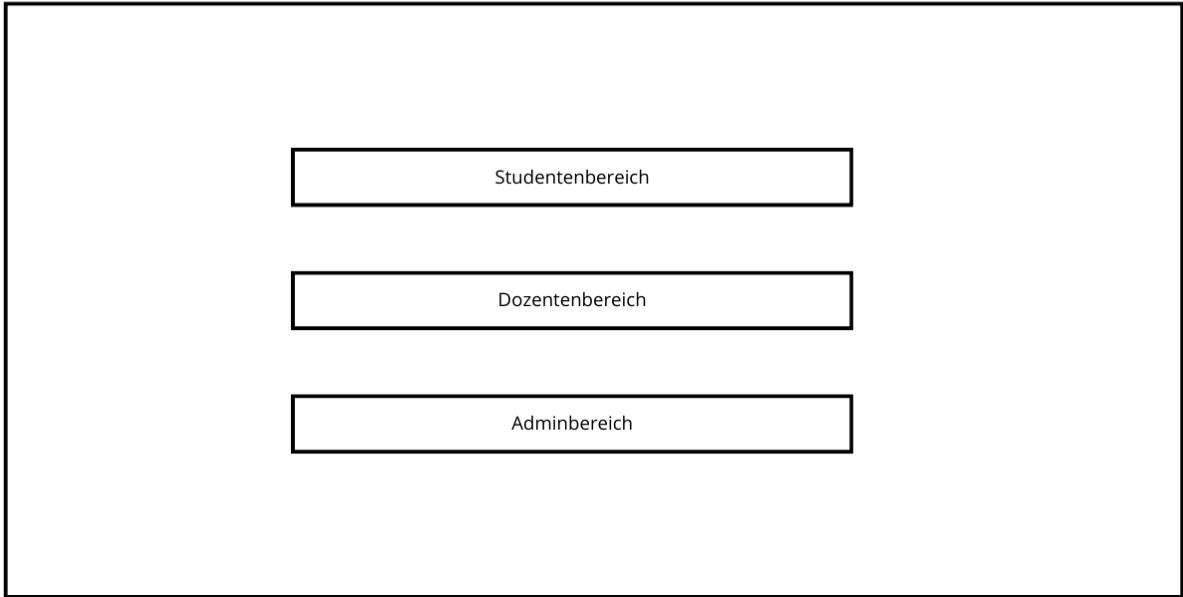
Es wird eine My-sql Datenbank verwendet



7. GUI-Design

Allgemein

Startseite



Login

Startseite > Login

Username :

Password :

Für einen Studierende

Studentenbereich

Startseite > Login > Studentebereich

Student Name

Infos über die person

- Email Adresse
- Semester

Studentenbereich

Startseite > Login > Studentenbereich > Stundenplan einsehen

Student Name

Infos über die person

- Emaill Adresse
- Semesterzahl

Stundenplan

Mo.	Di.	Mi.	Do.	Fr.

Für einen Dozent

Dozentenbereich

Startseite > Login > Dozentenbereich

Student Name

Infos über die person

- Emaill Adresse
- Semester

Stundenplan einsehen

Abwesenheit melden

Dozentenbereich

Startseite > Login > Dozentenbereich > Stundenplan einsehen

Dozent Name

Infos über die person

- Emaill Adresse
- Spezialität

Stundenplan

Mo.	Di.	Mi.	Do.	Fr.

Abwesenheit melden

Startseite > Login > Dozentenbereich > Abwesenheit melden

Name :

Grund :

Datum :

Schicken

Ablehnen

Für einen Admin

Adminbereich Startseite > Login > Adminbereich

Student Name

Infos über die person

- Email Adresse
- Semester

Abwesenheiten einsehen

Stundenplan einsehen

Abwesenheiten einsehen

Startseite > Login > Adminbereich > Abwesenheiten einsehen

Student Name

Infos über die person

- Emaill Adresse
- Semester

Name	Grund	Datum	Actions
			<div>AkzeptierenAblehnen</div>
			<div>AkzeptierenAblehnen</div>
			<div>AkzeptierenAblehnen</div>

Stundenplan einsehen

Startseite > Login > Adminbereich > Stundenplan einsehen

Dozent Name

Einsatzplan generierenAn alle teilen

Infos über die person

- Emaill Adresse
- Spezialität

Stundenplan KW 20

Mo.	Di.	Mi.	Do.	Fr.