

## Fallstudie Coll@HBRS

V 1.3

c/o Sascha Alda, 2018 - 2021

### Use Case Nr. 1

**System:** Coll@HBRS

**Name:** Jobangebot suchen

**Akteure:** Student

### Vorbedingung:

- Registrierter Student ist im System eingeloggt.
- Student hat von der Startseite des Systems und die Unterseite zur Suche nach Jobangeboten angefordert. Das System zeigt ihm diese an (von nun an bezeichnet als Maske „SucheDialog“).

### Ereignisfluss:

1. Der Student kann innerhalb der Maske „SucheDialog“ zunächst aus zwei verschiedenen Sucharten wählen: die Einfache Suche *oder* die Erweiterte Suche. In der Einfachen Suche kann der Student in einem Suchfeld einen Suchbegriff eingeben. In der Erweiterten Suche können separate Eingaben für die Felder „Arbeitgeber“, „Job-Beschreibung“, „Gehalt“, „Gewünschte Studienrichtung“ gemacht werden. Die *jeweilige* Eingabe wird als ein spezifisches „Suchauftrags-Objekt“ an das System weitergeleitet (durch Betätigung des Button Suche).
2. Das System empfängt den Suchauftrag und sucht in der internen Datenbank nach den passenden Job-Angeboten. Aus den gewonnenen JobAngebot-Objekten werden die für die Suche relevanten Daten (ID, Jobtitel) herauskopiert und als Datensatz (Objekt vom Typ JobTreffer) in einer Ergebnisliste ablegt. Sowohl die Ergebnisliste als auch die JobTreffer sind *temporärer* Natur.
3. Die Ergebnisliste, welche die einzelnen JobTreffer-Objekte aggregiert, wird vom System an den Studenten als Ergebnis der Suche zurückgeliefert.
4. Die Ergebnisse werden dem Studenten auf einer neuen Ergebnisseite angezeigt, die aus diesem Use Case initiiert wird. Die Ergebnisseite enthält die Informationen zu den einzelnen Jobtreffern.
5. Der Student kann in der Ergebnisseite einzelne JobTreffer als *relevant* markieren. Bei jeder Markierung wird der betreffende JobTreffer an das System zurückgesendet.
6. Das System empfängt die zurückgesendeten JobTreffer-Objekte. Das System identifiziert das zugehörige JobAngebot-Objekt und verlinkt es mit dem Studenten-Profil des Studenten. (Das Verlinkung erfolgt durch die Verwendung des internen Supplier Use Case „Verwaltung von Suchanfragen“). *Anmerkung:* Das Studenten-Profil wird durch das Entity Student repräsentiert).

### Nachbedingung:

Die Suchergebnisse wurden vollständig zurückgeliefert.

Falls relevante Job-Treffer identifiziert: Die JobAngebot-Objekte sind mit dem Studenten-Objekt im System persistent verlinkt und gespeichert.

#### **User Story Nr. 1:**

Als Student möchte ich, dass eine *beliebige* Anzahl von relevanten Job-Angeboten, die ich *während* der Suche von Job-Angeboten gefunden habe, in meinem Studenten-Profil verlinkt werden kann, damit ich auch später auf diese wieder zugreifen kann, und ich in Ruhe aus meinen favorisierten Job-Angeboten selektieren kann.

#### *Details:*

Nach Auskunft des Kunden ist umgekehrt jedes Job-Angebot genau einem Studenten(-Profil) zugeordnet. Ist diese Aussage valide? Wenn nein, bitte korrigieren sie die Aussage in ihrem Modell!

#### **Technical Story Nr. 2:**

Als Software-Architekt möchte ich, dass JobTreffer-Objekte (= DTOs) in einer Ergebnisliste wiederverwendet werden, damit der Speicherbedarf der Anwendung reduziert wird.

#### *Details:*

Eine Ergebnisliste kann natürlich mehrere JobTreffer-Objekte (= DTOs) beinhalten. In diesem Modell soll die Beziehung zwischen Ergebnisliste und JobTreffer als Aggregation dargestellt werden. Falls keine JobTreffer gefunden wurden, wird eine leere Ergebnis-Liste zurückgegeben.