

## Beleg 3

### „adViz“ – Frontend und Backend mit DB

**Abgabe/Präsentation:** Ihre Übung am Fr, 17.07.2020

**User Story:** Als Product Owner des „AdViz“-Projekt würde ich gern eine letzte Version der adViz-Webapp dem Management vorstellen. Diese Version soll die Kontaktdaten aus einer Datenbank beziehen, neue Kontakte in dieser Datenbank speichern und Kontakte in der DB updaten und löschen. In die „adViz“-Webapp kann ich mich als „Admin“ oder als ein „normale/r“ Nutzer/in einloggen. Als „Admin“ darf ich Kontakte hinzufügen, updaten und löschen, als „normale/r“ Nutzer/in darf ich mir nur die public Kontakte des Admin anzeigen lassen.

Acceptance Criteria:

(Voraussetzung: AC von Beleg 2 wurden umgesetzt. Sollte das nicht der Fall sein, dann muss das erst noch erfolgen. Soll heißen: welche Screens wann und wie angezeigt werden sollen, wird hier nicht noch einmal beschrieben.)

- Es wurde eine Datenbank mit zwei Tabellen angelegt: eine User-Tabelle und eine Tabelle für die Kontakte. Die Attribute von User sind userid, password, firstname, lastname, isAdmin (true für „admin“ oder false für „nicht-admin“).
- Es sollen zwei User angelegt werden: einmal „admin“ und eine UserId für „nicht-admin“.
- Die Kontakte, die für Beleg 2 angelegt wurden, müssen jetzt in die entsprechende DB-Tabelle übernommen werden und dürfen nicht mehr im JavaScript hard-coded sein. Alle Kontakte gehören „admin“.
- Wenn die/der PO die URL, <http://localhost:8080/adViz>, im Browser öffnet, wird der Login-Screen im Browser angezeigt.
- Einloggen als „admin“ lässt den Login-Screen verschwinden und zeigt den Main-Screen mit Kontakten und den entsprechenden Markierungen auf der Karte, dem Add-Button, **einem Logout-Button** und **einem Gruß an User, z.B. „Hallo Maxime Muster, eingeloggt als „admin“ an.**
- Einloggen als „nicht-admin“ lässt den Login-Screen verschwinden und zeigt den Main-Screen mit den public Kontakten von admin und den entsprechenden Markierungen auf der Karte an, **einem Logout-Button** und **einem Gruß an User, z.B. „Hallo Max Mustermann, eingeloggt als „nicht-admin“ an.**
- Einloggen mit ungültiger userID/password-Kombination zeigt eine Fehlermeldung an. Der Main-Screen darf nicht angezeigt werden.
- Für beide User soll das Klicken des Logout-Buttons den Main-Screen verschwinden lassen und der Login-Screen soll wieder angezeigt werden, Gruß verschwindet auch.
- Ihr Server stellt zwei Endpoints zur Verfügung:
  - einen für den Login - „/login“: akzeptiert und validiert die userID/password-Kombination. Bei ungültiger userID/password-Kombo, wird der HTTP-Statuscode 401 („Unauthorized“) zurückgeschickt. Bei gültiger userID/password-Kombination: StatusCode 200 und entsprechendes User-POJO als JSON in der Payload.

o einen für Kontakte - „/contacts“. Dieser Endpoint handhabt alle 4 HTTP-Methoden:

HTTP	CRUD-Operation	SQL
POST	Create a new contact	INSERT
GET	Read or retrieve contacts	SELECT
PUT	Update a contact	UPDATE
DELETE	Delete a contact	DELETE

- POST `http://localhost:8080/adViz/contacts` mit Payload `{“firstname“:“Garfield“, “lastname“:“Kater“, “Street“:“Finowstr 10“,..., “lat“:“52.5“, “lng“:“13.4“}`: Server legt diesen neuen Kontakt in der DB an und schickt die neue ID (zum Beispiel 7) an den Client zurück.
- GET `http://localhost:8080/adViz/contacts`: schickt alle Kontakte zurück
- PUT `http://localhost:8080/adViz/contacts/7` mit Payload `{“id“:7, “firstname“:“Garfield“, “lastname“:“Kater“, “Street“:“Neuestr 22“,..., “lat“:“50.5“, “lng“:“10.4“}`: Server führt das entsprechende Update in der DB durch und schickt, falls erfolgreich, den Statuscode 204 zurück.
- DELETE `http://localhost:8080/adViz/contacts/7` ohne Payload: Server löscht den entsprechenden Kontakt in der DB und schickt, falls erfolgreich, den Statuscode 204 zurück

Details, Fragen, etc. werden in der Vorlesung am 03.07.2020 besprochen.