

Beleg 2

„Adviz“ – Frontend mit JavaScript

- **Letzter git-Commit: Sonntag, 19.06.2022 um 23:59 Uhr**
- **Präsentationen finden ONLINE in Zoom statt in KW 27 (Woche vom 04.07.)**
- Mittwochsgruppe präsentiert um 15:45 Uhr am 4.7.
- Montagsgruppe präsentiert um 17:30 Uhr am 4.7.

Um dies zu durchführen zu können, werden von jedem Team folgende Informationen gebraucht:

- **Teamname & gitHub-Repo**
- **Name, Vorname, Matrikelnummer der beiden Teammitglieder**
- **den git commit hash Ihres finalen Commits für diesen Beleg**

Diese Infos bitte am 20.06. per Email an Frau Schüler.

Wichtig: Die finale Version Ihres Codes für Beleg 2 muss in den main-Branch committet werden.

User Story: Als PO-Person des „Adviz“-Projekt möchte ich eine zweite Version unseres Frontend-Prototyps unserem Management vorstellen. „Adviz“ soll nun eine richtige SPA (Single Page Application) werden, allerdings noch ohne Backend & DB.

Acceptance Criteria:

1. Fast alle Anforderungen an die 4 Screens aus Beleg 1 bleiben bestehen, siehe <https://moodle.htw-berlin.de/course/view.php?id=36179#section-3>. **1. Änderung:** Den Owner-Dropdown brauchen wir nicht mehr, d.h., **Owner-Dropdown im „Add“- und im „Update“-Screen muss herausgenommen werden.**
2. Wenn die PO-Person den File-URI .../adviz/index.html im Browser öffnet, soll nur der Login-Screen im Browser angezeigt werden.
3. In Ihre Webapplikation müssen sich diese zwei User-Personen einloggen können:
 - Username: admina, Password: password
 - Username: normalo, Password: password

Damit der Login funktioniert und da die Webanwendung noch kein Backend hat, müssen beide User als Objekte im JavaScript-Code hart-kodiert sein:

```
let admina = {username: "admina", password: "password", role:"admin"};  
let normalo = {username: "normalo", password: "password", role:"normal"};
```

Dieser Code muss bzw. sollte genau so ^^^ übernommen werden. „Admina“ ist die Admin-Person der Webanwendung und hat mehr Rechte als „normalo“. Sie darf eigene Kontakte anlegen, löschen und bearbeiten. Sie darf **alle** Kontakte von normalo sehen, bearbeiten und löschen. Kontakte für normalo darf sie

nicht anlegen (siehe Anforderungsänderung in 1)). Normalo darf eigene Kontakte anlegen, bearbeiten und löschen. Admina darf admina's publike Kontakte sehen, aber **nicht** bearbeiten und **nicht** löschen.

4. Beide User, admina & normalo, müssen jeweils 2 Kontakte haben, einen privaten und einen publike. Auch diese Kontakte müssen erst einmal in Ihrem JavaScript-Quellcode hart-kodiert werden.
5. Als admina einloggen: Login-Kombo (admina/password) eingeben → das Klicken des Login-Buttons lässt den Login-Screen verschwinden und zeigt den Main-Screen mit einem Header an. Der Header zeigt eine persönliche Begrüßungsnachricht an, z.B., „Hallo admina!“ (neu). und optional den Logout-Button. Der Main-Screen besteht aus Kontaktliste (links), Karte (mitte) und 3 bzw 4 Buttons:
 - nur adminas Kontakte werden in der Liste angezeigt
 - adminas Kontakte werden als Markierung auf der Karte angezeigt
 - Es werden die Buttons „Show all contacts“, „Show my contacts“, „Add new contact“ angezeigt.
 - Der Logout-Button kann in den neuen Header verschoben werden.
6. Als normalo einloggen: Login-Kombo (normalo/password) eingeben → Das Klicken des Login-Buttons lässt den Login-Screen verschwinden und zeigt den Main-Screen mit einem Header an. Der Header zeigt eine persönliche Begrüßungsnachricht an, z.B., „Hello, normalo!“ (neu) und optional den Logout-Button. Der Main-Screen besteht aus Kontaktliste (links), Karte (mitte) und drei Buttons:
 - nur normalos Kontakte werden in der Liste angezeigt
 - normalos Kontakte werden als Markierung auf der Karte angezeigt
 - Es werden die Buttons „Show all contacts“, „Show my contacts“, „Add new contact“ angezeigt.
 - Der Logout-Button kann in den neuen Header verschoben werden.
7. Versucht sich jemand mit einer inkorrekten username/passwort-Kombo einzuloggen, muss diese User-Person abgewiesen werden mit, z.b., alert-Box, aber der Login-Screen verschwindet nicht.
8. Als admina kann ich:
 - „Show all contacts“-Button klicken → es werden alle Kontakte (4 Stück) in der Liste und auf der Karte angezeigt
 - „Show my contacts“-Button klicken → es werden nur adminas Kontakte (public & private) angezeigt
 - Auf einen Kontakt in der Liste klicken → es öffnet sich der „Update & Delete contact“-Screen, die Inputfelder sind mit den Daten des Kontaktes belegt. Als admina darf ich alle Kontakte updaten und löschen. Die „Update“ und „Delete“-Buttons müssen nicht funktionieren.
 - „Add new contact“-Button klicken → es öffnet sich der „Add Contact“-Screen, admina darf nur Kontakte für sich selbst anlegen. Der „Add“-Button muss funktionieren. Siehe 10.
9. Als normalo kann ich:
 - „Show all contacts“-Button klicken → es werden normalos und adminas publike Kontakte (3 insgesamt) in der Liste und auf der Karte angezeigt
 - „Show my contacts“-Button klicken → es werden nur normalos Kontakte (public & private) angezeigt
 - auf einen Kontakt in der Liste klicken → es öffnet sich der „Update & delete contact“-Screen, die Inputfelder sind mit den Daten des Kontaktes belegt. Normalo darf nur eigenene Kontakte updaten und löschen. Die „Update“ und „Delete“-Buttons müssen nicht funktionieren.
 - „Add new contact“-Button klicken → es öffnet sich der „Add contact“-Screen, normalo darf nur Kontakte für sich anlegen. Der „Add“-Button muss funktionieren. Siehe 10.
10. Als Admina oder normalo einen neuen Kontakt anlegen: Nachdem Befüllen der Inputfelder und des Klicken des „Save“-Buttons wird der Main-Screen angezeigt. Der neue Kontakt befindet sich links in der Kontaktliste und deren Geokoordinate/Markierung wird auf der Karte angezeigt. (Wir werden in der Übung besprechen, welche WebServices wir zum Ermitteln der Geo-Koordinaten der eingegebenen Adresse nutzen können.) Bitte beachten Sie, dass Ihre Nutzer:innen nicht unbedingt real-existierende Adressen eingeben müssen.... Was bedeutet das?

11. Die „Update“ und „Delete“-Buttons müssen noch nicht funktionieren.
12. Der Logout-Button loggt admina oder normalo aus und zeigt wieder den Login-Screen an.

Happy Coding!