

## Kontrakte

### Operation: **signIn(Id, Password) : Authentifizierungsobjekt**

**Beschreibung:** Prüft die Anmeldedaten und gibt dem Nutzer ein Authentifizierungsobjekt

**Vorbedingung:**

- Der Nutzer ist nicht angemeldet
- Der Nutzer verfügt über eine Id und ein Passwort
- Die Id und das Passwort werden mit einem Nutzerprofil assoziiert

**Nachbedingung:**

- Das Ergebnisobjekt wurde zurückgegeben
- Der Server hat Id und Passwort mit einem Nutzerprofil assoziiert
- Der Nutzer hat Zugriff auf die Anwendung entsprechend der Zugriffsrechte des Nutzerprofils

**Ausnahme:** Die Daten existieren nicht im System, Server ist überlastet

**Ergebnis:** Authentifizierungsobjekt

---

### Operation: **signOut()**

**Beschreibung:** Der Nutzer wird von der Anwendung abgemeldet

**Vorbedingung:**

- Der Nutzer ist angemeldet

**Nachbedingung:**

- Der Nutzer hat (ohne erneute Anmeldung) keinen Zugriff auf Anwendung
  - Der Nutzer befindet sich auf der Anmeldeseite
- 

### Operation: **showProfile(Id) : Profile**

**Beschreibung:** Übergibt dem eingeloggten MitarbeiterIn deren Profil  
**Vorbedingung:**

- Es existiert ein Nutzerprofil mit zugehörigen Daten

**Nachbedingung:**

- Das Ergebnisobjekt wurde zurückgegeben
- Der Nutzer wurde auf die Profilseite weitergeleitet
- Die dem Nutzerprofil entsprechenden Daten wurden auf der Profilseite in den entsprechenden Anzeigeflächen hinterlegt

**Ergebnis:** Profil(EmployeeObjekt ohne Passwort)

**Ausnahme:** (unauthorisierter Zugriff) • kein Profil liegt vor

---

## **Operation: updateProfile(Profile)**

(FirstName, LastName, Picture, Rate-Card-Level, Skills, Projects, Id)

**Beschreibung:** Übernimmt die Änderungswünsche des Nutzers an dessen/deren Profil, speichert diese und gibt einen Hinweistext zurück

**Vorbedingung:** Die Daten konnten an das Modell mit dessen Validierungs-Attributen gebunden werden, d.h.:

- Das Bild ist valide (Auflösung und Größe entsprechend dem korrekten Format)
- Die Veränderung erfüllt das Eingabeformat (z.B. Profilname nur aus Buchstaben)
- Es wurde eine "echte" Änderung vollzogen (mindestens ein Detail wurde verändert)

**Nachbedingung:**

- Alle eingegeben Daten wurden übernommen und mit dem (aktualisierten) Nutzerprofil assoziiert
- Falls sich das Bild geändert hat, wurde es hochgeladen und verlinkt

**Ausgabe:** „Ihre Änderungen wurden übernommen“

**Ausnahme:** Eingabe an der Stelle ... nicht erlaubt

---

## **Operation: showAllOffers() : Set(Offer)**

**Beschreibung:** Es werden alle vorhandenen Angebote angezeigt

**Vorbedingung:**

- Es ist mindestens ein Angebot vorhanden
- Der Nutzer hat die entsprechenden Zugriffsrechte, um die Anfrage der Angebotsübersicht anfordern zu können

**Nachbedingung:**

- Das Ergebnisobjekt wurde zurückgegeben
- Der Nutzer wurde auf die Seite der Angebotsübersicht weitergeleitet

**Ergebnis:** Menge an allen vorhanden Angeboten: Set(Offer)

**Ausnahme:** Es existiert kein Angebote

---

## **Operation: createOffer(Title)**

**Beschreibung:** Erstellt ein neues Angebot mit dem Titel „Title“

**Vorbedingung:**

- Es liegt kein Angebot mit diesem Namen vor
- Der Nutzer hat die entsprechenden Zugriffsrechte, um ein Angebot erstellen zu können

**Nachbedingung:**

- Ein Angebot mit dem Titel wurde erstellt
- Dem Nutzer wird eine Benachrichtigung über die erfolgreiche Erstellung mitgeteilt

**Ausnahme:** Angebot mit selben Namen bereits vorhanden; kein Titel

---

## **Operation: deleteOffer(Offer)**

**Beschreibung:** Sucht und löscht übergebenes Angebot

**Vorbedingung:**

- Es existiert ein Angebot mit den Attributen des Objekts

**Nachbedingung:**

- Angebot liegt nicht mehr vor
- dazugehörige Dokumentenkonfiguration wurde gelöscht
- verknüpfte Mitarbeiterprofil-Kopien wurden gelöscht

**Ausgabe:** „Angebot wurde gelöscht“

**Ausnahme:** Das Angebot existiert nicht mehr (in dieser Form)

---

**Operation: showOffer(Title)**

**Beschreibung:** Die Details zum Angebot werden übermittelt **Vorbedingung:**

- Es existiert ein Angebot mit angegebenen Namen

**Nachbedingung:**

- Das Ergebnisobjekt wurde zurückgegeben
- mit Dokumentenkonfiguration
- mit verlinkten Mitarbeitern

**Ergebnis:** Angebot **Ausnahme:** Kein Angebot liegt vor

---

**Operation: updateOffer(Offer)**

**Beschreibung:** Übernimmt die Änderungswünsche für das übergebene Angebot

**Vorbedingung:**

- Die Daten wurden an das Modell mit dessen Validierung Notizen gebunden
- Es existiert ein Angebot mit angegebenem Namen

**Nachbedingung:**

- Alle eingegeben Daten wurden übernommen

**Ausgabe:** „Ihre Änderungen wurden übernommen“

**Ausnahme:** Das Angebot existiert nicht mehr • kein Titel

---

## Operation: showSearchResults(query) : Set(Employee)

**Beschreibung:** Berechnet anhand der übergebenen Parameter mögliche Profilvorschläge, die den Kriterien auch in Gewichtung entsprechen

**Vorbedingung:**

**Nachbedingung:**

- Das Ergebnisobjekt wurde zurückgegeben

**Ergebnis:** Menge an Profilen **Ausgabe:** „Leider konnte niemand mit den geforderten Kriterien gefunden werden, sollen die Nächstbesten angezeigt werden?“

**Ausnahme:** „Leere Suche, Möchten Sie alle Mitarbeiter anziehen?“

---

## Operation: Add(Offer, Employee)

**Beschreibung:** Speichert eine Kopie der Daten des Mitarbeiters im Angebot, die auch die Originaldaten referenziert

**Vorbedingung:**

- Das Angebot und der Mitarbeiter existieren noch

**Nachbedingung:**

- Die Mitarbeiterliste des Angebots wurde um den Mitarbeiter,
- mit all seine Profildaten erweitert.

**Ausgabe:** „Der Mitarbeiter <e.Name> wurde zum <O.Name> hinzugefügt“

**Ausnahme:** Angebot und Mitarbeiter konnten nicht mehr gefunden werden

---

## Operation: Remove(Offer, Employee)

**Beschreibung:** Löscht den Mitarbeiters aus dem Angebot

**Vorbedingung:**

- Der Mitarbeiter ist Teil des Angebots

**Nachbedingung:**

- Der Mitarbeiter ist nicht mehr in der Mitarbeiterliste des Angebots.

**Ausgabe:** „Der Mitarbeiter <e.Name> wurde entfernt“

**Ausnahme:** Die Assoziation existiert schon nicht mehr

---

## **Operation: updateDocumentConfig(Offer)**

**Beschreibung:** Ändert/erstellt für das Angebot eine Konfigurations-Optionen. Der Nutzer entscheidet durch "die Auswahl von Elementen" was im Dokument angezeigt wird und legt deren Reihenfolge fest

**Vorbedingung:**

- Die Daten wurden an das Modell mit dessen Validierung Notizen gebunden
- Angebot existiert

**Nachbedingung:**

- es wurde eine Konfiguration erstellt/geändert und
  - diese im Angebot gespeichert
- 

## **Operation: generateDocument(Offer): Document**

**Beschreibung:** Erstellt anhand des Angebots und der Konfiguration eine Word Datei

**Vorbedingung:**

- das Angebot, Config existiert

**Nachbedingung:**

- es wurde ein Dokument generiert **Ergebnis:** Word Dokument
- 
- 
- 

## **Operation: showAllProjects() : Set(Project)**

**Beschreibung:** es werden alle vorhandenen Projekte angezeigt

**Vorbedingung:**

- Nutzer hat Admin/Vertriebsrechte, es ist mindestens ein Projekt vorhanden

**Nachbedingung:**

- keine Änderungen

**Ergebnis:** Menge an Projekten

**Ausnahme:** kein Projekt liegt vor

---

## **Operation: createProject(Title)**

**Beschreibung:** Erstellt ein neues Projekt mit dem Namen „Title“ und den eingegebenen Daten

**Vorbedingung:**

- Es liegt kein Projekt mit diesem Namen vor
- es wurde ein Titel eingegeben

**Nachbedingung:**

- Ein Projekt mit übergebenem Titel und restlichen Daten ist in der DB gespeichert

**Ausnahme:** Projekt mit selben Namen bereits vorhanden • kein Titel

---

## **Operation: deleteProject(Project) (optional)**

**Beschreibung:** sucht und löscht übergebenes Projekt und löscht aus jedem Profil den Projektnamen

**Vorbedingung:**

- Nutzer hat Vertriebs-/Adminrechte
- es existiert ein Projekt mit angegebenem Namen

**Nachbedingung:**

- Projekt liegt nicht mehr vor

**Ausgabe:** Hinweistext „Projekt wurde gelöscht“

**Ausnahme:** kein Projekt liegt vor • Fehler beim Löschen auf Datenbank

---

## **Operation: showProject(Titel) : Project**

**Beschreibung:** einem Nutzer mit Vertriebs-/Adminrechten wird ein von ihm mit Projektname angesprochenes Projekt angezeigt

**Vorbedingung:**

- Nutzer hat Vertriebs-/Adminrechte
- es existiert ein Projekt mit angegebenen Namen

**Nachbedingung:**

- Keine Änderungen wurden gemacht

**Ergebnis:** Projekt **Ausnahme:** kein Projekt liegt vor

---

## **Operation: updateProject(Project)**

**Beschreibung:** einem Nutzer hat Änderungswünsche an ein Projekt übergeben, das System speichert diese und gibt einen Hinweistext zurück

**Vorbedingung:**

- Nutzer hat Rechte
- Die Daten wurden an das Modell mit dessen Validierung Notizen gebunden
- es existiert ein Projekt mit angegebenem Namen

**Nachbedingung:**

- Die Änderungen wurden aufgenommen und gespeichert.

**Ergebnis:** Projekt

**Ausgabe:** Hinweistext „Ihre Änderungen wurden übernommen“

**Ausnahme:** kein Projekt liegt vor • Änderung nicht erlaubt (leeres Namensfeld etc)



---

---

---

## **Operation: updateBasicDataSet(JSON)**

**Beschreibung:** Der Nutzer verändert mit Hilfe einer JSON Datei die Datenbasis des Systems

**Vorbedingung:**

- Der Nutzer hält eine ältere/modifizierte datenbasis.json auf seinem Rechner
- Der Aufbau dieser widerspricht nicht den Regeln des Systems.

**Nachbedingung:**

- Das System hat die Datenbank entsprechend aktualisiert.
- Falls Skills gelöscht wurden, wurden
  - diese auch bei allen Mitarbeitern
  - und allen Angeboten gelöscht

**Ausgabe:** „Ihre Änderungen wurden übernommen“

**Ausnahme:** JSON konnte nicht deserialisiert werden