

# Testdrehbuch

Version 1.0

**Team: 3**

**Proseminargruppe: 1**

**Datum: 29.01.2024**

**Hinweis:** Dieses Dokument ist ein Mustertestdrehbuch für die teaminternen Tests des Teamprojekts. Es deckt weitgehend alle erwarteten Funktionalitäten des Projekts ab. Naturgemäß werden sich die Umsetzungen im Detail unterscheiden, so dass die Testfälle ggf. angepasst werden müssen.

Das Testprotokoll stützt sich auf dieses Testdrehbuch. Es bescheinigt die vollständige Abarbeitung der gelisteten Testfälle und verweist ggf. auf gefundene Abweichungen (z.B. in Form der Issue-Nummer in Gitlab).

Das Testdrehbuch ist optional und ist kein verpflichtendes Artefakt für die Abgabe.

## Änderungshistorie

Version	Datum	Status	Autor	Änderung	Abnahme durch (Name)	Abnahmedatum
0.9	20.12.2023	Zum internen Review	Michael Breu	Initialer Inhalt	-	-
1.0	05.01.2023	Freigegeben	Michael Breu	Einarbeiten der Reviewanmerkungen		
1.1	26.01.2024		Florian Harrasser	1.Durchlauf des Drehbuchs		

## Inhaltsverzeichnis

1.Testvorbereitung.....	4
1.1.Testdaten.....	4
1.2.Testeingangskriterien .....	4
2.Testfälle .....	4
2.1.Testfälle Login.....	5
2.2.Testfälle F1 .....	8
2.3.Testfälle F2 .....	9
2.4.Testfälle F3 .....	9
2.5.....F4 .....	11
2.6.....F5 .....	11
2.7.....F6 .....	12
2.8.Premium Abonnement.....	13
2.9.Abrechnung .....	15
2.10.Nutzermanagement .....	16
3.Anhang .....	17
3.1.Glossar .....	17
3.2.Referenzierte Dokument.....	17

## 1. Testvorbereitung

### 1.1. Testdaten

Zu Testbeginn sind folgende Nutzer eingerichtet:

Nutzer / Password	Rolle	Bemerkung
admin / passwd	Administrator	Rolle ADMIN, EMPLOYEE
user2 / passwd	Testnutzer	Rolle USER, EMPLOYEE
testPremium / premiumPasswd	Testnutzer mit Premium Status	Rolle PREMIUM mit gültigen Zahlungsinformationen, Abostart zum 2. Jan. 2024 mit offener Rechnung und Guthaben, EMPLOYEE, Zahlungsinformationen: 3475298374, Balance: 10
testPremiumBad / premiumPasswd	Testnutzer mit Premium Status	Rolle PREMIUM mit offener Rechnung und zu wenig Guthaben, EMPLOYEE Zahlungsinformationen: 7659656959, Balance: 0
User1 / passwd	Manager	Rolle MANAGER, EMPLOYEE

Email für alle User: swaprojektarbeitwetterapp@gmail.com

Initialer Testdatenbestand:

- Nutzerdaten (siehe oben), einschließlich weiterer Daten (Email-Adresse, Zahlungsinformationen, etc.)
- Ereignisse/Buchungen für die letzten Monate: Keine
- Konfigurationen (z.B. SMTP-Server):

```
• spring.mail.host=smtp.gmail.com
  spring.mail.port=587
  spring.mail.username=swaprojektarbeitwetterapp@gmail.com
  spring.mail.password=zjpp nzjl suca taur
  spring.mail.properties.mail.smtp.auth=true
  spring.mail.properties.mail.smtp.starttls.enable=true
  logging.level.org.springframework.mail=DEBUG
```

*Hinweis: Sie sollten diese Daten in Form eines einfach zu importierenden Datenbank-Dumps bereitstellen.*

## 1.2. Testeingangskriterien

Die Integrationstests können gestartet werden, wenn

1. Alle Entwicklertests erfolgreich abgeschlossen und
2. alle JUnit-Tests vollständig und erfolgreich ausgeführt wurden.
3. Ein sinnvoller initialer Testdatenbestand bereitsteht.

## 2. Testfälle

Die hier beschriebenen Testfälle decken die in der Konzeptbeschreibung angeführten Use Cases vollumfänglich ab. Weitere Testfälle wurden zur Überprüfung allgemeiner funktionaler Anforderungen ergänzt. Abweichungen von den erwarteten Ergebniszustände wurden im Rahmen des durchgeführten Tests dokumentiert und entsprechend der nachfolgenden Einstufungen klassifiziert.

- **OK:** Keine Abweichungen gefunden.
- **Kosmetische Abweichungen:** Kleinere Layout Probleme: z.B. Zeilenumbrüche im Text ungeschickt, Texte für Buttons zu lang, usw.
- **Mittlere Abweichungen:** Die Funktionalität ist grundsätzlich vorhanden, kann aber nur eingeschränkt benutzt werden, z.B. einige erwartete Einträge in einer Dropdownliste fehlen, Datenänderungen sind erst nach Schließen und wieder Öffnen eines Dialoges sichtbar, usw.
- **Große Abweichungen:** Die Funktionalität ist nicht benutzbar, z.B. Aktionsbuttons zeigen keine Reaktion, Daten werden nicht korrekt in die Datenbank geschrieben, usw.
- **System unbenutzbar:** Die Durchführung dieses Tests hinterlässt das System in einem unbenutzbaren Zustand, z.B. System stürzt ab. Datenbank wird inkonsistent, Daten werden (ungeplant) gelöscht.

### 2.1. Testfälle Login

TC Auth1.0: Unregistrierter Zugriff	
<b>Use Case:</b>	Authentisierung und Rollen
<b>Ausgangszustand:</b>	Der Nutzer ist ausgeloggt.
<b>Aktion</b>	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Die Startseite erscheint.</li><li>3. Der Nutzer kann Orte suchen.</li><li>4. Der Nutzer kann auf einer Detailseite zu einem Ort die Wetterdaten einsehen</li></ol>
<b>erwarteter Ergebniszustand:</b>	Der Nutzer wird nicht zum Login aufgefordert.

**beobachtete Abweichung**

OK

**TC Auth1.1: Login Nutzer****Use Case:**

Authentisierung und Rollen

**Ausgangszustand:**

Der Nutzer testPremium ist ausgeloggt.

**Aktion**

1. Der Nutzer gibt die Start-URL ein.
2. Es erscheint die Loginseite
3. Der Nutzer gibt seine korrekten Daten ein

**erwarteter Ergebniszustand:**

Der Nutzer ist angemeldet.

Der Nutzer sieht die geschützte Seite und kann die entsprechende Funktionalität nutzen.

**beobachtete Abweichung**

OK

**TC Auth1.2: Login Nutzer (Falsches Passwort)****Use Case:**

Authentisierung und Rollen

**Ausgangszustand:**

Der Nutzer ist ausgeloggt.

**Aktion**

1. Der Nutzer gibt die Start-URL ein.
2. Es erscheint die Loginseite.
3. Der Nutzer gibt ein falsches Passwort ein

**erwarteter Ergebniszustand:**

Der Nutzer ist nicht angemeldet.

Es erscheint die Fehlermeldung „Benutzername und Passwort stimmen nicht überein.“

**beobachtete Abweichung**

Kosmetische Abweichung -&gt; Fehlermeldung erscheint nicht

**TC Auth1.3: Login Nutzer (Falscher Nutzernamen)**

<b>Use Case:</b>	Authentisierung und Rollen
<b>Ausgangszustand:</b>	Der Nutzer ist ausgeloggt.
<b>Aktion</b>	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Es erscheint die Loginseite.</li><li>3. Der Nutzer gibt einen falschen Nutzernamen (und ein beliebiges Passwort) ein</li></ol>
<b>erwarteter Ergebniszustand:</b>	Der Nutzer ist nicht angemeldet.  Es erscheint die Fehlermeldung „Benutzername und Passwort stimmen nicht überein.“
<b>beobachtete Abweichung</b>	Kosmetische Abweichung -> keine Fehlermeldung

**TC Auth1.4: Selbstregistrierung**

<b>Use Case:</b>	Authentisierung und Rollen
<b>Ausgangszustand:</b>	Der Nutzer ist ausgeloggt.
<b>Aktion</b>	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt die Funktion zur Selbstregistrierung.</li><li>3. Der Nutzer gibt die geforderten Daten ein.</li><li>4. Der Nutzer erhält eine Mail mit einem Bestätigungslink</li><li>5. Der Nutzer klickt auf den Link</li><li>6. Der Nutzer gelangt auf eine Bestätigungsseite</li></ol>
<b>erwarteter Ergebniszustand:</b>	Der Nutzer ist angemeldet. Seine Daten sind korrekt hinterlegt
<b>beobachtete Abweichung</b>	OK

**TC Auth1.6: Selbstregistrierung (Edge-Cases)**

Use Case:	Authentisierung und Rollen
Ausgangszustand:	Der Nutzer ist ausgeloggt.
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt die Funktion zur Selbstregistrierung.</li><li>3. Der Nutzer gibt unzulässige Daten ein<ol style="list-style-type: none"><li>a. Z.B. leere Felder</li><li>b. Überlange Felder (z.B. bei Passwort)</li></ol></li></ol>
erwarteter Ergebniszustand:	Es erscheinen geeignete Fehlermeldungen. Der Nutzer hat die Gelegenheit die Daten zu korrigieren. Die Nutzerdaten werden nicht gespeichert.
beobachtete Abweichung	OK

**TC Auth1.7: Passwort vergessen E-Mail auslösen**

Use Case:	
Ausgangszustand:	Der Nutzer ist ausgeloggt.
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt die Funktion „Passwort vergessen“</li><li>3. Der Nutzer gibt den Username an</li><li>4. Der Nutzer erhält eine Mail mit einem Link zum Passwort zurücksetzen</li><li>5. Der Nutzer klickt auf den Link und erhält temporäres Passwort</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Der Nutzer erhält das „Passwort vergessen“ Mail und kann sein Passwort zurücksetzen</li></ul>
beobachtete Abweichung	OK



## 2.2.

## 2.3. Testfälle F1

TC F1.1: Suche nach Ort	
Use Case:	F1
Ausgangszustand:	
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer gibt auf der Suchseite einige Anfangsbuchstaben eines Ortes ein.</li><li>3. Der Nutzer testet verschiedene Eingaben</li><li>4. Der Nutzer wählt einen Treffer</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Je nach Eingabe werden dem Nutzer verschiedene Vorschläge für Treffer gemacht.</li><li>• Es werden korrekte Wetterdaten für den gewählten Treffer angezeigt</li></ul>
beobachtete Abweichung	OK

## 2.4. Testfälle F2

TC F2.1: Detailseite Wetterdaten	
Use Case:	F2
Ausgangszustand:	
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer gibt auf der Suchseite einige Anfangsbuchstaben eines Ortes ein.</li><li>3. Der Nutzer wählt einen Treffer</li><li>4. Der Nutzer sieht die aktuellen Wetterdaten</li></ol>

	<ol style="list-style-type: none"> <li>Der Nutzer wählt auf der Suchseite nochmals denselben Ort aus.</li> <li>Der Nutzer sieht die aktuellen Wetterdaten</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>Es werden aktuelle Wetterdaten für den gewählten Treffer angezeigt</li> <li>Es ist nur ein Zugriff auf die Wetter-API erfolgt.</li> </ul>
beobachtete Abweichung	System unbenutzbar

## 2.5. Testfälle F3

TC F3.1: Favoriten speichern (nicht angemeldeter Nutzer)	
Use Case:	F3
Ausgangszustand:	Der Nutzer ist nicht eingeloggt
Aktion	<ol style="list-style-type: none"> <li>Der Nutzer gibt die Start-URL ein.</li> <li>Der Nutzer gibt auf der Suchseite einige Anfangsbuchstaben eines Ortes ein.</li> <li>Der Nutzer wählt einen Treffer</li> <li>Der Nutzer sieht die aktuellen Wetterdaten</li> <li>Der nicht eingeloggte Nutzer hat keine Option „Als Favoriten speichern“</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>Hat keine Möglichkeit Favoriten zu speichern</li> </ul>
beobachtete Abweichung	OK

TC F3.2: Favoriten speichern (eingeloggt)
---

Use Case:	F3
Ausgangszustand:	Der Nutzer ist eingeloggt. Er hat ggf. keine Anpassungen an den Favoriteneinstellungen vorgenommen.
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer gibt auf der Suchseite einige Anfangsbuchstaben eines Ortes ein.</li><li>3. Der Nutzer geht ins Weather Submenu und wählt „Manage Favourites“</li><li>4. Kann den Ort durch Suchfeld auswählen und speichern</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Der Ort wird als Favorit permanent gespeichert: Nach Aus-/Einloggen ist der Favorit noch vorhanden</li><li>• Auf der Homeseite erscheinen Basiswetterdaten (<i>current: temp, feels_like, weather</i>) zu den gewählten Favoriten.</li></ul>
beobachtete Abweichung	OK

**TC F3.3: Favoriten sortieren**

Use Case:	F3
Ausgangszustand:	Der Nutzer ist eingeloggt.
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt die Einstellung zum Bearbeiten der Favoriten</li><li>3. Der Nutzer ändert die Reihenfolge der Anzeige der Favoriten</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Der Reihenfolge der Favoriten wird permanent gespeichert: Nach Aus-/Einloggen ist die Reihenfolge noch vorhanden.</li></ul>
beobachtete Abweichung	OK

#### TC F3.4: Favoriten löschen

Use Case:	F3
Ausgangszustand:	Der Nutzer ist eingeloggt.
Aktion	<ol style="list-style-type: none"> <li>Der Nutzer gibt die Start-URL ein.</li> <li>Der Nutzer wählt die Einstellung zum Bearbeiten der Favoriten</li> <li>Der Nutzer löscht den Favoriten</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>Der Favorit ist permanent gelöscht.</li> </ul>
beobachtete Abweichung	OK

## 2.6. F4

#### TC F4.1: Favoriteneinstellungen

Use Case:	F4
Ausgangszustand:	Der Premium Nutzer ist eingeloggt und hat einige Favoriten gespeichert
Aktion	<ol style="list-style-type: none"> <li>Der Nutzer gibt die Start-URL ein.</li> <li>Der Nutzer wählt die Option Favoriteneinstellungen aus</li> <li>Der Nutzer kann die auf der Übersichtsseite angezeigten Messwerte von <i>current</i> (<i>sunrise, sunset, temp, feels_like, pressure, humidity, dew_point, clouds, uvi, visibility, wind_speed, wind_deg, rain, snow, weather.icon, weather.main</i>) konfigurieren</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>Die Einstellungen werden permanent gespeichert: Nach Aus-/Einloggen sind die Einstellungen noch vorhanden</li> <li>Auf der Übersichtsseite erscheinen die eingestellten Messwerte zu den gewählten Favoriten.</li> </ul>
beobachtete Abweichung	System unbenutzbar

## 2.7. F5

TC F5.1: Premiumanzeige	
Use Case:	F5
Ausgangszustand:	Der Premium Nutzer ist eingeloggt
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt in der Suchfunktion einen Ort aus</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Als Premium Nutzer sieht er weitere Messwerte:<ul style="list-style-type: none"><li>▪ Messwerte des aktuellen und der nächsten 8 Tage: <i>daily: humidity, pressure, dew_point, clouds, moon_phase</i></li><li>▪ Messwerte aktuell und der nächsten 48 Stunden: <i>current/hourly: humidity, pressure, clouds, visibilty</i></li></ul></li></ul>
beobachtete Abweichung	OK

## 2.8. F6

TC F6.1: Urlaubsplanung	
Use Case:	F6
Ausgangszustand:	Der Premium Nutzer ist eingeloggt
Aktion	<ol style="list-style-type: none"><li>1. Der Nutzer gibt die Start-URL ein.</li><li>2. Der Nutzer wählt in der Suchfunktion einen Ort aus</li></ol>

<p>erwarteter Ergebniszustand:</p>	<ol style="list-style-type: none"> <li>Der Nutzer wählt auf der Detailseite die Funktion „Choose Dates“ und definiert dort einen Zeitraum von maximal 2 Wochen im kommenden Jahr</li> <li>Wählt „Get holiday forecast“</li> </ol>
<p>beobachtete Abweichung</p>	<ul style="list-style-type: none"> <li>Es werden folgende Werte angezeigt: Diagramm: Vorhersage des nächsten Jahres mit den Mittelwerten: <i>temperature (min, max, afternoon, night, evening, morning), precipitation, humidity.</i></li> </ul> <p>OK</p>

### TC F6.2: Urlaubsplanung (Edge Case)

Use Case:	F6
Ausgangszustand:	Der Premium Nutzer ist eingeloggt
Aktion	<ol style="list-style-type: none"> <li>1. Der Nutzer gibt die Start-URL ein.</li> <li>2. Der Nutzer wählt in der Suchfunktion einen Ort aus</li> <li>3. Der Nutzer wählt auf der Detailseite die Funktion „Choose Dates“ und versucht dort einen Zeitraum von mehr 2 Wochen im kommenden Jahr zu definieren</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>• Der Nutzer hat nicht die Möglichkeit mehr als zwei Wochen zu definieren</li> </ul>
beobachtete Abweichung	OK

## 2.9. Premium Abonnement

## TC Premium.1: Premium Abo abschließen (Zahlungsdaten noch nicht hinterlegt)

Use Case:	Premium Abo
-----------	-------------

Ausgangszustand:	Der user2 ist eingeloggt.
Aktion	<ol style="list-style-type: none"> <li>1. Der Nutzer wählt die Funktion“(un)subscribe“ in „Premium Submenu“</li> <li>2. Der Nutzer wird aufgefordert seine Zahlungsdaten bereitzustellen.</li> <li>3. Wählt „provide Paymentinfomation“</li> <li>4. Der Nutzer gibt seine Zahlungsdaten ein.</li> <li>5. Der Nutzer wählt „Subscribe“ im „Premium Submenu“</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>• Der Nutzer hat nun die Rolle PREMIUM.</li> <li>• Der Nutzer kann alle PREMIUM Funktionen nutzen</li> <li>• In der Datenbank ist das Anfangsdatum des Premium-Abos hinterlegt</li> </ul> <p><b>Hinweis:</b> Für die Wiederholbarkeit des Tests, anschließend dem testUser die Rolle PREMIUM wieder entziehen (siehe TC Premium.3).</p>
beobachtete Abweichung	OK

#### TC Premium.2: Premium Abo abschließen (Zahlungsdaten bereits hinterlegt)

Use Case:	Premium Abo
Ausgangszustand:	Der testUser ist eingeloggt.
Aktion	<ol style="list-style-type: none"> <li>1. Der Nutzer wählt die Funktion „Premium Abo“</li> <li>2. Der Nutzer bestätigt den Abschluss des „Premium Abo“</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>• Der Nutzer hat nun die Rolle PREMIUM.</li> <li>• Der Nutzer kann alle PREMIUM Funktionen nutzen</li> <li>• In der Datenbank ist das Anfangsdatum des Premium-Abos hinterlegt</li> </ul>

beobachtete Abweichung	<b>Hinweis:</b> Für die Wiederholbarkeit des Tests, anschließend dem testUser die Rolle PREMIUM wieder entziehen (siehe TC Premium.3)..
	OK

TC Premium.3: Premium Abo beenden	
Use Case:	Premium Abo
Ausgangszustand:	Der premiumUser ist eingeloggt.
Aktion	1. Der Nutzer wählt die Funktion „Unsubscribe“ im „Premium Submenu“
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Dem Nutzer wurde die Rolle PREMIUM entzogen.</li><li>• Der Nutzer kann keine PREMIUM Funktionen mehr nutzen</li><li>• In der Datenbank ist das Abo-Ende hinterlegt in Invoice.</li><li>• Beim nächsten Rechnungslauf wird das Abo anteilig verrechnet.</li></ul> <b>Hinweis:</b> Für die Wiederholbarkeit des Tests, anschließend dem testPremium die Rolle PREMIUM wieder zuordnen (siehe TC Premium.2).
beobachtete Abweichung	OK

## 2.10. Abrechnung

Vorbemerkung: Je nach Umsetzung der Abrechnung können die Testfälle anders ausfallen.

TC Abrechnung.1: Rechnungsprüfung	
Use Case:	Abrechnung
Ausgangszustand:	Der user1 ist eingeloggt.



Aktion	<ol style="list-style-type: none"> <li>1. Der Manager lässt sich den aktuellen Status der Abrechnungen anzeigen in „Manager Submenu“ – „Premium User“.</li> <li>2. In diesem Monat neu erstellte Abos werden nur anteilig verrechnet angezeigt.</li> <li>3. Der Manager hinterlegt für jede Zahlung, ob die Abbuchung erfolgreich oder nicht erfolgreich war.</li> <li>4. Der Manager entzieht dem Nutzer testPremiumBad die Rolle PREMIUM.</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>• Der Nutzer testPremiumBad hat keine PREMIUM-Rolle mehr.</li> <li>• Ein Mail mit dieser Information wird an den Benutzer testPremiumBad versendet</li> </ul> <p><b>Hinweis:</b> Für die Wiederholbarkeit des Tests, anschließend dem testPremiumBad die Rolle PREMIUM wieder zuordnen (siehe TC Premium.2).</p>
beobachtete Abweichung	OK

## 2.11. Nutzermanagement

TC Nutzermanagement.1: Anzeige aller Nutzer	
Use Case:	Nutzermanagement
Ausgangszustand:	Der admin ist eingeloggt.
Aktion	<ol style="list-style-type: none"> <li>1. Der Administrator zeigt sich alle Nutzer an in „Admin Submenu“ – „Manage users“</li> </ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"> <li>• Die Anzeige spiegelt den aktuellen Status wieder</li> </ul>
beobachtete Abweichung	OK

TC Nutzermanagement.2: Ändern von Nutzerdaten
---

Use Case:	Nutzermanagement
Ausgangszustand:	Der Administrator ist eingeloggt.
Aktion	<ol style="list-style-type: none"><li>1. Der Administrator zeigt sich Nutzer an.</li><li>2. Der Administrator sucht nach einem bestimmten Nutzer</li><li>3. Der Administrator ändert Details zu diesem Nutzer</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Die Details sind permanent gespeichert.</li><li>• Wurde die Rolle geändert, wird das in einem Log protokolliert.</li></ul>
beobachtete Abweichung	Große Abweichung

**TC Nutzermanagement.3: Löschen von Nutzern**

Use Case:	Nutzermanagement
Ausgangszustand:	Der Administrator ist eingeloggt.
Aktion	<ol style="list-style-type: none"><li>1. Der Administrator zeigt sich alle Nutzer an.</li><li>2. Der Administrator sucht nach einem bestimmten Nutzer</li><li>3. Der Administrator löscht diesen Nutzer</li></ol>
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>• Der Nutzer ist permanent (zusammen mit seinen weiteren Daten z.B. Favoriten, Zahlungsinformationen, Favoriteinstellungen) gelöscht</li></ul> <p><b>Hinweis:</b> Nach dem Löschen des Nutzers ggf. die Datenbank wieder konsistent herstellen.</p>
beobachtete Abweichung	Große Abweichung

**TC Nutzermanagement.4: Passwort vergessen E-Mail auslösen**

Use Case:	Nutzermanagement
Ausgangszustand:	Der Administrator ist eingeloggt.
Aktion	<ol style="list-style-type: none"><li>6. Der Administrator zeigt sich alle Nutzer an.</li><li>7. Der Administrator sucht nach einem bestimmten Nutzer</li></ol>

erwarteter Ergebniszustand:	8. Der Administrator triggert das „Passwort vergessen“ E-Mail für diesen Nutzer
beobachtete Abweichung	<ul style="list-style-type: none"><li>Der Nutzer erhält das „Passwort vergessen“ Mail und kann sein Passwort zurücksetzen</li></ul>
	OK

## 2.12. Weitere allgemeine Testfälle

TC Usability.1	
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>Alle Seiten haben ein einheitliches Look &amp; Feel</li><li>Nutzerfehler (z.B. falsche Eingaben) werden konsistent rückgemeldet</li><li>Systemfehler (z.B. Datenbankzugriffsfehler) werden dem Nutzer sinnvoll mitgeteilt.</li></ul>
beobachtete Abweichung	OK

TC Performance.1	
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>Die Antwortzeiten sind angemessen.</li><li>Bei längeren Antwortzeiten (z.B. API-Calls) erscheinen ggf. Warteanzeigen</li></ul>
beobachtete Abweichung	OK

TC Security.1	
erwarteter Ergebniszustand:	<ul style="list-style-type: none"><li>Es werden keine sicherheitskritischen Daten (Passwörter, API Token, ...) im Klartext abgelegt, bzw. im GIT persistent gespeichert.</li></ul>
beobachtete Abweichung	OK

## 3. Anhang

### 3.1. [Glossar](#)

Keine Einträge vorhanden

### 3.2. [Referenzierte Dokumente](#)

- Semesterprojekt Aufgabenstellung (aus LMS: Aufgabenstellung\_WS23\_23-12-11.pdf)