

AvailabilityMatrix

Johannes Karrer & Elias Walder

14. Juli 2024

1 Problem

Es wird eine geeignete Datenstruktur benötigt, um die verfügbaren Zeiten der einzelnen Räume abzubilden und gleichzeitig das Eintragen von CourseSessions zu ermöglichen.

Die Datenstruktur muss darüber hinaus effizient auf verschiedenen Constraints überprüft werden können, unter anderem:

- Kurse desselben Semsters dürfen nicht am gleichen Tag zur gleichen Zeit stattfinden, damit sie sich für die Studierenden des jeweiligen Semsters nicht überschneiden.
- Kurse, die zeitlich aufgeteilt wurden (z. B. eine 3-stündige Vorlesung in 2 und 1 Stunde), sollten nicht am gleichen Tag stattfinden.
- Kurse mit mehreren Gruppen (PS) sollten nicht alle zur selben Zeit stattfinden, damit die Einteilung für Studierende flexibler möglich ist.

2 Lösungsansatz: AvailabilityMatrix

Eigenschaften:

- $n \times 5$ Matrix
 - Jede Spalte repräsentiert dabei einen Wochentag (S1: MON, S2: DIE, ...)
 - Jede Zeile repräsentiert 15 Minuten des Tages (Z1: $[t_0, t_0+15)$, Z2: $[t_0+15, t_0+30)$, ...)
 - Jeder Eintrag in der AvailabilityMatrix eines Raumes ist eine Referenz auf die CourseSession, die diesem Raum zu einer bestimmten Zeit zugeordnet ist.

	MON	TUE	WED	THU	FRI
08:00 - 08:15 Uhr	C ₁₁	C ₁₂	C ₁₃	C ₁₄	C ₁₅
08:15 - 08:30 Uhr	C ₂₁	C ₂₂	C ₂₃	C ₂₄	C ₂₅
08:30 - 08:45 Uhr	C ₃₁	C ₃₂	C ₃₃	C ₃₄	C ₃₅
08:45 - 09:00 Uhr	C ₄₁	C ₄₂	C ₄₃	C ₄₄	C ₄₅
	■	■	■		
	■	■	■		

- Vorteile:

- Durch die 15-Minuten-Intervalle lassen sich Kurse unterschiedlicher Länge gut darstellen.
- Die Darstellung der Matrix ist sehr intuitiv und ähnelt der Darstellung eines Wochenkalenders
- Der Ansatz ist flexibel und für unsere Anforderungen auch gut skalierbar.