

Testdrehbuch und Protokoll

Team: Coffee @Beans/ Nr. 4

Mitglied 1: Natalie Höpperger, 01516180

Mitglied 2: Paul Bernhard Kirchebner, 11829301

Mitglied 3: David Josef Arnold, 11719198

Mitglied 4: Christoph Wittauer, 11723544

Mitglied 5: Christoph Kugler, 01316789

Proseminargruppe: 2

Datum: 28.05.2020

Hinweis: Dieses Dokument versteht sich als Guideline und dient der Strukturierung des Testdrehbuchs und zugehörigen Testprotokolls. Dieses Dokument definiert die Testdurchführung für die Integrationstests (und ist ein Vorschlag für die Abnahmetests). Zielgruppe sind die Verantwortlichen für die Durchführung der Integrationstest. Dieses Dokument kann durch Ergänzung der Kapitel 3 und 4 auch als Testprotokoll verwendet werden.

Änderungshistorie

Version	Datum	Status	Autor	Änderung	Abnahme durch (Name)	Abnahmedatum
0.1	14.4.2020	In Arbeit	Michael Breu	Initiale Beispiele	-	-
1.0	18.06.2020	Vollständig	Coffe @Beans	Vervollständigung des Testdrehbuchs	Coffe @Beans	18.06.2020

Inhaltsverzeichnis

1.	Testvorbereitung	4
1.1.	Testdaten	4
1.2.	Testeingangskriterien	4
2.	Testprotokoll.....	4
3.	Testfälle	5
3.1.	Testfälle Login.....	5
3.2.	Testfälle Einstellungen	7
3.3.	Testfälle Produktivitätsanalyse	7
3.4.	Testfälle Admin.....	8
3.5.	Testfälle Mitarbeiter.....	12
3.6.	Testfälle Teamleiter.....	14
3.7.	Testfälle Abteilungsleiter.....	16
3.8.	Testfälle Raspberry Pi	17
3.9.	Weitere nichtfunktionale Testfälle.....	19
4.	Anhang.....	Fehler! Textmarke nicht definiert.
4.1.	Glossar.....	Fehler! Textmarke nicht definiert.
4.2.	Referenzierte Dokumente	Fehler! Textmarke nicht definiert.

1. Testvorbereitung

1.1. Testdaten

Zu Testbeginn sind folgende Nutzer eingerichtet:

Nutzer / Password	Rolle	Bemerkung
admin/ passwd	Admin	
user64/ passwd	Employee (Mitarbeiter)	Department 1, Team 1, Room 1
user003/ passwd	Department Manager (Abteilungsleiter)	Department 1
user030/ passwd	Team Leader (Teamleiter)	Department 1, Team 1, Room 1

Initialer Testdatenbestand: (Beispiele)

- Nutzerdaten (siehe oben), ggf. einschließlich weiterer Daten (E-Mail-Adressen, etc.)
- Ereignisse/Buchungen für den letzten Monat
- Zugeordnete Devices
- Konfigurationen (z.B. SMTP-Server)

1.2. Testeingangskriterien

Die Integrationstests können gestartet werden, wenn (z.B.)

1. Alle Entwicklertests erfolgreich abgeschlossen
2. Alle JUnit-Tests vollständig und erfolgreich ausgeführt wurden

2. Testprotokoll

Testdatum: 18.06.2020 (wann wurde getestet? ggf. Zeitraum)

Tester: Coffe@Beans (wer hat getestet?)

Getestete Version: Abgabeversion (z.B. GIT Tag)

Testeingangskriterien erfüllt: ja/nein laut ja

(Verweis auf Email der Entwickler, Maventestprotokoll, ...)

Testumgebung: Anwendung in Docker

(z.B. Anwendung lokal auf eigenem Rechner, Datenbank auf Server)

3. 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 (vgl. Kapitel 2, Testprotokoll) 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.

3.1. Testfälle Login

TC Login.1: Login Nutzer	
Use Case:	Login
Ausgangszustand:	Der Nutzer user64 ist ausgeloggt.
Aktion:	<ol style="list-style-type: none"> 1. Der Nutzer gibt die URL "localhost:8080" ein. 2. Die Login-Maske erscheint. 3. Der Nutzer gibt "user64" und sein Nutzerpasswort "passwd" ein und klickt Login.
erwarteter Ergebniszustand:	<p>Der Nutzer ist angemeldet.</p> <p>Der Nutzer sieht die Seite "Produktivitätsanalyse" unter "Dashboard".</p>
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Login.2: Login Nutzer (Falsches Passwort)

Use Case:	Login
Ausgangszustand:	Der Nutzer ist ausgeloggt.
Aktion:	<ol style="list-style-type: none">1. Der Nutzer gibt die URL "localhost:8080" ein.2. Die Login-Maske erscheint.3. Der Nutzer gibt "nutzer1" und ein falsches Nutzerpasswort (z.B. "123") ein und klickt Login.
erwarteter Ergebniszustand:	Der Nutzer ist nicht angemeldet. Es erscheint die Fehlermeldung „Der Benutzername oder das Kennwort ist falsch.“
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	

TC Login.3: Login deaktivierter Nutzer

Use Case:	Login
Ausgangszustand:	Der Nutzer ist ausgeloggt. Das Profil des Nutzers wurde von einem Administrator deaktiviert.
Aktion:	<ol style="list-style-type: none">1. Der Nutzer gibt die URL "localhost:8080" ein.2. Die Login-Maske erscheint.3. Der Nutzer gibt die Logindaten eines deaktivierten Nutzers ein und klickt Login.
erwarteter Ergebniszustand:	Der Nutzer ist nicht angemeldet. Der Nutzer sieht eine Meldung „Der Benutzername oder das Kennwort ist falsch.“
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	

3.2. Testfälle Einstellungen

TC Einstellungen.1: Eigene Einstellungen bearbeiten

Use Case:	Login
Ausgangszustand:	Der Nutzer ist mit einem beliebigen Benutzer, der kein Administrator ist, angemeldet.
Aktion:	<ol style="list-style-type: none"> 1. Der Nutzer wählt den Punkt "Einstellungen" unter "Account". 2. Die Einstellungen öffnen sich. 3. Der Nutzer gibt eine neue E-Mail-Adresse ein und betätigt den "Speichern" Button. 4. Der Nutzer ändert die Einstellungen für die E-Mail-Benachrichtigungen und betätigt den "Speichern" Button.
erwarteter Ergebniszustand:	Der Nutzer sieht eine "Erfolg" Benachrichtigung. Änderungen wurden persistent gespeichert.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

3.3. Testfälle Produktivitätsanalyse

TC Produktivitätsanalyse.1: Einsehen der Produktivitätsanalyse

Use Case:	Einsehen der firmenweiten Produktivitätsanalyse
Ausgangszustand:	Der Nutzer ist angemeldet.
Aktion:	<ol style="list-style-type: none"> 1. Der Nutzer öffnet den Punkt "Produktivitätsanalyse" unter "Dashboard".
erwarteter Ergebniszustand:	Dashboard mit einer Übersicht der gesamten Firma und der Preisträger (Badges) der letzten Woche öffnet sich.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

3.4. Testfälle Admin

TC Admin.1: Bearbeiten eines Benutzers

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Benutzer" unter "Verwaltung".2. Der Nutzer klickt den "Bearbeiten" Button bei einem beliebigen Nutzer.3. Der Nutzer gibt beliebige Änderung ein.4. Der Nutzer betätigt den "Speichern" Button.
erwarteter Ergebniszustand:	Die Änderungen werden in der Datenbank gespeichert. Der Nutzer sieht bei erneutem Öffnen des Bearbeiten Dialogs die geänderten Daten.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.2: Hinzufügen eines Benutzers

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Benutzer" unter "Verwaltung".2. Der Nutzer klickt den "Benutzer hinzufügen" Button.3. Der Nutzer gibt einen Benutzernamen ein.4. Der Nutzer füllt weitere Daten nach Belieben aus.5. Der Nutzer klickt auf "Erstellen".
erwarteter Ergebniszustand:	Der neu erstellte Benutzer wird in der Datenbank gespeichert und kann vom Nutzer in der Liste eingesehen werden.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.3: Löschen eines Benutzers

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Benutzer" unter "Verwaltung".2. Der Nutzer klickt den "Löschen" Button bei einem beliebigen Nutzer.3. Der Nutzer bestätigt das Löschen.
erwarteter Ergebniszustand:	Der Nutzer wird in der Datenbank deaktiviert und wird aus der Liste entfernt.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.4: Hinzufügen eines Teams/Abteilung

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Team" unter "Verwaltung".2. Der Nutzer klickt den "Team hinzufügen".3. Der Nutzer gibt einen Teamnamen ein.4. Der Nutzer wählt einen Teamleiter aus.5. Der Nutzer klickt auf "Erstellen".
alternativer Ablauf:	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Abteilung" unter "Verwaltung".2. Der Nutzer klickt den "Abteilung hinzufügen".3. Der Nutzer gibt einen Abteilungsnamen ein.4. Der Nutzer wählt einen Abteilungsleiter aus.5. Der Nutzer klickt auf "Erstellen".
erwarteter Ergebniszustand:	Das Team/die Abteilung wird in der Datenbank gespeichert und kann vom Nutzer in der Liste eingesehen werden.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.5: Hinzufügen eines Raums/Miniservers

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Raum" unter "Verwaltung".2. Der Nutzer klickt den "Raum hinzufügen" Button.3. Der Nutzer klickt auf "Erstellen".
alternativer Ablauf:	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Miniserver" unter "Verwaltung".2. Der Nutzer klickt den "Miniserver hinzufügen".3. Der Nutzer klickt auf "Erstellen".
erwarteter Ergebniszustand:	Der Raum/Miniserver wird in der Datenbank gespeichert und kann vom Nutzer in der Liste eingesehen werden.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.6: Hinzufügen eines TimeFlips

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "TimeFlip Würfel" unter "Verwaltung".2. Der Nutzer klickt den "Würfel hinzufügen" Button.3. Der Nutzer gibt eine MAC-Adresse ein.4. Der Nutzer klickt auf "Erstellen".
erwarteter Ergebniszustand:	Der neu erstellte TimeFlip wird in der Datenbank gespeichert und kann vom Nutzer in der Liste eingesehen werden.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.7: Einsehen Audit Log

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	1. Der Nutzer wählt den Punkt "Audit Log" unter "Verwaltung".
erwarteter Ergebniszustand:	Der Audit Log öffnet sich für den Nutzer.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.8: Einsehen Badges

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	1. Der Nutzer wählt den Punkt "Badges" unter "Dashboard".
erwarteter Ergebniszustand:	Eine Übersicht aller vergebenen Badges öffnet sich für den Nutzer.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Admin.9: Löschen eines Badges

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Administrator angemeldet.
Aktion	1. Der Nutzer wählt den Punkt "Badges" unter "Dashboard". 2. Der Nutzer klickt den "Löschen" Button bei einem beliebigen Badge. 3. Der Nutzer bestätigt das Löschen.
erwarteter Ergebniszustand:	Das Badge wird aus der Datenbank gelöscht und wird aus der Liste entfernt.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

3.5. Testfälle Mitarbeiter

TC Employee.1: Einsehen der eigenen Aktivitäten

Use Case:	Einsehen der eigenen Aktivitätsdaten
Ausgangszustand:	Der Nutzer ist als Benutzer ("user64") angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Eigene Aktivitäten" unter "Aktivitätsdaten".2. Der Nutzer wählt im Kalender "29. Mai 2020" und klickt auf "Aktivitäten anzeigen" (=Tagesansicht).3. Der Nutzer wählt Wochenansicht und klickt auf "Aktivitäten anzeigen".4. Der Nutzer wählt Monatsansicht und klickt auf "Aktivitäten anzeigen".
erwarteter Ergebniszustand:	Die Aktivitätsdaten des Benutzers werden in der jeweiligen Ansicht angezeigt als Tabelle und Diagramm angezeigt.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Employee.2: Korrektur von Aktivitätsdaten

Use Case:	Korrektur von Daten
Ausgangszustand:	Der Nutzer ist als Benutzer ("user64") angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Aktivitätskorrektur" unter "Aktivitätsdaten".2. Der Nutzer wählt eine Aktivität, welche nicht älter als 2 Wochen ist.3. Der Nutzer klickt den "Bearbeiten" Button neben der gewählten Aktivität.4. Der Nutzer gibt beliebige Korrekturen ein.5. Der Nutzer speichert die Korrekturen.
erwarteter Ergebniszustand:	Die korrigierten Aktivitätsdaten werden in der Datenbank gespeichert und sind in der Liste einsehbar.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Employee.3: Korrektur von älteren Aktivitätsdaten

Use Case:	Korrektur von Daten
Ausgangszustand:	Der Nutzer ist als Benutzer ("user64") angemeldet.
Aktion	<ol style="list-style-type: none"> 1. Der Nutzer wählt den Punkt "Aktivitätskorrektur" unter "Aktivitätsdaten". 2. Der Nutzer wählt eine Aktivität welche älter als 2 Wochen ist. 3. Der Nutzer klickt den "Bearbeiten" Button neben der gewählten Aktivität. 4. Der Nutzer gibt beliebige Korrekturen ein. 5. Der Nutzer speichert die Korrekturen.
erwarteter Ergebniszustand:	Eine Anfrage zur Änderung der Aktivitätsdaten wurde in der Datenbank gespeichert und kann vom jeweiligen Team- und Abteilungsleiter eingesehen werden.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	

TC Employee.4: Einsehen der eigenen Badges

Use Case:	Einsehen der eigenen Badges
Ausgangszustand:	Der Nutzer ist als Benutzer ("user64") angemeldet.
Aktion	<ol style="list-style-type: none"> 1. Der Nutzer wählt den Punkt "Eigene Badges" unter "Badges".
erwarteter Ergebniszustand:	Die Übersicht aller erhaltenen Badges öffnet sich.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	

TC Employee.5: Urlaub beantragen

Use Case:	Urlaub eintragen
Ausgangszustand:	Der Nutzer ist als Benutzer ("user64") angemeldet.
Aktion	<ol style="list-style-type: none"> 1. Der Nutzer wählt den Punkt "Urlaub beantragen" unter "Urlaub". 2. Der Nutzer wählt ein beliebiges Start- und Enddatum aus. 3. Der Nutzer klickt auf "Speichern".
erwarteter Ergebniszustand:	Der beantragte Urlaub erscheint in der Liste unterhalb mit dem

Status "Ausstehend".

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
☐ System unbenutzbar

3.6. Testfälle Teamleiter

TC Teamleader.1: Einsehen von Aktivitätsdaten

Use Case:

Einsehen von Aktivitätsdaten der Mitarbeiter

Ausgangszustand:

Der Nutzer ist als Benutzer ("user030") angemeldet.

Aktion

1. Der Nutzer wählt den Punkt "Aktivitäten des Teams" unter "Aktivitätsdaten".
2. Der Nutzer wählt im Kalender 18.05.2020.
3. Der Nutzer klickt auf "Aktivitäten anzeigen".
4. Der Nutzer wechselt zwischen Wochen- und Monatsansicht.

erwarteter Ergebniszustand:

Alle Aktivitäten des Teams werden in der jeweiligen Ansicht angezeigt.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
☐ System unbenutzbar

TC Teamleader.2: Bestätigung/Löschen von Korrekturanfragen

Use Case:

Bestätigung einer Korrekturanfrage

Ausgangszustand:

Der Nutzer ist als Benutzer ("user030") angemeldet. Der Test TC Employee.3 wurde erfolgreich durchgeführt.

Aktion

1. Der Nutzer wählt den Punkt "Korrekturanfragen" unter "Aktivitätsdaten".
2. Der Nutzer klickt den "Bestätigen" Button bei einer Anfrage.
3. Der Nutzer bestätigt die Änderung.

erwarteter Ergebniszustand:

Die korrigierte Änderung wird in der Datenbank gespeichert und die Anfrage wird aus der Liste der Änderungsanfragen entfernt.

alternativer Ablauf:

1. Der Nutzer wählt den Punkt "Korrekturanfragen" unter "Aktivitätsdaten".
2. Der Nutzer klickt den "Löschen" Button bei einer Anfrage.
3. Der Nutzer bestätigt die Änderung.

alternativer erwarteter Ergebniszustand:

Die Änderung werden nicht in der Datenbank gespeichert und die Anfrage wird aus der Liste der Änderungsanfragen entfernt.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
- ☐ System unbenutzbar

TC Teamleader.3: Bestätigung/Löschen von Urlaubsanfragen**Use Case:**

Bestätigung einer Urlaubsanfrage

Ausgangszustand:

Der Nutzer ist als Benutzer ("user030") angemeldet. Der Test TC Employee.5 wurde erfolgreich durchgeführt.

Aktion

1. Der Nutzer wählt den Punkt "Urlaubsanfragen" unter "Urlaub".
2. Der Nutzer klickt den "Bestätigen" Button bei einer Anfrage.
3. Der Nutzer bestätigt die Änderung.

erwarteter Ergebniszustand:

Der Urlaub wird auf den Status "Bestätigt" gesetzt und die Anfrage wird aus der Liste gelöscht.

alternativer Ablauf:

1. Der Nutzer wählt den Punkt "Urlaubsanfragen" unter "Urlaub".
2. Der Nutzer klickt den "Ablehnen" Button bei einer Anfrage.
3. Der Nutzer bestätigt die Änderung.

alternativer erwarteter Ergebniszustand:

Der Urlaub wird auf den Status "Abgelehnt" gesetzt und die Anfrage wird aus der Liste gelöscht.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
- ☐ System unbenutzbar

3.7. Testfälle Abteilungsleiter

TC Departmentleader.1: Einsehen von Aktivitätsdaten

Use Case:	Einsehen von Aktivitätsdaten der Mitarbeiter
Ausgangszustand:	Der Nutzer ist als Benutzer ("user003") angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Aktivitäten der Mitarbeiter" unter "Aktivitätsdaten".2. Der Nutzer wählt bei der Monatsauswahl "Apr 2020".3. Der Nutzer wählt bei der Teamauswahl "Alle Teams" oder trifft eine andere Auswahl.4. Der Nutzer klickt auf "Aktivitäten anzeigen"
erwarteter Ergebniszustand:	Alle Aktivitäten des/der Teams werden in Monatsansicht angezeigt.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

TC Departmentleader.2: Einsehen Audit Log

Use Case:	Verwaltung
Ausgangszustand:	Der Nutzer ist als Abteilungsleiter ("user003") angemeldet.
Aktion	<ol style="list-style-type: none">1. Der Nutzer wählt den Punkt "Audit Log" unter "Verwaltung".
erwarteter Ergebniszustand:	Der Audit Log öffnet sich für den Nutzer.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

3.8. Testfälle Raspberry Pi

TC Raspberry Pi.1: Starten des RestService auf der Seite des PI

Use Case:	Einsehen der eigenen Aktivitätsdaten
Ausgangszustand:	Der Nutzer hat die Software ausgepackt und der Würfel hat den Namen "TimeFlip1". Der Testcase TC Raspberry PI.2 wurde ausgeführt.
Aktion	<ol style="list-style-type: none"> 1. Sollte der PI neu aufgesetzt sein sollte das Script "install.sh" ausgeführt werden. 2. Im entsprechenden Ordner (src_raspberry/BluetoothConnection) den Befehl "mvn clean install" ausführen. 3. Abschließend den Befehl "sudo java -cp target/ble-start-0.0.1-SNAPSHOT.jar:./lib/tinyb.jar:./target/lib/* com.example.setup.HelloTinyB <IP:Port>" eingeben. IP entspricht der IP-Adresse des Servers (eingestellt in der Properties Datei, wurde das Programm mit Docker gestartet sollte die locale IP-Adresse eingetragen werden) und der Port entspricht den Port des Servers (8080). 4. Da die Daten immer um kurz vor 24 Uhr eingelesen werden muss die Zeit auf den Pi zurückgestellt werden oder bis 0 Uhr warten. Dies kann mit dem Befehlen "systemctl disable systemd-timesyncd.service" und "sudo date -s "Thu May 28 21:57:00 UTC 2020"" erreicht werden. Oder einfach bis 00 warten bis die Daten gesendet werden.
erwarteter Ergebniszustand:	1. Die Daten werden übertragen oder der Server wartet auf 0 Uhr.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	

TC Raspberry Pi.2: Starten des RestService auf der Seite des Skeleton

Use Case:	Einsehen der eigenen Aktivitätsdaten
Ausgangszustand:	Der Nutzer ist als Benutzer admin angemeldet. Außerdem sollen die Seitennumerierungen des Würfels bekannt sein. Für die ersten 4 Punkte sollte der Server noch nicht gestartet sein,
Aktion	<ol style="list-style-type: none"> 1. Auch hier kann entweder wieder bis 24 Uhr gewartet werden oder in der Klasse "RestService" die Zeiten ändern. Hierzu muss bei "writeTaks" das @Scheduled(cron="0 00 * * *) geändert werden (Format Sekunden Minuten Stunden * * *). 2. Die Seiten des Würfels in der Klasse "RestService"

erwarteter Ergebniszustand: beobachtete Abweichung <input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	eintragen da sich diese bei jedem TimeFlipWürfel unterscheiden. 3. In Properties die IP-Adresse und einen Port eingeben der genutzt werden sollt. Dies kann ignoriert werden wenn der Server mit Docker gestartet wurde. 4. Das Programm starten. 5. Einen neuen TimeFlip Würfel erstellen (siehe TC Admin.6) mit der MAC-Adresse des Würfels denn man besitzt. Den Würfel auch einen Nutzer seiner Wahl zuweisen. 6. Mit dem zugewiesenen Nutzer einloggen. 7. Nach ablaufen der Zeit kann auf "Eigene Aktivitäten" geklickt werden. 8. Auf "Aktivitäten Anzeigen" klicken. Wenn auch die Schritte von TC Raspberry Pi.1 durchgeführt worden sind wird ein Pie Diagramm von den einzelnen Daten wird angezeigt.
---	---

TC Raspberry Pi.3: Niedrige Batteriewarnung

Use Case: Ausgangszustand: Aktion erwarteter Ergebniszustand: beobachtete Abweichung <input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar	/ Der Nutzer ist als Benutzer der im vorherigen Testcase den Würfel zugewiesen worden ist angemeldet und die Schritte von Testcase Pi.1 und Pi.2 wurden durchgeführt. Für Testzwecke könnte man den Stand erhöhen ab den die Warnung gesendet wird. 1. Auch hier kann entweder wieder bis 24 Uhr gewartet werden oder in der Klasse "RestService" die Zeiten ändern. Hierzu muss bei "checkWarning" das @Scheduled(cron="0 0 0 * * *") geändert werden (Format Sekunden Minuten Stunden * * *). 2. Das Programm starten. Nach Ablauf der Zeit wird dem Nutzer eine Mail-Benachrichtigung gesendet.
---	--

TC Raspberry Pi.4: Verbindungsproblem Warnung

Use Case:	/
Ausgangszustand:	Der Nutzer ist als Benutzer der im vorherigen Testcase den Würfel zugewiesen worden ist angemeldet und die Schritte von Testcase Pi.1 und Pi.2 wurden durchgeführt. Der Würfel sollte nicht erreichbar sein. (Keine Batterie zum Beispiel).
Aktion	1. Gleicher Ablauf wie Pi.3
erwarteter Ergebniszustand:	Nach Ablauf der Zeit wird dem Nutzer eine Mail-Benachrichtigung gesendet.
beobachtete Abweichung	
<input checked="" type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen	
<input type="checkbox"/> System unbenutzbar	

3.9. Weitere nichtfunktionale Testfälle

NFT.1: Mailversand

Use Case:	Urlaub eintragen/ Bestätigung einer Urlaubsanfrage
Ausgangszustand:	Der Nutzer ist als Mitarbeiter ("user64") angemeldet.
Aktion	<ol style="list-style-type: none"> Der Nutzer wählt den Punkt "Einstellungen" unter "Account" und ändert die E-Mail-Adresse auf eine real existierende Adresse. <p>Es soll getestet werden, ob der Mailversand funktioniert.</p> <p>Für eine optimale Testung sind weitere Schritte zu erfüllen:</p> <ol style="list-style-type: none"> Der Nutzer ändert auch beim Teamleiter ("user030") die E-Mail-Adresse auf eine real existierende Adresse. Der Nutzer ändert auch beim Abteilungsleiter ("user003") die E-Mail-Adresse auf eine real existierende Adresse. <p>Nach der Änderung sind folgende Schritte durchzuführen:</p> <ol style="list-style-type: none"> Der Nutzer wählt den Punkt "Urlaubsanfragen" unter "Urlaub". Der Nutzer wählt ein Start- und ein Enddatum in der Zukunft. Der Nutzer klickt auf Speichern. <p>Danach kann diese Anfrage beim Abteilungs- und/ oder Teamleiter bestätigt/ gelöscht werden (siehe TC Teamleader.3).</p>

erwarteter Ergebniszustand:

Eine Bestätigungsmail wird an den Nutzer gesendet. Der Team- und der Abteilungsleiter erhalten eine Informationsmail und können die Anfrage anschließend bestätigen/ löschen, woraufhin eine Informationsmail an den User gesendet wird.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
☐ System unbenutzbar

Usability.1: TimeFlip Konfiguration**Use Case:**

Verwaltung

Ausgangszustand:

Der Nutzer ist als Administrator angemeldet.

Aktion

1. Der Nutzer fügt einen neuen TimeFlip Würfel hinzu.
2. Der Nutzer konfiguriert die Seiten des Würfels.

erwarteter Ergebniszustand:

Der Nutzer kann die Seiten des TimeFlip Würfels konfigurieren.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
☐ System unbenutzbar

Security.1: Sicherheit**Use Case:**

Verwaltung

Ausgangszustand:

Der Nutzer ist als Mitarbeiter angemeldet.

Aktion

1. Der Nutzer gibt die URL
"localhost:8080/menuitems/audit_log.xhtml"

erwarteter Ergebniszustand:

Die "Zugriff verweigert." Seite öffnet sich.

beobachtete Abweichung

- ☒ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen
☐ System unbenutzbar