

# TEAM 1 - SERVICE 7

PERSONS



# AGENDA

TEAM

SERVICE BESCHREIBUNG

HTTP-ANFRAGEN

NÄCHSTE SCHRITTE



TEAM

TOBIAS KÄRST

JONAS LIEHMANN

NIKLAS SCHUMANN

JUSTIN NOSKE

# SERVICE BESCHREIBUNG



Personen suchen



## Gespeichert



**Raj Raunak**

Dekan Angewandte  
Informatik



**Julieta McLain**

Professorin in  
Datenbanksystemen

Mehr



**Pablo Zulauf**

Professor in IT-  
Sicherheit

## Gefunden



**Carmela Nieves (AI)**

Hochschulrat



**Carlos Holmer (WI)**

Studierendenrat



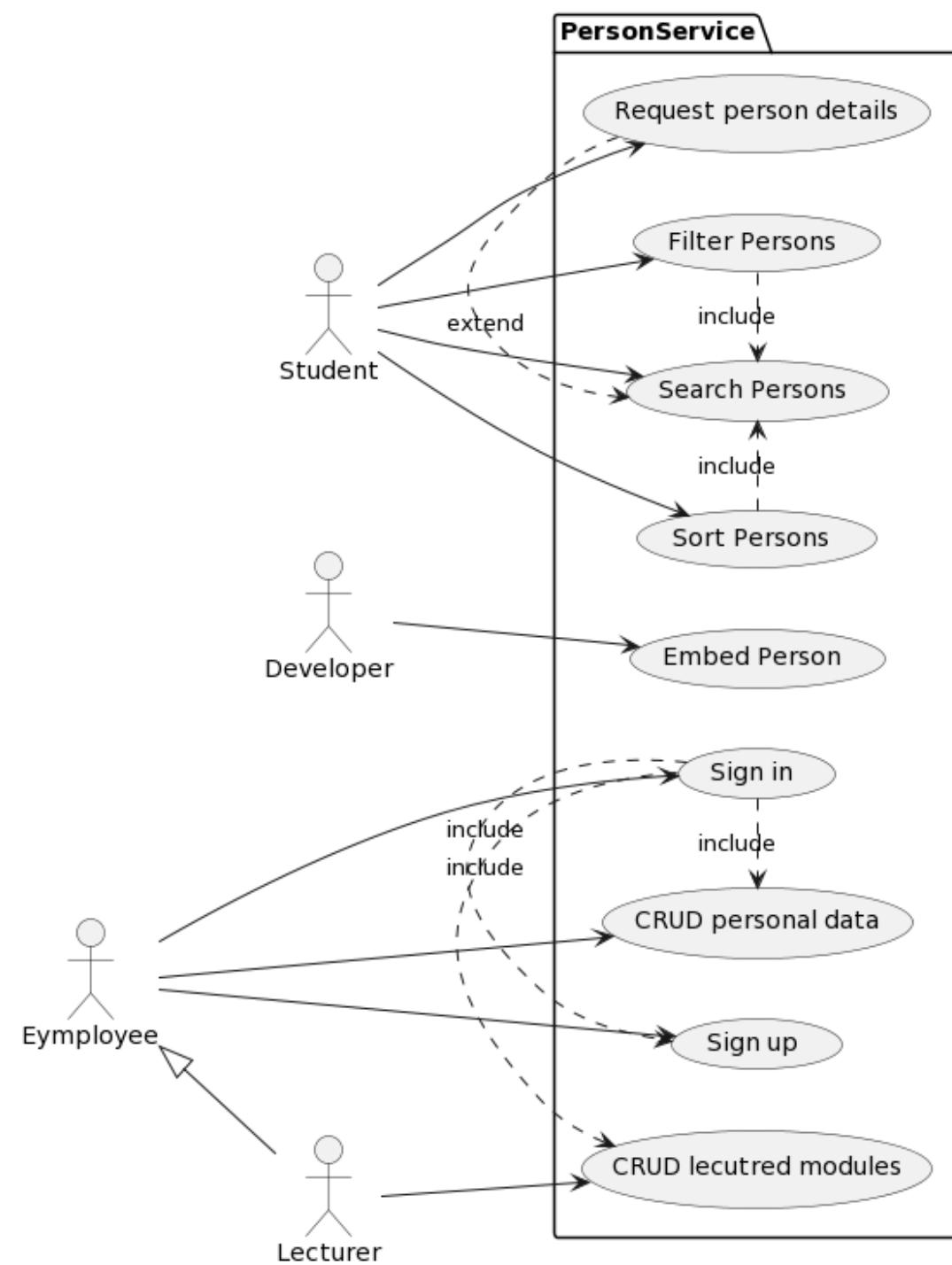
# ÜBERSICHT

Alle Informationen/Funktionen zur Kontaktaufnahme mit den Angestellten der FHE

Intuitive Suche nach Personen nach bestimmten Filter- und Sortierkriterien.

Übersicht über für Studierende relevanten Personen

Erstellen und Verwalten von Nutzeraccounts der Angestellten



# ANFORDERUNGEN

Angestellte können im System ihre (Account-) Daten erstellen und pflegen.

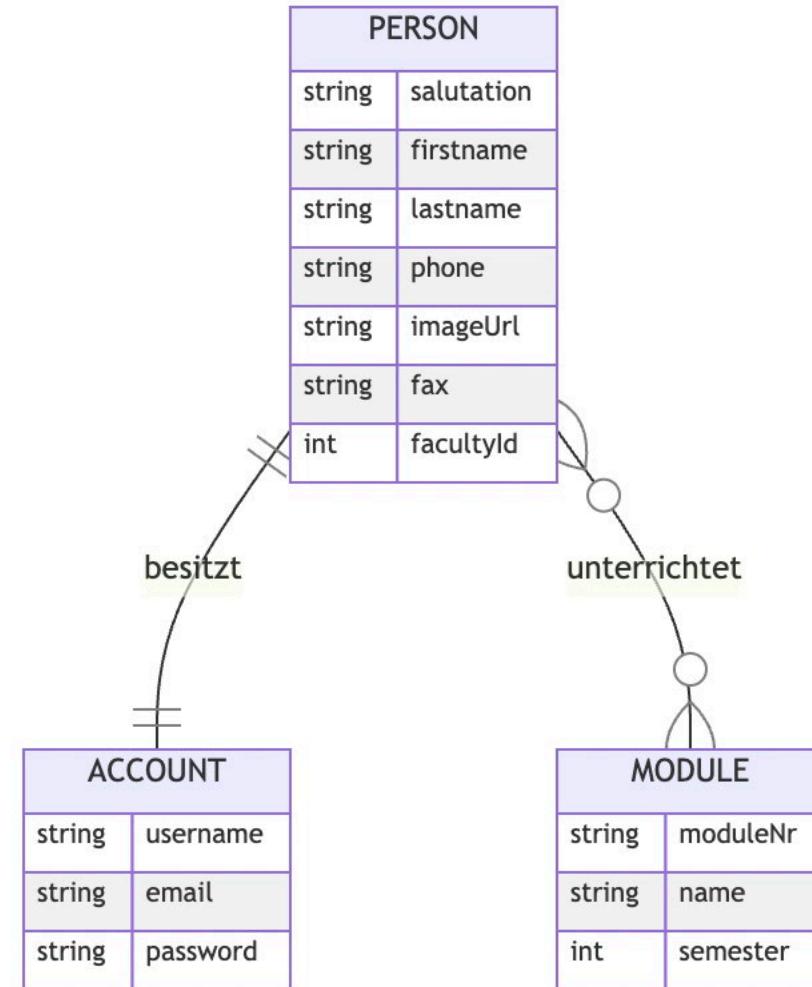
Dozenten können ihre aktuellen Module im System hinterlegen.

Studierende können Angestellte im System nach trivialen Kriterien (Name, Fakultät ...) suchen.

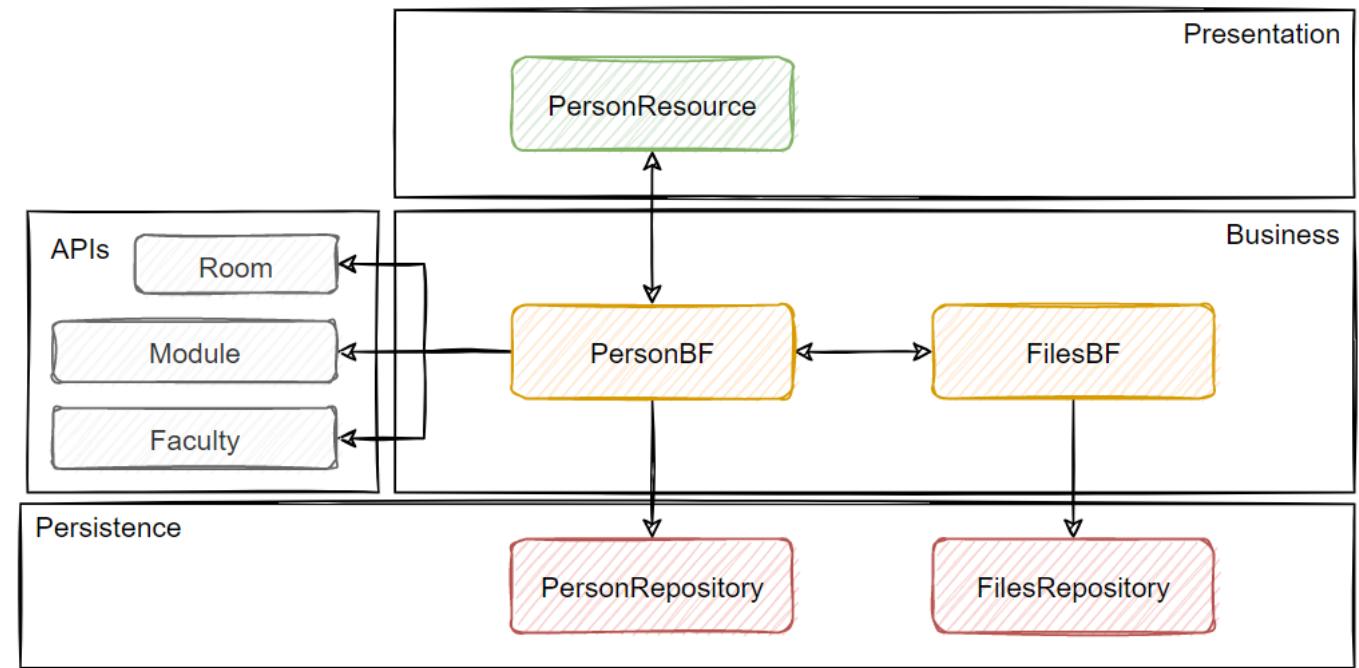
Studierende können Dozenten nach ihrem Semester suchen.

Studierenden wird ein Suchverlauf erstellt und Personen können gespeichert werden.

# ENTITY RELATIONSHIP MODEL



# ARCHITEKTUR



# HTTP-ANFRAGEN



## GET /PERSONS

*Liste aller Personen Filter- und Sortieroptionen.*

- ?facultyId: int [ID der Fakultät]
- ?semester: string [Semester indem Person tätig ist]
- ?query: string [Suchanfrage (Name Person/Fakultät)]
- ?sort\_by: string [Feldname der Sortierreferenz]
- ?order\_by: string [ASC = aufsteigend, DESC = absteigend]
- ?skip: int [Anzahl übersprungener Ergebnisse]
- ?take: int [Anzahl der Ergebnisse]

## GET /PERSONS/{PERSON\_ID}

*Person mit der übergebenen ID.*

## GET /PERSONS/{PERSON\_ID}/MODULE

*Lister aller Module, welche die Person mit der ID in einem bestimmten Fachsemester unterrichtet.*

**?sort\_by:** string [Feldname der Sortierreferenz]

**?order\_by:** string [ASC = aufsteigend, DESC = absteigend]

**?skip:** int [Anzahl übersprungener Ergebnisse]

**?take:** int [Anzahl der Ergebnisse]





## POST /PERSONS

*Legt eine neue Person mit den übergebenen Stammdaten im System an.*



```
{  
  "firstname": string,  
  "lastname": string,  
  "salutation": string,  
  "phone": string,  
  "profile_img": FILE,  
  "fax": string,  
  "facultyId": number,  
  "account": {  
    "email": string,  
    "password": string  
  }  
}
```



## POST /PERSONS/{PERSON\_ID}/MODULE

Weist der/dem Dozent:in mit der passenden ID ein neues Modul zu.



```
{  
  "moduleNr": string,  
  "name": string,  
  "semester": string  
}
```



## PUT /PERSONS/{PERSON\_ID}

*Aktualisiert die Daten einer einzigen Person.*



```
{  
  "firstname": string,  
  "lastname": string,  
  "salutation": string,  
  "phone": string,  
  "profile_img": FILE,  
  "fax": string,  
  "facultyId": number,  
  "account": {  
    "email": string,  
    "password": string  
  }  
}
```





## **DELETE /PERSON/{PERSON\_ID}**

*Entfernt die Person mit der gegebenen ID.*

## **DELETE /PERSON/{PERSON\_ID}/{MODULE\_NR}**

*Entfernt ein unterrichtetes Modul aus der Liste der gegebenen Person.*



# NÄCHSTE SCHRITTE



# NÄCHSTE SCHRITTE

- Testabdeckung weiter verbessern
- Dokumentation finalisieren





VIELEN DANK!