

A decorative graphic on the left side of the slide, consisting of a network of thin, light-blue lines and small circles, resembling a circuit board or a stylized tree structure.

SERVICE-PRÄSENTATION

TEAM 1, SERVICE 7 (PERSONS)

AGENDA

- Teammitglieder
- Service
- Product Backlog
- ER-Modell
 - Person
 - Person Controller
 - Sorting- & Search-Modul
- Lessons Learned
- Weiteres Vorgehen

TEAMMITGLIEDER

- Jonas Liehmann (DevLead)
- Tobias Kärst (PO)
- Niklas Schumann
- Justin Noske

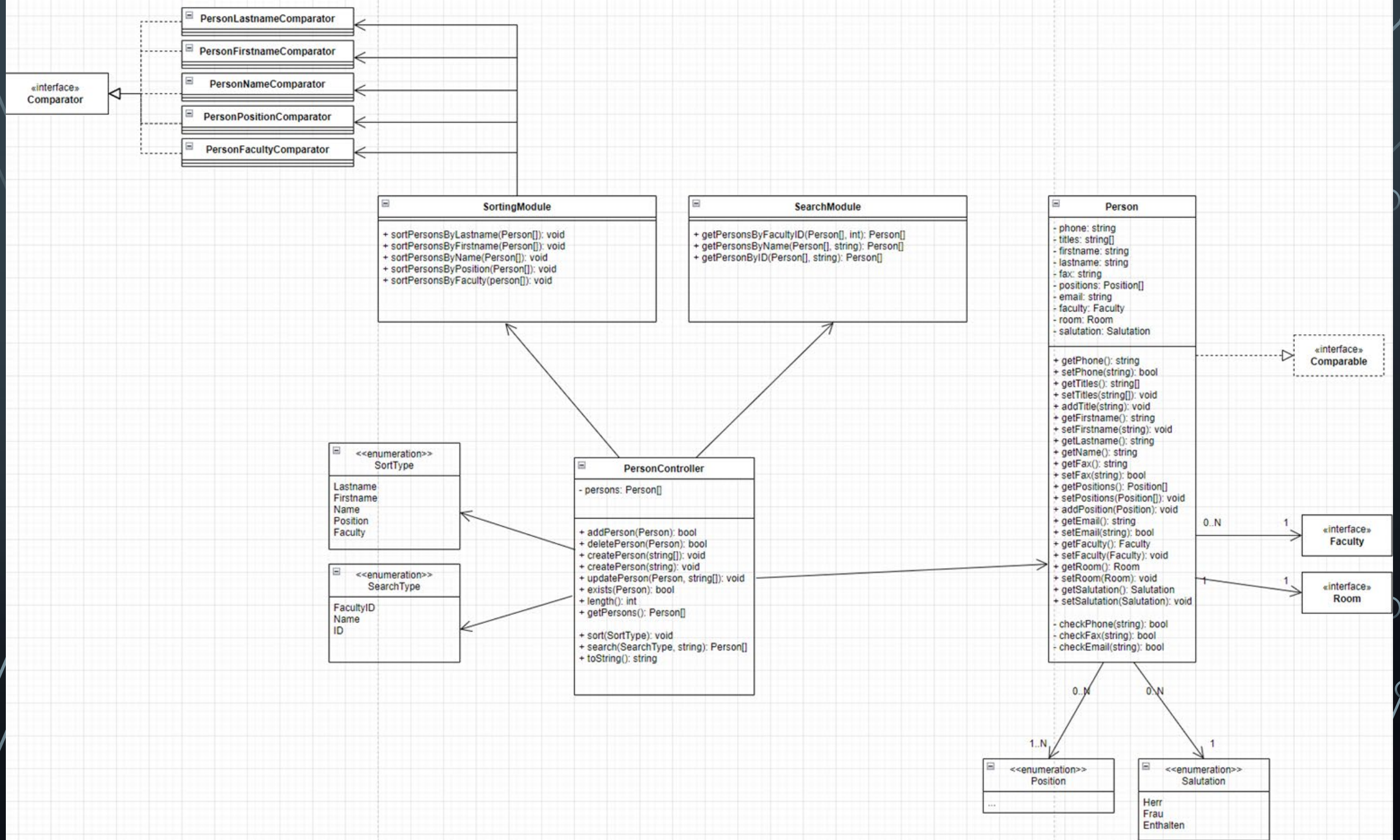
SERVICE

- Beinhaltet alle Informationen und Funktionen zur Kontaktaufnahme mit den Lehrenden an der FHE
- Ermöglicht eine intuitive Suche nach Personen nach bestimmten Suchkriterien
- Bietet eine Übersicht über die für Studierende relevanten Personen

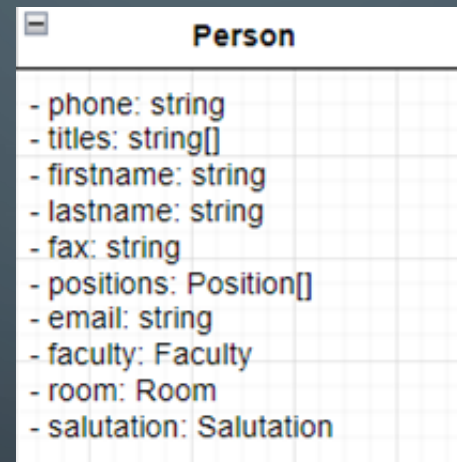
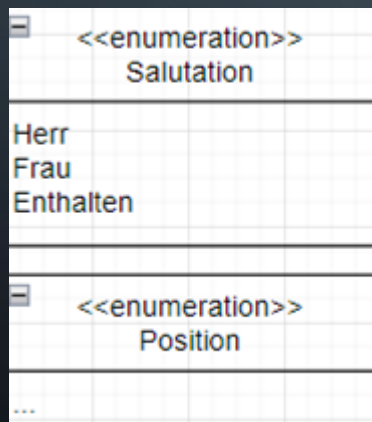
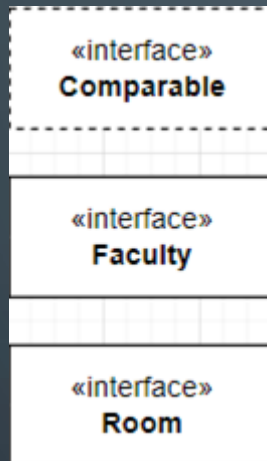


PRODUCT BACKLOG

- Lehrende sollen in der Lage sein, ihre Daten im System zu erstellen
 - Lehrende sollen in der Lage sein, ihre Daten zu verändern
 - Lehrende sollen in der Lage sein, ihre Daten zu löschen
-
- Studierende sollen in der Lage sein, Lehrende nach Namen zu suchen
 - Die Namenssuche, soll keinen direkten Stringvergleich durchführen, sondern nach Namen suchen, die der Eingabe ähnlich sehen
 - Studierende sollen in der Lage sein, die Lehrenden nach der Fakultät filtern zu können
 - Studierende können die Personenliste nach folgenden Kriterien sortieren: Vorname, Nachname, Name, Position, Fakultät
 - Für die Studierenden soll ein Suchverlauf erstellt werden



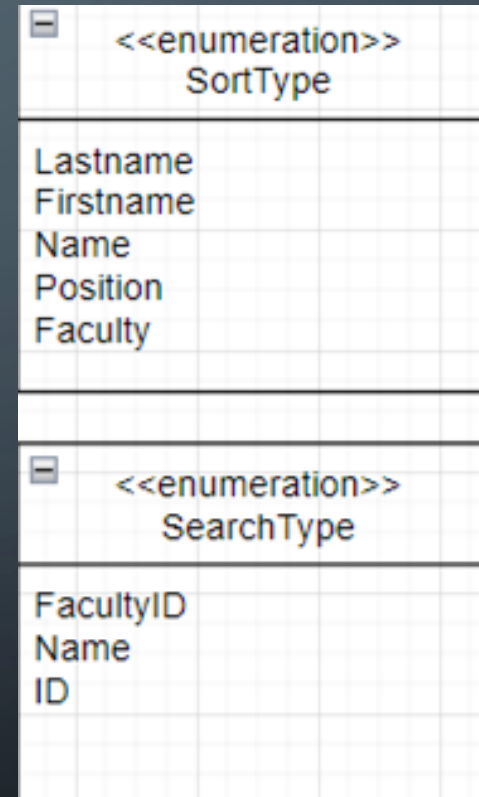
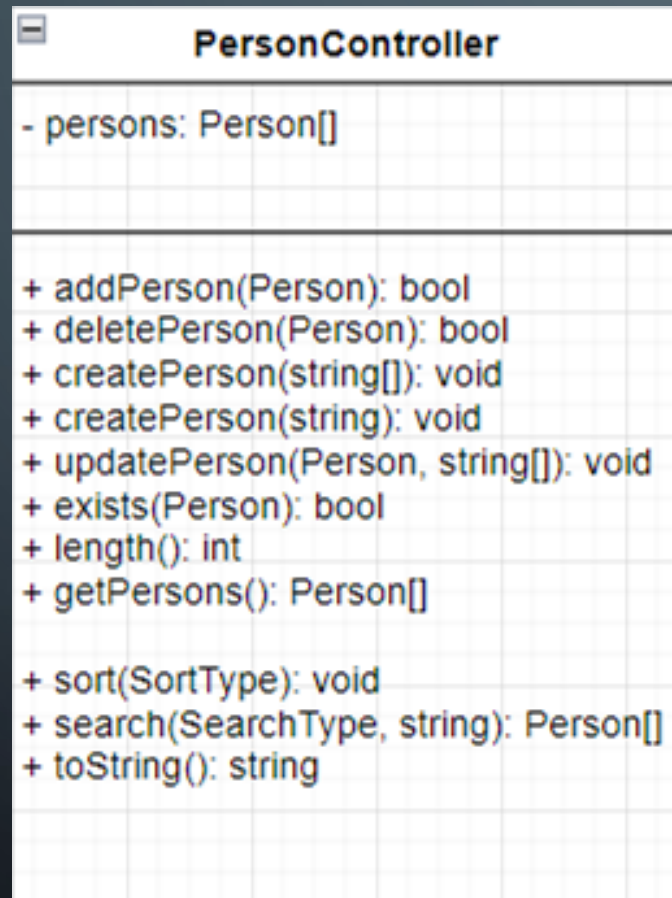
KLASSE PERSON



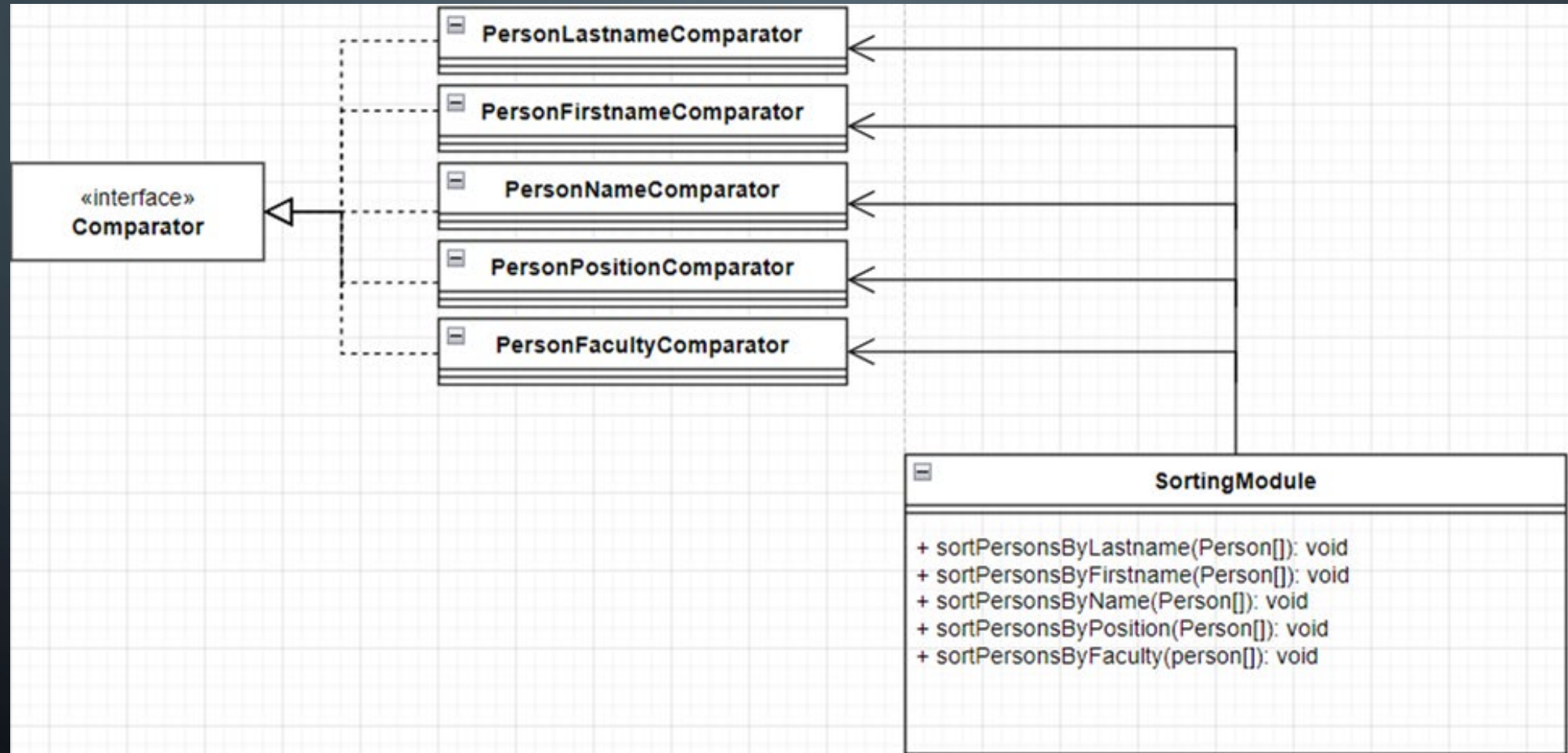
```
+ getPhone(): string
+ setPhone(string): bool
+ getTitles(): string[]
+ setTitles(string[]): void
+ addTitle(string): void
+ getFirstname(): string
+ setFirstname(string): void
+ getLastname(): string
+ getName(): string
+ getFax(): string
+ setFax(string): bool
+ getPositions(): Position[]
+ setPositions(Position[]): void
+ addPosition(Position): void
+ getEmail(): string
+ setEmail(string): bool
+ getFaculty(): Faculty
+ setFaculty(Faculty): void
+ getRoom(): Room
+ setRoom(Room): void
+ getSalutation(): Salutation
+ setSalutation(Salutation): void

- checkPhone(string): bool
- checkFax(string): bool
- checkEmail(string): bool
```

PERSON CONTROLLER



SORTING MODULE



SearchModule	
+ getPersonsByFacultyID(Person[], int): Person[]	
+ getPersonsByName(Person[], string): Person[]	
+ getPersonByID(Person[], string): Person[]	

SearchModule	
+ getPersonsByFacultyID(Person[], int): Person[]	
+ getPersonsByName(Person[], string): Person[]	
+ getPersonByID(Person[], string): Person[]	

SearchModule	
+ getPersonsByFacultyID(Person[], int): Person[]	
+ getPersonsByName(Person[], string): Person[]	
+ getPersonByID(Person[], string): Person[]	

LESSONS LEARNED

- Abkapselung in Module besonders für Teamaufteilung und Tests sinnvoll
- Testgetriebener Ansatz: Architektur wird sofort testbar/sauber implementiert und es entsteht kein ungetesteter Code

WEITERES VORGEHEN

- UML Modell finalisieren und verfeinern
- Alle Interfaces implementieren und an andere Services schicken
- Unit-Tests definieren und implementieren (Test driven development)
- Klassen implementieren

The background is a dark blue gradient. In the corners, there are decorative white line art elements resembling circuit boards or neural networks, with lines and small circles.

VIELEN DANK FÜR IHRE AUFMERKSAMKEIT