

Technische Dokumentation

Jinba

vorgelegt von

Max Riedel

Tony Nutzmann

Mathias Enderlein



Technische Hochschule Brandenburg

Fachbereich Informatik

Studiengang Informatik

Betreuer

Ilonka Wolpert M.Sc.

Versionshistorie

Version	Datum	Autor(en)	Änderungen
0.1	28.6.2023	Max Riedel	Erstellung

Inhaltsverzeichnis

1	Einführung	3
2	Anwendungsfälle	4
2.1	Aktoren	4
2.2	Anwendungsfälle	4
3	Technische Umsetzung	6
3.1	Technologien	6
3.2	Architektur	6
3.3	Entity Relationship Diagramm	6
3.4	Sequenzdiagramme	8
3.4.1	Jobangebot erstellen	9
3.4.2	Bewerben	10
3.4.3	Bewerbung annehmen	10

1 Einführung

Das Projekt Jinba, ist eine Jobplattform, die es Jobsuchenden und Unternehmen erleichtern soll zusammen zu finden und eine erste Kommunikationsaufnahme zu erreichen. Um das zu erreichen, soll die Vermittlung vorrangig über die Fähigkeiten (Skills) der Jobsuchenden und geforderten Fähigkeiten in den Jobangeboten eines Unternehmens erfolgen.

Dabei ist wichtig anzumerken, dass die Plattform keinen kompletten Bewerbungsprozess abbilden soll, sondern nur die erste Kontaktaufnahme zwischen Unternehmen und Jobsuchenden.

2 Anwendungsfälle

2.1 Aktoren

Aktor	Beschreibung
Jobsuchender	Ein Jobsuchender ist eine Person, die auf der Suche nach einem Job ist.
Unternehmen	Ein Unternehmen ist eine Organisation, die Jobs anbietet.

Tabelle 2: Liste der Aktoren

2.2 Anwendungsfälle

Usecase	Aktor	Beschreibung	Done
Registrierung	Jobsuchender, Unternehmen	Ein Benutzer kann sich auf der Plattform registrieren.	Ja
Anmeldung	Jobsuchender, Unternehmen	Ein Benutzer kann sich auf der Plattform anmelden.	Ja
Profil bearbeiten	Jobsuchender, Unternehmen	Ein Benutzer kann sein Profil bearbeiten.	Ja
Profil anzeigen	Jobsuchender, Unternehmen	Ein Benutzer kann sein Profil anzeigen.	Ja
Nach Jobs suchen	Jobsuchender	Ein Jobsuchender kann nach Jobs suchen.	Ja
Jobs vorgeschlagen bekommen	Jobsuchender	Ein Jobsuchender kann Jobs vorgeschlagen bekommen.	Ja
Jobangebot erstellen	Unternehmen	Ein Unternehmen kann ein Jobangebot erstellen.	Ja
Jobangebot bearbeiten	Unternehmen	Ein Unternehmen kann ein Jobangebot bearbeiten.	Ja
Jobangebot anzeigen	Jobsuchender, Unternehmen	Ein Benutzer kann ein Jobangebot anzeigen.	Ja

Usecase	Aktor	Beschreibung	Done
Bewerben	Jobsuchender	Ein Jobsuchender kann sich auf ein Jobangebot bewerben.	Ja
Bewerbung ansehen	Unternehmen	Ein Unternehmen kann sich die Bewerbungen auf ein Jobangebot ansehen.	Ja
Bewerbung annehmen	Unternehmen	Ein Unternehmen kann eine Bewerbung auf ein Jobangebot annehmen.	Ja
Bewerbung ablehnen	Unternehmen	Ein Unternehmen kann eine Bewerbung auf ein Jobangebot ablehnen.	Ja

Tabelle 3: Anwendungsfälle

3 Technische Umsetzung

3.1 Technologien

- Datenbank: MySQL 8
- Programmiersprache: Java 17
- Framework: Spring Boot 3.0.6
- Template Engine: Thymeleaf
- Authentifizierung: Sessions mit Spring Security
- Frontend: HTML, CSS, JavaScript
- CSS-Framework: Coreui 4
- Build-Tool: Maven

3.2 Architektur

Die Abbildung 1 zeigt ansatzweise unser Architekturdiagramm. Wir haben uns für eine 3 bzw. 5 Schichten Architektur entschieden. Die 3 zentralen Schichten bestehen aus:

- Präsentationlayer mit den Controllern
- Businesslayer mit den Services
- Data Access Layer mit den Repositories

Darüber gibt es dann noch zum einen die View Schicht, welche für das Rendern der HTML-Seiten zuständig ist. Und zum anderen die Persistenzschicht, welche bei uns eine MySQL Datenbank ist.

3.3 Entity Relationship Diagramm

Im folgenden ist das ER-Diagramm zu sehen. Es zeigt die Beziehungen zwischen den einzelnen Entitäten. Als zentralen User existiert ein 'AppUser', der sowohl ein Jobsuchender als auch ein Unternehmen sein kann. Das wird durch seine Rolle bestimmt.

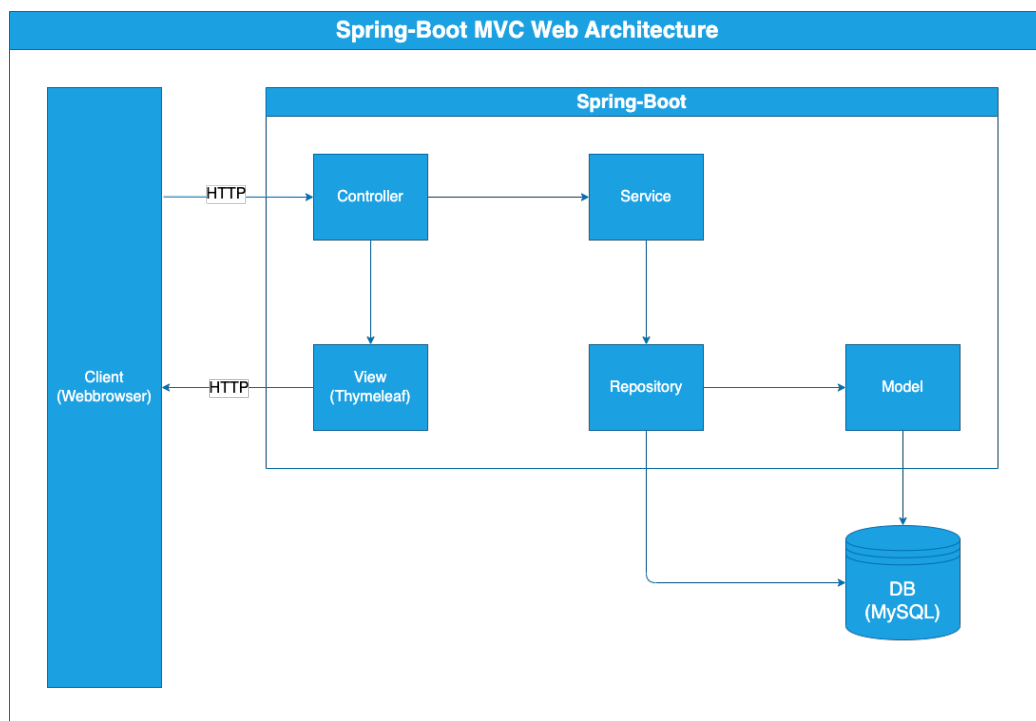


Abbildung 1: Architektordiagramm

Ein Jobsuchender kann sich auf ein Jobangebot ('JobOffer') bewerben ('JobApplication'). Zudem kann er seine Fähigkeiten ('Skill') angeben. Das wird mittels einer Verknüpfungstabelle ('AppUserSkill') realisiert, die zudem das Level des Skills speichert.

Ein Unternehmen wird von einem AppUser administriert. Dieser kann dann für das Unternehmen ein Jobangebot erstellen ('JobOffer'). Dabei kann es Skills ('Skill') angeben, die ein Jobsuchender haben sollte. Auch hier wird das Level des Skills gespeichert ('JobOfferSkill').

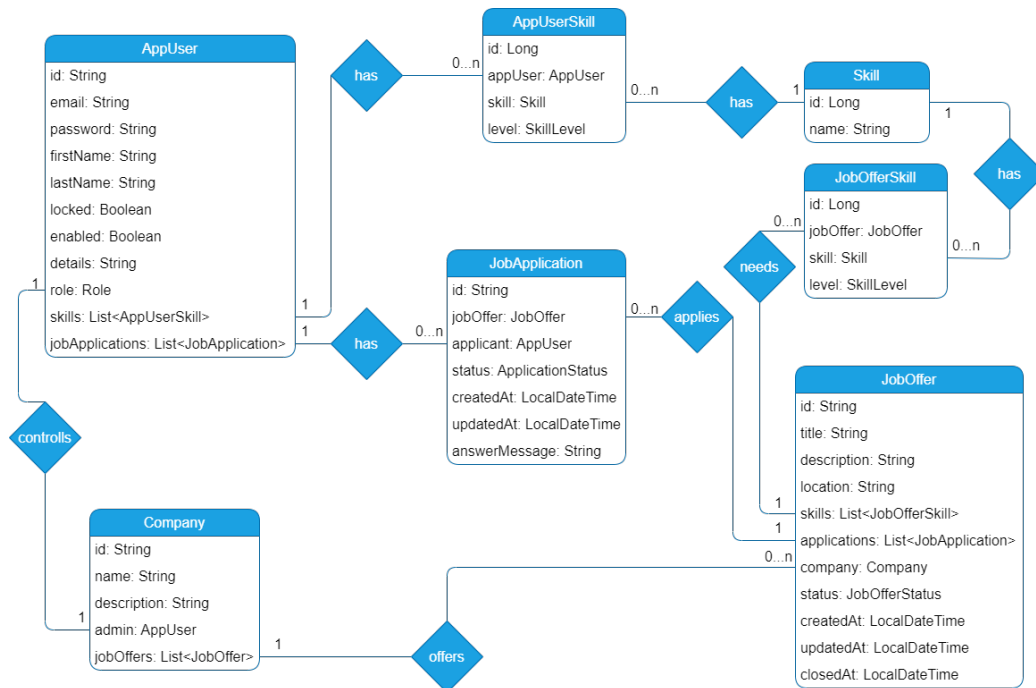


Abbildung 2: Entity Relationship Diagramm

3.4 Sequenzdiagramme

Im Folgenden sind Sequenzdiagramme für 3 wichtige Anwendungsfälle beispielhaft zu sehen. Dabei wird die Kommunikation zwischen den verschiedenen Schichten der Anwendung dargestellt.

3.4.1 Jobangebot erstellen

Die Abbildung 3 zeigt die Kommunikation, wenn ein Unternehmen ein Jobangebot erstellt. Dafür muss zuerst die Firma des aktuell angemeldeten Benutzers ermittelt werden. Dann wird ein neues Jobangebot erstellt, mit den im Form übermittelten Daten. Dabei ist zu beachten, dass die Skills, die noch nicht existieren, erstmal erstellt werden müssen und dann dem Jobangebot hinzugefügt werden. Zuletzt wird das Jobangebot in der Datenbank gespeichert.

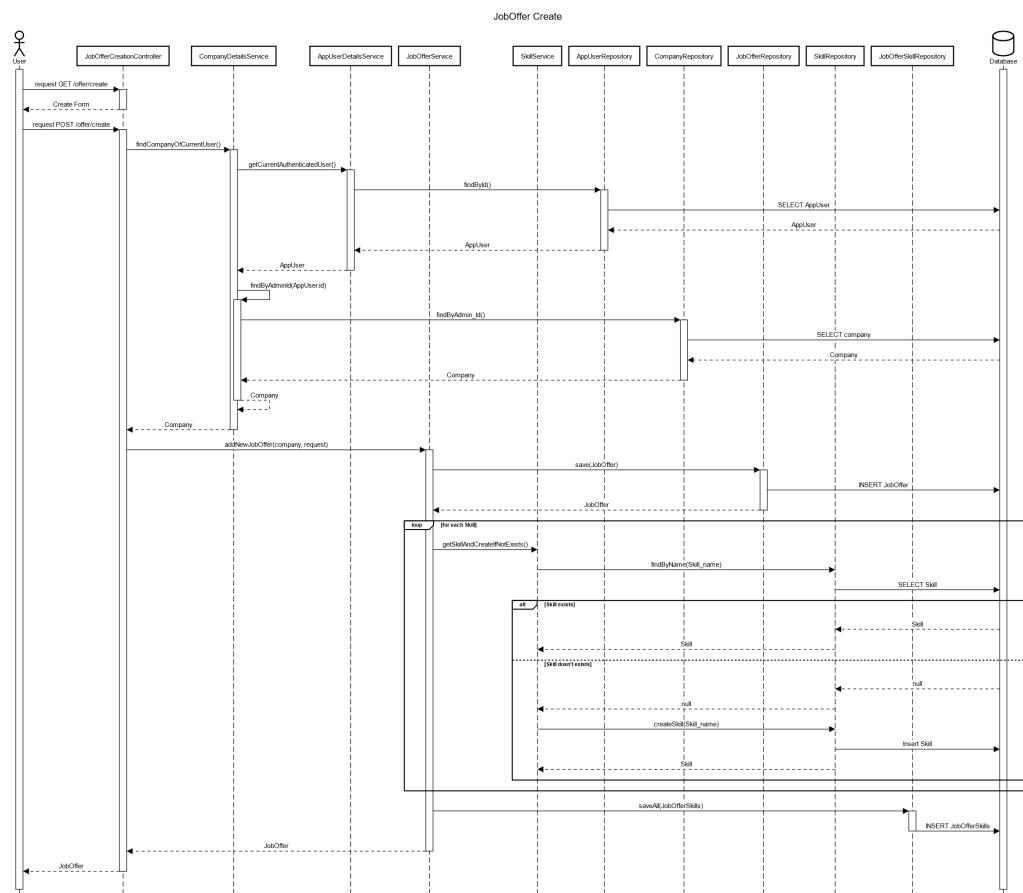


Abbildung 3: Sequenzdiagramm - Jobangebot erstellen

3.4.2 Bewerben

Die Abbildung 4 zeigt die Kommunikation, wenn sich ein Jobsuchender auf ein Jobangebot bewirbt. Dafür muss zuerst der aktuell angemeldete Benutzer ermittelt werden. Dann wird eine neue Bewerbung erstellt und in der Datenbank gespeichert.

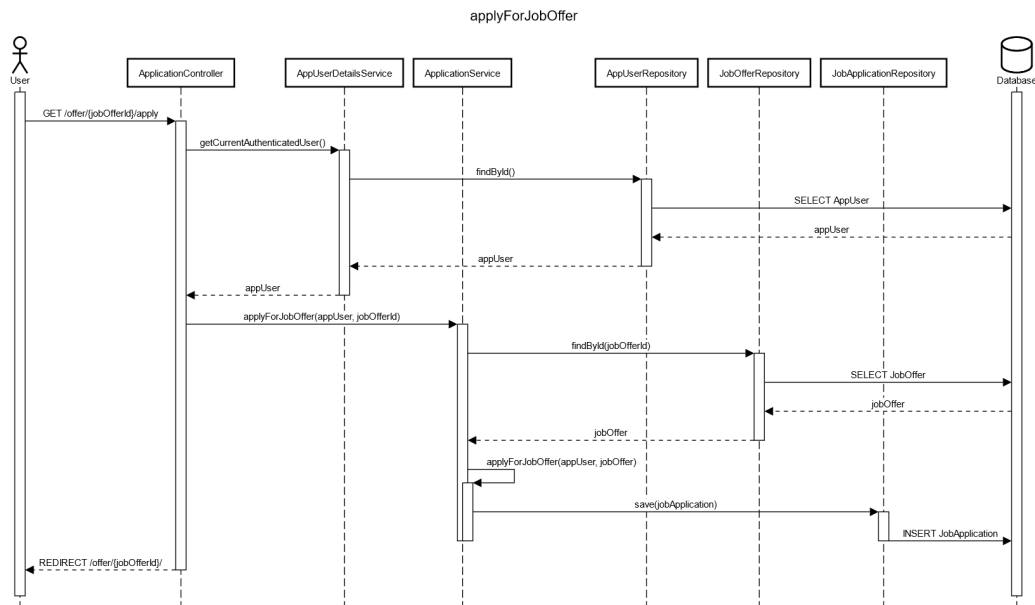


Abbildung 4: Sequenzdiagramm - Bewerben

3.4.3 Bewerbung annehmen

Die Abbildung 5 zeigt die Kommunikation, wenn ein Unternehmen eine Bewerbung auf ein Jobangebot annimmt. Dafür muss zuerst der aktuell angemeldete Benutzer ermittelt werden, um zu überprüfen, dass auch nur der Admin die Bewerbung annimmt. Dann wird die Bewerbung aus der Datenbank geladen, der Status auf 'ACCEPTED' gesetzt und die vom Admin übermittelte Nachricht eingetragen. Zuletzt wird die Bewerbung wieder in der Datenbank gespeichert.

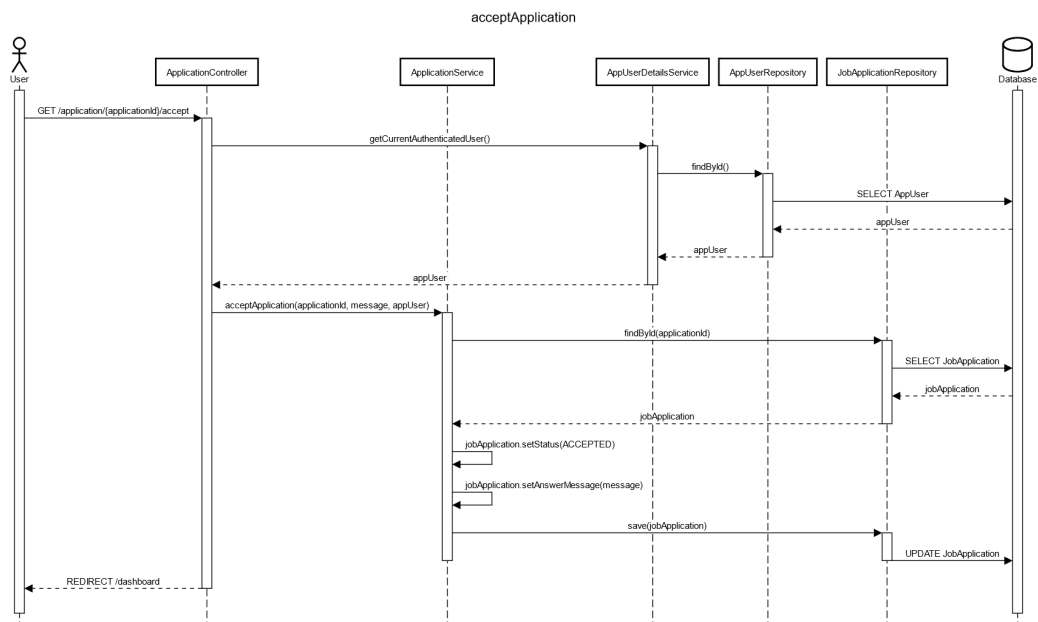


Abbildung 5: Sequenzdiagramm - Bewerbung annehmen