

Testdrehbuch und Protokoll

Team: Team 2

Mitglied 1: Hans Georg Gallmetzer, 11706889

Mitglied 2: Elias Pfurtscheller, 01417965

Mitglied 3: Maria Wurzer, 01417577

Mitglied 4: Christina Strohmer, 11779004

Mitglied 5: Stefanie Haid, 11714701

Mitglied 6: Tobias Höpfl, 11716481

Proseminargruppe: Gruppe 1

Datum: 13.05.2021

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 Integrationstests. 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 |
|---------|-------|--------|-------|----------|----------------------|--------------|
| | | | | | | |

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 weitere Use Cases | 8 |
| 3.3. | Weitere nichtfunktionale Testfälle | Fehler! Textmarke nicht definiert. |
| 4. | Anhang | 30 |
| 4.1. | Glossar | Fehler! Textmarke nicht definiert. |
| 4.2. | Referenzierte Dokument | 30 |

1. Testvorbereitung

Um die Tests durchzuführen, muss der Server gestartet sein. Für manche Test muss ein Würfelwurf simuliert werden (etwa mit Insomnia) oder der Client auf dem RaspberryPi muss gestartet sein und sich mit einem TimeFlip Würfel verbunden haben.

1.1. Testdaten

Zu Testbeginn sind folgende Nutzer eingerichtet:

| Nutzer / Password | Rolle | Bemerkung |
|-------------------|-------------------|-----------|
| admin / passwd | Administrator | |
| susi1 / passwd | Spieleerverwalter | |
| max2 / passwd | Spieler | |
| elvis / passwd | Administrator | |
| armin / passwd | Administrator | |

Initialer Testdatenbestand: (Beispiele)

- Nutzerdaten (siehe oben), ggf. einschließlich weiterer Daten (Email-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: 17.05.2021 - 20.05.2021

Tester: Gruppe 1 Team 1

Getestete Version:

possible_virus_g1t2-master.zip__g1t2-master_src_main_webapp_resources_js_chat.j-

Testeingangskriterien erfüllt: unbekannt

Testumgebung: Docker (Server), Raspberry Pi 4 mit "frischer" Installation von Raspberry Pi OS Lite (Client)

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 Auth1.1: Login Nutzer

| | |
|---|--|
| Use Case: | Login |
| Ausgangszustand: | Der Nutzer (armin) 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 armin und sein Nutzerpasswort passwd ein und klickt Login. |
| erwarteter Ergebniszustand: | <p>Der Nutzer ist angemeldet.</p> <p>Der Nutzer sieht die Startseite.</p> |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth1.2: 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 einen falschen Nutzernamen und/oder ein falsches Nutzerpasswort ein und klickt Login. |
| erwarteter Ergebniszustand: | Der Nutzer ist nicht angemeldet. |

beobachtete Abweichung

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

Es erscheint die Fehlermeldung „Login unsuccessful. Try another username or password.“

TC Auth1.3: Logout

Use Case: Logout

Ausgangszustand: Der Nutzer ist eingeloggt und in keinem aktiven Spiel.

Aktion 1. Der Nutzer klickt auf „Logout“.

erwarteter Ergebniszustand: Der Nutzer ist nicht mehr angemeldet.

Die Login-Seite ist sichtbar. Der User hat die Möglichkeit sich erneut einzuloggen.

beobachtete Abweichung

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

TC Auth1.4: Registrierung

Use Case: Auth2

Ausgangszustand: Der Nutzer hat keine gültigen Logindaten und muss sich erst registrieren.

Aktion 1. Der Nutzer klickt auf „Register“. Es erscheint das Formular „Register as new Player“.
2. Der Nutzer gibt Username, Passwort, Vor- und Nachname, sowie E-Mail-Adresse in die dafür vorgesehenen Felder ein und klickt auf „Save“.

erwarteter Ergebniszustand: Das Feld „Thanks for your Registration“ erscheint. Darunter der Hinweis, dass die Daten zuerst von einem Admin genehmigt werden müssen.

Nach Schließen des Pop-Ups ist die Login-Seite sichtbar.

beobachtete Abweichung

- ☐ OK ☐ kosmetische Abweichungen ☐ mittlere

Abweichungen ☐ große Abweichungen

☐ System unbenutzbar

TC Auth1.5: Registrierung fehlgeschlagen

| | |
|--|---|
| Use Case: | Auth2 |
| Ausgangszustand: | Der Nutzer hat keine gültigen Logindaten und muss sich erst registrieren. |
| Aktion | <ol style="list-style-type: none"> 1. Der Nutzer klickt auf „Register“. Es erscheint das Formular „Register as new Player“. 2. Der Nutzer gibt Username, Passwort, Vor- und Nachname, sowie E-Mail-Adresse ein. Ein oder mehrere Felder bleiben frei und klickt auf „Save“. |
| erwarteter Ergebniszustand: | Es erscheint die Warnung, dass eines oder mehrere Felder nicht ausgefüllt wurden und der Hinweis es erneut zu versuchen. Wird dieses Feld geschlossen, bleibt das Feld mit den unvollständigen Eingaben und der Nutzer kann diese vervollständigen. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth1.6: Registrierung - Username schon vergeben

| | |
|-----------------------------|---|
| Use Case: | Auth2 |
| Ausgangszustand: | Der Nutzer hat keine gültigen Logindaten und muss sich erst registrieren. |
| Aktion | <ol style="list-style-type: none"> 1. Der Nutzer klickt auf „Register“. Es erscheint das Formular „Register as new Player“. 2. Der Nutzer gibt einen bereits vorhandenen Username, Passwort, Vor- und Nachname, sowie E-Mail-Adresse ein und klickt auf „Save“. |
| erwarteter Ergebniszustand: | Es erscheint die Warnung, dass dieser Username bereits vergeben wurde und man deshalb einen anderen wählen soll. Die bereits ausgefüllten Felder bleiben ausgefüllt. |
| beobachtete Abweichung | |

| | |
|---|--------------------------------------|
| <input 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 weitere Use Cases

TC Auth3.1: Spiel erstellen

| | |
|---|---|
| Use Case: | Spiel erstellen |
| Ausgangszustand: | Der Nutzer ist eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Nutzer klickt auf „Player Submenu“ und anschließend auf „Make new game“. 2. Es erscheint eine Maske zur Eingabe der Spieldaten. 3. Er vergibt dem Spiel einen Namen (erforderlich). 4. Bei Bedarf können die Werte in den Feldern „Score to win“ und „number of teams“ überschrieben werden. 5. Der Nutzer vergibt einen Namen für sein Team (Feld „Name of your team“ (erforderlich). 6. Der Nutzer wählt ein Topic und einen RaspberryPi. 7. Der Nutzer klickt auf „Save“. |
| erwarteter Ergebniszustand: | <p>Ein Pop-Up erscheint, dass das Spiel erfolgreich erstellt wurde.</p> <p>Nach Schließen des Pop-Ups ist der User automatisch im Spielraum.</p> |
| beobachtete Abweichung | man muss neu laden wenn neuer würfel hinzugefügt wird |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.2: Spiel erstellen – unvollständige Eingabe

| | |
|------------------|--|
| Use Case: | Spiel erstellen |
| Ausgangszustand: | Der Nutzer ist eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Nutzer klickt auf „Player Submenu“ und anschließend auf „Make new game“. 2. Es erscheint eine Maske zur Eingabe der Spieldaten. 3. Er vergibt dem Spiel einen Namen (erforderlich). 4. Bei Bedarf können die Werte in den Feldern „Score to win“ und „number of teams“ überschrieben werden. |

| | |
|---|---|
| | <ol style="list-style-type: none"> Der Nutzer vergibt einen Namen für sein Team (Feld „Name of your team“ (erforderlich). Eines oder mehrere Felder bleiben leer. Der Nutzer wählt ein Topic und einen RaspberryPi. Der Nutzer klickt auf „Save“. |
| erwarteter Ergebniszustand: | Es erscheint ein Pop-Up mit der Warnung, dass eines oder mehrere Felder leer sind und dass man es erneut versuchen soll. |
| beobachtete Abweichung | keine |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.3: Spiel erstellen – Spielname vergeben

| | |
|---|---|
| Use Case: | Spiel erstellen |
| Ausgangszustand: | Der Nutzer ist eingeloggt. |
| Aktion | <ol style="list-style-type: none"> Der Nutzer klickt auf „Player Submenu“ und anschließend auf „Make new game“. Es erscheint eine Maske zur Eingabe der Spieldaten. Er vergibt dem Spiel einen Namen, der bereits vergeben wurde. Bei Bedarf können die Werte in den Feldern „Score to win“ und „number of teams“ überschrieben werden. Der Nutzer vergibt einen Namen für sein Team (Feld „Name of your team“ (erforderlich). Der Nutzer wählt ein Topic und einen RaspberryPi. Der Nutzer klickt auf „Save“. |
| erwarteter Ergebniszustand: | Es erscheint ein Pop-Up mit der Warnung, dass dieser Spielname bereits vergeben wurde und dass man es erneut versuchen soll. |
| beobachtete Abweichung | Pop-up erscheint, man kann aber kein neues Spiel erstellen wenn man den Namen ändert -> Raspberry Pi wird auf „available: false“ gestellt, obwohl nicht zugewiesen, das neuerliche speichern funktioniert nicht (siehe Fehlermeldung beim Speichern in die Datenbank) |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.4: Spieldaten ändern

| | |
|---|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt, aber noch nicht gestartet. Der Nutzer befindet sich im Spielraum. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht die Details des Spieles, in welchem er sich befindet. 2. Der Benutzer klickt auf das Stift-Symbol neben dem betreffenden Spiel, um die Spieledaten zu ändern. 3. Pop-Up erscheint, in dem die Änderungen vorgenommen werden können: <ol style="list-style-type: none"> a. Der Benutzer wählt ein neues Thema. b. Der Benutzer gibt ein neues Punktemaximum ein. c. Der Benutzer gibt eine neue Teamanzahl ein. 4. Benutzer klickt auf „Save“. |
| erwarteter Ergebniszustand: | In den entsprechenden Feldern im Spielraum stehen die geänderten Daten. Nicht ausgefüllte Felder werden auf den Default-Wert gesetzt. |
| beobachtete Abweichung | keine |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.5: Spiel beitreten – bestehendes Team

| | |
|------------------|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt, befindet sich aber in keinem Spiel. Ein Spiel wurde erstellt, aber noch nicht gestartet. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Player Submenu“ und anschließend auf „Game Lobby“. 2. Der Nutzer sieht unter der Spielstatistik eine Liste aller offenen Spiele. 3. Der Benutzer klickt neben dem gewünschten Spielraum auf das Check-Symbol („Join“). 4. Ein Pop-Up erscheint, in dem die bisherigen Teams aufgelistet sind, darunter ist der Button „Make new team“. 5. Der Benutzer klickt neben dem Team, welchem er beitreten möchte, auf das Check-Symbol („Join team“). 6. Ein Pop-Up erscheint, dass das Spiel erfolgreich gespeichert wurde. 7. Nach dem Schließen des Pop-Ups befindet sich der Benutzer automatisch im Spielraum. |

| | |
|---|---|
| erwarteter Ergebniszustand: | Der Benutzer befindet sich im Spielraum und wurde dem entsprechenden Team zugeordnet. |
| beobachtete Abweichung | keine |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.5: Spiel beitreten – neues Team

| | |
|---|---|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt, befindet sich aber in keinem Spiel. Ein Spiel wurde erstellt, aber noch nicht gestartet. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Player Submenu“ und anschließend auf „Game Lobby“. 2. Der Nutzer sieht unter der Spielstatistik eine Liste aller offenen Spiele. 3. Der Benutzer klickt neben dem gewünschten Spielraum auf das Check-Symbol („Join“). 4. Ein Pop-Up erscheint, in dem die bisherigen Teams aufgelistet sind, darunter ist der Button „Make new team“. 5. Der Benutzer klickt auf „Make new team“. 6. Pop-Up erscheint, der Nutzer gibt einen Namen für sein Team ein und bestätigt mit Klick auf das Check-Symbol. Ist der Teamname bereits vergeben, erscheint eine entsprechende Warnung und der Nutzer wird aufgefordert einen anderen Namen zu wählen. 7. War die Eingabe erfolgreich, erscheint ein Pop-Up, dass das Spiel erfolgreich gespeichert wurde. 8. Nach dem Schließen des Pop-Ups befindet sich der Benutzer automatisch im Spielraum. |
| erwarteter Ergebniszustand: | Der Benutzer befindet sich im Spielraum und wurde dem entsprechenden Team zugeordnet. |
| beobachtete Abweichung | keine |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.6: Spiel beitreten – zu viele Teams

| | |
|-----------|-------|
| Use Case: | Spiel |
|-----------|-------|

| | |
|---|--|
| Ausgangszustand: | Der Nutzer ist eingeloggt, befindet sich aber in keinem Spiel. Ein Spiel wurde erstellt, aber noch nicht gestartet. Es gibt bereits x/x Teams. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Player Submenu“ und anschließend auf „Game Lobby“. 2. Der Nutzer sieht unter der Spielstatistik eine Liste aller offenen Spiele. 3. Der Benutzer klickt neben dem gewünschten Spielraum auf das Check-Symbol („Join“). 4. Ein Pop-Up erscheint, in dem die bisherigen Teams aufgelistet sind, darunter ist der Button „Make new team“. 5. Der Benutzer klickt auf „Make new team“. |
| erwarteter Ergebniszustand: | Ein Pop-Up erscheint, dass es bereits genug Teams gibt, der Nutzer wird aufgefordert einem bestehenden Team beizutreten. |
| beobachtete Abweichung | Dialog zum Auswählen eines bestehenden Team wird geschlossen |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth3.8: Spiel beitreten – zweites Spiel

| | |
|-----------------------------|---|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich bereits in einem Spiel. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Player Submenu“ und anschließend auf „Game Lobby“. 2. Der Nutzer sieht unter der Spielstatistik eine Liste aller offenen Spiele. 3. Der Benutzer klickt neben dem gewünschten Spielraum auf das Check-Symbol („Join“). 4. Ein Pop-Up erscheint, dass der Nutzer bereits in einem Spiel oder in einem Team in diesem Spiel ist. 5. Der Nutzer klickt auf „Put me back“. |
| erwarteter Ergebniszustand: | Der Nutzer ist im Spielraum. |
| beobachtete Abweichung | <p>bei Schritt 3. passiert nichts, erst beim Anklicken des Check-Symbols „join team“; funktioniert wenn sich Spieler für das Game in dem er sich aktuell befindet nochmal anmelden will</p> <p>wenn sich Spieler für ein anderes Spiel anmelden will, passiert nichts wenn man auf „join team“ klickt, wenn man auf „make new Team“ klickt, kann ein neuer Team-Name ausgewählt werden, danach kommt eine Fehlermeldung - die allerdings nicht passt, da der Spieler ja noch nicht in diesem Spiel ist, sondern in einem anderen.. wenn man das ganze wiederholt und den Team-Namen</p> |

nicht verändert, passiert nichts mehr, wenn man auf den "Confirm-Button" klickt

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

TC Auth4.1: Würfel – Seiten zuordnen

| | |
|--|--|
| Use Case: | Würfel |
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt, aber noch nicht gestartet. Der Nutzer befindet sich im Spielraum und hat den Würfel. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht die Details des Spieles, in welchem er sich befindet. 2. Der Benutzer dreht den Würfel, wählt die entsprechende Einstellung aus und klickt auf „Apply changes“. |
| erwarteter Ergebniszustand: | Im Spiel wird bei Drehung des Würfels die richtige Seite angezeigt. |
| beobachtete Abweichung | alle Spiel-Teilnehmer können die Zuordnung vornehmen, nicht nur der Spieler mit dem Würfel, man erkennt nicht ob alle Seiten belegt sind, mehrere Seiten können mit dem gleichen Face belegt werden |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth5.1: Spiel starten

| | |
|------------------|---|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt, aber noch nicht gestartet. Der Nutzer befindet sich im Spielraum. Weiter Vorbedingungen nicht erwähnt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht die Details des Spieles, in welchem er sich befindet. 2. Der Benutzer klickt auf „start game“ 3. Ein Pop-Up erscheint, mit der Frage, ob das Spiel gestartet werden soll. 4. Der Benutzer klickt auf „Start“. 5. Bei allen anderen eingeloggten Spielern in dem Spielraum erscheint ein Pop-Up, mit der Frage, ob sie bereit sind. 6. Die anderen eingeloggten Spieler bestätigen, dass sie bereit sind. |

| | |
|---|---|
| erwarteter Ergebniszustand: | <p>7. Bis alle bestätigt haben, erscheint ein Pop-Up, dass noch auf andere Spieler gewartet wird.</p> <p>Das Pop-Up, dass noch auf andere Spieler gewartet wird, schließt sich.</p> <p>Der Benutzer sieht, in welchem Team er sich befindet (Rateteam oder Kontrollteam):</p> <p>Der Benutzer befindet sich im Rateteam: Er sieht, welches Teammitglied an der Reihe ist, welche Aktion ausgeführt werden muss und wie viele Punkte das Erraten gibt.</p> <p>Der Benutzer befindet sich im Kontrollteam: Er sieht, welcher Begriff mit welcher Aktion dargestellt werden muss und kann den Begriff beim Erraten bestätigen sowie einen Regelverstoß vermerken.</p> |
| beobachtete Abweichung | Begriff wird erst nach würfeln angezeigt. |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth5.2: Spiel starten – weniger als 2 Teams

| | |
|---|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt, aber noch nicht gestartet. Der Nutzer befindet sich im Spielraum. Es ist nur ein Team im Spiel. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht die Details des Spieles, in welchem er sich befindet. 2. Der Benutzer klickt auf „start game“ 3. Ein Pop-Up erscheint, mit der Frage, ob das Spiel gestartet werden soll. 4. Der Benutzer klickt auf „Start“. |
| erwarteter Ergebniszustand: | <p>Es erscheint ein Pop-Up mit der Warnung, dass weniger als zwei Teams eingeloggt sind und das Spiel so nicht gestartet werden kann.</p> <p>Das Spiel wird nicht gestartet. Nach Schließen des Pop-Ups befindet sich der User wieder in der Spieleübersicht.</p> |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth5.3: Spiel starten – weniger als 2 Teammitglieder

| | |
|-----------|-------|
| Use Case: | Spiel |
|-----------|-------|

| | |
|---|--|
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt, aber noch nicht gestartet. Der Nutzer befindet sich im Spielraum. <i>Es ist nur ein Team im Spiel. (Ist das nicht 5.2? - Nehme an hier 2 Teams, 1 mit nur 1 Spieler).</i> |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht die Details des Spieles, in welchem er sich befindet. 2. Der Benutzer klickt auf „start game“ 3. Ein Pop-Up erscheint, mit der Frage, ob das Spiel gestartet werden soll. 4. Der Benutzer klickt auf „Start“. |
| erwarteter Ergebniszustand: | Es erscheint ein Pop-Up mit der Warnung, dass weniger als zwei Spieler in einem oder mehreren Teams eingeloggt sind und das Spiel so nicht gestartet werden kann. Das Spiel wird nicht gestartet. Nach Schließen des Pop-Ups befindet sich der User wieder in der Spieleübersicht. |
| beobachtete Abweichung | <i>Funktioniert - Vorbedingung falsch beschrieben.</i> |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth5.4: Spiel beenden

| | |
|--|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt. Ein Spiel wurde erstellt und gestartet. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht wer an der Reihe ist und den Punktestand. 2. Der Benutzer klickt auf „end game“. |
| erwarteter Ergebniszustand: | Es wird eine Punkteübersicht angezeigt und der Hinweis, dass ein Mitspieler das Spiel beendet hat. Mit „Leave Game“ wird diese Übersicht geschlossen und man landet wieder in der Game Lobby. |
| beobachtete Abweichung | <i>Unklar ob Spiel für Statistik gewertet werden soll oder nicht.</i> |
| <input type="checkbox"/> OK <input type="checkbox"/> <i>kosmetische Abweichungen</i> <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth6.1: Begriff erraten

| | |
|-----------|-------|
| Use Case: | Spiel |
|-----------|-------|

| | |
|---|---|
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Ein gegnerisches Team ist an der Reihe einen Begriff zu erraten und hat diesen richtig erraten. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht den vom anderen Team zu erratenden Begriff, sowie die Button „Richtig“ und „Regelverstoß“. 2. Der Benutzer klickt auf „Richtig“. |
| erwarteter Ergebniszustand: | <p>Der Begriff verschwindet vom Bildschirm.</p> <p>Der Punktestand des Rateteams wird erhöht.</p> <p>Der Spieler vom nächsten Team, der jetzt den Begriff darstellen soll, wird angezeigt.</p> |
| beobachtete Abweichung | die Buttons "Richtig" und "Regelverstoß" erscheinen erst wenn der Würfel eine neue Seite sendet - Testcase 6.6 beschreibt den UseCase treffender... |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth6.2: raten – Zeit abgelaufen

| | |
|---|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Ein gegnerisches Team ist an der Reihe einen Begriff zu erraten und hat diesen in der vorgegebenen Zeit nicht erraten. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht „Zeit um“. |
| erwarteter Ergebniszustand: | <p>Der Begriff verschwindet vom Bildschirm.</p> <p>Der Punktestand des Rateteams wird nicht erhöht.</p> <p>Der Spieler vom nächsten Team, der jetzt den Begriff darstellen soll, wird angezeigt.</p> |
| beobachtete Abweichung | es erscheint ein PopUp zum Bestätigen - der Rest entspricht der Beschreibung |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth6.3: raten – alle Begriffe wurden durchgespielt

| | |
|------------------|---|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Ein gegnerisches Team ist an der Reihe einen Begriff zu |

| | |
|--|--|
| Aktion | erraten und hat diesen richtig erraten. Dies war der letzte in der gewählten Kategorie. |
| erwarteter Ergebniszustand: | <ol style="list-style-type: none"> 1. Der Benutzer sieht den vom anderen Team zu erratenden Begriff, sowie die Button „Richtig“ und „Regelverstoß“. 2. Der Benutzer klickt auf „Richtig“. <p>Der Begriff verschwindet vom Bildschirm.</p> <p>Der Punktestand des Rateteams wird erhöht.</p> <p>Das Spiel wird beendet.</p> |
| beobachtete Abweichung | der Bildschirm „Spiel beendet“ wird nur bei einem Team angezeigt - das zweite Team verbleibt im Spielraum... |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth6.4: raten - Regelverstoß

| | |
|--|---|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Ein gegnerisches Team ist an der Reihe einen Begriff zu erraten und hat gegen eine Regel verstoßen. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer sieht den vom anderen Team zu erratenden Begriff, sowie die Button „Richtig“ und „Regelverstoß“. 2. Der Benutzer klickt auf „Regelverstoß“. |
| erwarteter Ergebniszustand: | <p>Die zu erreichenden Punkte für den derzeitigen Begriff, werden für das gegnerische Team um 1 verringert.</p> <p>Das Team hat bereits so oft gegen eine Regel verstoßen, dass die zu erreichenden Punkte für diesen Begriff 0 sind, dann erscheint der Button „Next“ und das nächste Team mit neuem Begriff ist an der Reihe.</p> |
| beobachtete Abweichung | Anforderungen aus der Aufgabenstellung nicht richtig umgesetzt |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth6.5: Spiel läuft

| | |
|------------------|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Er ist an der Reihe zu raten. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer wird aufgefordert den Würfel zu drehen und sich zu einem gegnerischen Team zu begeben. |

| | |
|---|---|
| erwarteter Ergebniszustand: | 2. Der Benutzer dreht den Würfel und liest den entsprechenden Begriff bei einem gegnerischen Team. Das Rateteam sieht den Countdown. |
| beobachtete Abweichung | Die Kontrollteams sehen den zu erratenden Begriff, sowie die Felder „Richtig“ und „Regelverstoß“. Die Felder „Richtig“ und „Regelverstoß“ erscheinen erst bei erneutem Würfelwurf... Testcase 6.6 beschreibt den UseCase treffender... |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth6.6: Begriff erraten – Würfel gedreht

| | |
|---|--|
| Use Case: | Spiel |
| Ausgangszustand: | Der Nutzer ist eingeloggt und befindet sich in einem aktuellen Spiel. Der Nutzer ist an der Reihe einen Begriff zu erraten und hat diesen richtig erraten. |
| Aktion | 1. Der Benutzer dreht den Würfel. |
| erwarteter Ergebniszustand: | Das gegnerische Team kann bestätigen, ob der Begriff ohne Regelverstoß erraten wurde. Der Begriff verschwindet vom Bildschirm. Der Punktestand des Rateteams wird erhöht. Der Spieler vom nächsten Team, der jetzt den Begriff darstellen soll, wird angezeigt. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth7.1: Themengebiet erstellen

| | |
|------------------|--|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer klickt auf „Create new Topic“. 4. Ein Pop-Up Fenster erscheint, in dem Angaben zum neuen Thema gemacht werden: |

| | |
|---|--|
| | <ol style="list-style-type: none"> Der Benutzer gibt dem neuen Themengebiet einen Namen. Der Benutzer gibt die Begriffe ein, die im Themengebiet enthalten sein sollen. <ol style="list-style-type: none"> Der Benutzer klickt auf „Submit“. Pop-Up erscheint, in dem nachgefragt wird, ob das neue Thema gespeichert werden soll. Benutzer klickt auf „Yes“. |
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich.</p> <p>In der Liste mit den Themengebieten scheint das neue Thema auf (nach Neuladen der Seite).</p> <p>Die eingegebenen Begriffe sind dem Thema zugeordnet.</p> |
| beobachtete Abweichung | Neuladen alleine hat nicht funktioniert - musste erst auf andere Seite und dann zurück |
| <input type="checkbox"/> OK <input checked="" type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.2: Themengebiet erstellen – bereits vorhanden

| | |
|---|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> Der Benutzer klickt auf „Manager Submenu“. Der Benutzer klickt auf „Topic Management“. Der Benutzer klickt auf „Create new Topic“. Ein Pop-Up Fenster erscheint, in dem Angaben zum neuen Thema gemacht werden: <ol style="list-style-type: none"> Der Benutzer gibt dem neuen Themengebiet einen Namen, den es bereits gibt. Der Benutzer gibt die Begriffe ein, die im Themengebiet enthalten sein sollen. Der Benutzer klickt auf „Submit“. |
| erwarteter Ergebniszustand: | Pop-Up erscheint, in dem darauf hingewiesen wird, dass der Name angepasst werden muss. |
| beobachtete Abweichung | Meldung: „Error: Either no terms given or topic already exists“ ist unspezifisch. |
| <input type="checkbox"/> OK <input checked="" type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.3: Themengebiet erstellen – doppelte Begriffe

| | |
|---|--|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer klickt auf „Create new Topic“. 4. Ein Pop-Up Fenster erscheint, in dem Angaben zum neuen Thema gemacht werden: <ol style="list-style-type: none"> a. Der Benutzer gibt dem neuen Themengebiet einen Namen. b. Der Benutzer gibt die Begriffe ein, die im Themengebiet enthalten sein sollen. Dabei werden ein oder mehrere Begriffe doppelt genannt. 5. Der Benutzer klickt auf „Submit“. 6. Pop-Up erscheint, in dem nachgefragt wird, ob das neue Thema gespeichert werden soll. 7. Benutzer klickt auf „Yes“. |
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich.</p> <p>In der Liste mit den Themengebieten scheint das neue Thema auf (nach Neuladen der Seite).</p> <p>Die eingegebenen Begriffe sind dem Thema zugeordnet. Doppelte Begriffe wurden nur einmal übernommen.</p> |
| beobachtete Abweichung | Wiederholte Begriffe erscheinen mehrfach. |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.4: Themengebiet entfernen

| | |
|-----------------------------|--|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer wählt aus der Liste der Themengebiete das zu löschende Thema aus. 4. Der Benutzer klickt auf das Mülleimer-Symbol. 5. Pop-Up mit Nachfrage, ob das Thema wirklich gelöscht werden soll, erscheint. 6. Der Benutzer klickt auf „Yes“. |
| erwarteter Ergebniszustand: | Das Thema verschwindet aus der Liste der Themengebiete. |

Die Begriffe, die dem Thema angehört haben, werden ebenfalls gelöscht.

beobachtete Abweichung

Immer noch im "Select Topic" Menü bis reload.

☐ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen

☐ System unbenutzbar

TC Auth7.5: Themengebiet entfernen – aktuell verwendet

Use Case: Spieleverwalter

Ausgangszustand: Der Nutzer ist als Spieleverwalter eingeloggt.

Aktion

1. Der Benutzer klickt auf „Manager Submenu“.
2. Der Benutzer klickt auf „Topic Management“.
3. Der Benutzer wählt aus der Liste der Themengebiete das zu löschende Thema aus. Dieses Thema wird gerade in einem Spiel verwendet.
4. Der Benutzer klickt auf das Mülleimer-Symbol.
5. Pop-Up mit Nachfrage, ob das Thema wirklich gelöscht werden soll, erscheint.
6. Der Benutzer klickt auf „Yes“.

erwarteter Ergebniszustand: Das Themengebiet wird gelöscht:
Das Themengebiet kann nicht mehr neu ausgewählt werden.
Das Themengebiet wird weiterhin im laufenden Spiel verwendet.

beobachtete Abweichung Themengebiet kann weiterhin ausgewählt werden, hat aber keine terme

☐ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen

☐ System unbenutzbar

TC Auth7.6: Themengebiet – Begriffe hinzufügen

Use Case: Spieleverwalter

Ausgangszustand: Der Nutzer ist als Spieleverwalter eingeloggt.

Aktion

1. Der Benutzer klickt auf „Manager Submenu“.
2. Der Benutzer klickt auf „Topic Management“.
3. Der Benutzer wählt aus der Liste der Themengebiete das zu erweiternde Thema aus.
4. Der Benutzer klickt auf „Add New Terms“.
5. Ein Pop-Up Fenster erscheint:
 - a. Der Benutzer gibt die/den neuen Begriff(e) ein.
 - b. Der Benutzer klickt auf „Submit“.

| | |
|---|--|
| erwarteter Ergebniszustand: | Die neuen Begriffe erscheinen in der Begriffsliste des Themengebietes. |
| | Auch wenn das Themengebiet aktuell in einem laufenden Spiel verwendet wird, erscheinen die Begriffe automatisch . |
| beobachtete Abweichung | Terme erscheinen nicht im laufenden Spiel. |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.6: Themengebiet – Begriffe löschen

| | |
|---|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer wählt aus der Liste der Themengebiete bei welchem er Begriffe löschen möchte. 4. Der Benutzer sieht eine Liste mit den Begriffen, die im gewählten Themengebiet enthalten sind. 5. Der Benutzer klickt neben dem Begriff, den er bearbeiten möchte, auf das Abfalleimer-Symbol. 6. Pop-Up Fenster erscheint, ob Begriff wirklich gelöscht werden soll. 7. Der Benutzer klickt auf „Yes“. |
| erwarteter Ergebniszustand: | Das Pop-Up Fenster schließt sich. Der Begriff scheint nicht mehr in der Begriffsliste des Themengebietes auf. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.7: Themengebiet – Begriffe löschen – wird verwendet

| | |
|------------------|--|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer wählt aus der Liste der Themengebiete bei welchem er Begriffe löschen möchte. |

| | |
|---|---|
| erwarteter Ergebniszustand: | <ol style="list-style-type: none"> Der Benutzer sieht eine Liste mit den Begriffen, die im gewählten Themengebiet enthalten sind. Der Benutzer klickt neben dem Begriff, den er bearbeiten möchte, auf das Abfalleimer-Symbol. Der Begriff wird gerade gespielt. Pop-Up Fenster erscheint und der Benutzer wird gefragt, ob der Begriff wirklich gelöscht werden soll. Der Benutzer klickt auf „Yes“. <p>Das Pop-Up Fenster schließt sich. Der Begriff wird gelöscht. Der Begriff wird in folgenden Spielen nicht mehr einem Team zugeteilt. Der Begriff wird weiterhin im laufenden Spiel verwendet.</p> |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth7.8: Themengebiet – Begriffe hinzufügen

| | |
|--|--|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> Der Benutzer klickt auf „Manager Submenu“. Der Benutzer klickt auf „Topic Management“. Der Benutzer wählt aus der Liste der Themengebiete, bei welchem er Begriffe bearbeiten möchte. Der Benutzer sieht eine Liste mit den Begriffen, die im gewählten Themengebiet enthalten sind. Der Benutzer klickt neben dem Begriff, den er bearbeiten möchte, auf das Stift-Symbol. Ein Pop-Up Fenster erscheint: <ol style="list-style-type: none"> Der Benutzer nimmt die Änderungen am Begriff vor. Der Benutzer klickt auf „Submit“. |
| erwarteter Ergebniszustand: | <p>Das Pop-Up-Fenster schließt sich.</p> <p>Der Begriff scheint geändert in der Begriffsliste des Themengebiets auf.</p> |
| beobachtete Abweichung | Meldung "To display this page, Firefox must send information that will repeat any action (such as a search or order confirmation) that was performed earlier." |
| <input type="checkbox"/> OK <input checked="" type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth7.8: Themengebiet – Begriffe hinzufügen

| | |
|---|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer wählt aus der Liste der Themengebiete, bei welchem er Begriffe bearbeiten möchte. 4. Der Benutzer sieht eine Liste mit den Begriffen, die im gewählten Themengebiet enthalten sind. 5. Der Benutzer klickt neben dem Begriff, den er bearbeiten möchte, auf das Stift-Symbol. Der Begriff wird in einem laufenden Spiel verwendet. 6. Ein Pop-Up Fenster erscheint: <ol style="list-style-type: none"> a. Der Benutzer nimmt die Änderungen am Begriff vor. b. Der Benutzer klickt auf „Submit“. |
| erwarteter Ergebniszustand: | <p>Das Pop-Up-Fenster schließt sich.</p> <p>Der Begriff wird zum Ändern vorgemerkt: Der Begriff wird in den laufenden Spielen keinem Team mehr zugeteilt.</p> <p>Sobald die letzte Runde mit dem Begriff beendet wird, wird der Begriff geändert.</p> |
| beobachtete Abweichung | Begriff wurde im Spiel zugeteilt |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.9: Spielräume beobachten

| | |
|---|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „View all Games“. |
| erwarteter Ergebniszustand: | Der Benutzer sieht ein Tabelle mit allen laufenden Spielen und kann weitere Informationen zu ihnen aufrufen. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth7.10: JSON-Datei importieren

| | |
|---|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt und verfügt über eine JSON-Datei. (Anm.: eine Beispiel-Datei ist im Git abgelegt) |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer klickt unter „Upload JSON File“ auf den Button „Choose“. 4. Ein Pop-Up öffnet sich, in welchem der lokale Speicher nach der gewünschten Datei durchsucht werden kann. 5. Der Benutzer klickt auf die gewünschte JSON Datei. 6. Der Benutzer klickt auf „Öffnen“. 7. Das Pop-Up schließt sich. 8. Der Benutzer sieht das neue Thema mit den Begriffen aufgelistet. 9. Der Benutzer klickt auf „Upload“. |
| erwarteter Ergebniszustand: | Das neue Thema erscheint in der Liste. |
| beobachtete Abweichung | Das erforderliche JSON-Format ist nirgends ersichtlich. |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth7.11: JSON-Datei importieren – Topic vorhanden

| | |
|------------------|---|
| Use Case: | Spielerverwalter |
| Ausgangszustand: | Der Nutzer ist als Spielerverwalter eingeloggt und verfügt über eine JSON-Datei. (Anm.: eine Beispiel-Datei ist im Git abgelegt) |
| Aktion | <ol style="list-style-type: none"> 1. Der Benutzer klickt auf „Manager Submenu“. 2. Der Benutzer klickt auf „Topic Management“. 3. Der Benutzer klickt unter „Upload JSON File“ auf den Button „Choose“. 4. Ein Pop-Up öffnet sich, in welchem der lokale Speicher nach der gewünschten Datei durchsucht werden kann. 5. Der Benutzer klickt auf die gewünschte JSON Datei. 6. Der Benutzer klickt auf „Öffnen“. 7. Das Pop-Up schließt sich. 8. Der Benutzer sieht das neue Thema mit den Begriffen aufgelistet. |

| | |
|--|--------------------------------------|
| | 9. Der Benutzer klickt auf „Upload“. |
| erwarteter Ergebniszustand: | Es erscheint eine Fehlermeldung. |
| beobachtete Abweichung | keine Fehlermeldung |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth8.1: User erstellen

| | |
|--|---|
| Use Case: | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt auf „Create Player“ 5. Ein Pop-Up Fenster erscheint <ol style="list-style-type: none"> a. Der Admin gibt einen Username für den neuen User ein b. Der Admin gibt ein Passwort für den neuen User ein c. Der Admin gibt den Vornamen für den neuen User ein d. Der Admin gibt den Nachnamen für den neuen User ein e. Der Admin gibt eine Email-Adresse für den neuen User ein f. Der Admin klickt an, ob der User enabled sein soll g. Der Admin definiert die Rolle des neuen Users indem er auf „Save as simple player“ oder „Save as game manager“ klickt 6. Ein neues Pop-Up erscheint, mit der Nachricht, dass der User erfolgreich gespeichert wurde |
| erwarteter Ergebniszustand: | <p>Nach Schließen des Pop-Up Fensters sieht der Admin die Liste der registrierten User.</p> <p>Der neue User scheint in der User-Liste auf.</p> <p>Der neu erstellte User kann sich ab jetzt mit Nutzernamen und Passwort einloggen.</p> |
| beobachtete Abweichung | Grammatikfehler in Titel, User müssen danach noch manuell enabled werden |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth8.2: User erstellen - unvollständig

| | |
|---|--|
| Use Case: | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt auf „Create Player“ 5. Ein Pop-Up Fenster erscheint, es werden nicht alle Felder ausgefüllt. <ol style="list-style-type: none"> a. Der Admin gibt einen Username für den neuen User ein b. Der Admin gibt ein Passwort für den neuen User ein c. Der Admin gibt den Vornamen für den neuen User ein d. Der Admin gibt den Nachnamen für den neuen User ein e. Der Admin gibt eine E-Mail-Adresse für den neuen User ein f. Der Admin klickt an, ob der User enabled sein soll g. Der Admin definiert die Rolle des neuen Users indem er auf „Save as simple player“ oder „Save as game manager“ klickt |
| erwarteter Ergebniszustand: | Es erscheint eine Fehlermeldung mit dem Hinweis, die Angaben zu vervollständigen. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth8.3: User erstellen – Username vergeben

| | |
|------------------|---|
| Use Case: | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt auf „Create Player“ 5. Ein Pop-Up Fenster erscheint. <ol style="list-style-type: none"> a. Der Admin gibt einen Username für den neuen User ein. Der Username ist bereits vergeben. b. Der Admin gibt ein Passwort für den neuen User ein c. Der Admin gibt den Vornamen für den neuen User ein |

| | |
|--|--|
| | <ul style="list-style-type: none"> d. Der Admin gibt den Nachnamen für den neuen User ein e. Der Admin gibt eine Email-Adresse für den neuen User ein f. Der Admin klickt an, ob der User enabled sein soll g. Der Admin definiert die Rolle des neuen Users indem er auf „Save as simple player“ oder „Save as game manager“ klickt |
| erwarteter Ergebniszustand: | Es erscheint eine Fehlermeldung mit dem Hinweis, dass der Name bereits vergeben ist. |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth8.4: User bearbeiten

| | |
|--|---|
| Use Case: | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ul style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt neben dem User den er bearbeiten möchte auf das Edit-Symbol 5. Ein Pop-Up Fenster erscheint <ul style="list-style-type: none"> a. Der Admin kann den Vornamen des Users ändern b. Der Admin kann den Nachnamen des Users ändern c. Der Admin kann eine andere Email-Adresse eingeben d. Der Admin kann auf „Enabled“ klicken e. Der Admin klickt auf „Save“ |
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich.</p> <p>Der Admin sieht die Ansicht „User-Verwaltung“.</p> <p>Der User scheint in der User-Liste mit vorgenommenen Änderungen auf.</p> |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen | |
| <input type="checkbox"/> System unbenutzbar | |

TC Auth8.5: User bearbeiten – User eingeloggt

| | |
|-----------|---------------|
| Use Case: | Administrator |
|-----------|---------------|

| | |
|---|--|
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt neben dem User den er bearbeiten möchte auf das Edit-Symbol. Der User ist momentan eingeloggt. 5. Ein Pop-Up Fenster erscheint <ol style="list-style-type: none"> a. Der Admin kann den Vornamen des Users ändern b. Der Admin kann den Nachnamen des Users ändern c. Der Admin kann eine andere Email-Adresse eingeben d. Der Admin kann auf „Enabled“ klicken e. Der Admin klickt auf „Save“ |
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich.</p> <p>Der Admin sieht die Ansicht „User-Verwaltung“.</p> <p>Die Änderungen werden vorgemerkt und durchgeführt, sobald der User ausgeloggt ist.</p> |
| beobachtete Abweichung | |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

TC Auth8.6: User bearbeiten – Felder bleiben leer

| | |
|-----------------------------|---|
| Use Case | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Submenu“ 2. Der Admin klickt auf „Manage Players“ 3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“) 4. Der Admin klickt neben dem User den er bearbeiten möchte auf das Edit-Symbol 5. Ein Pop-Up Fenster erscheint <ol style="list-style-type: none"> a. Der Admin kann den Vornamen des Users ändern b. Der Admin kann den Nachnamen des Users ändern c. Der Admin kann eine andere Email-Adresse eingeben d. Der Admin kann auf „Enabled“ klicken e. Der Admin klickt auf „Save“ |
| erwarteter Ergebniszustand: | Es kommt eine Fehlermeldung, dass einige Felder leer sind. |

beobachtete Abweichung

Führt man den Testfall wie angegeben aus tritt der erwartete Ergebniszustand nicht ein, da unter Aktionen nur von Ändern, nicht von Ausleeren die Rede ist.

☐ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen

☐ System unbenutzbar

TC Auth8.7: User löschen

Use Case

Administrator

Ausgangszustand:

Der Nutzer ist als Admin eingeloggt.

Aktion

1. Der Admin klickt auf die Registerkarte „Admin Submenu“
2. Der Admin klickt auf „Manage Players“
3. Der Admin sieht die Ansicht User-Verwaltung
4. Der Admin klickt neben dem User, den er löschen möchte, auf das Abfalleimer-Symbol
5. Ein Pop-Up Fenster erscheint, ob der User wirklich gelöscht werden soll
6. Der Admin klickt auf „Yes“

erwarteter Ergebniszustand:

Das Pop-Up Fenster schließt sich.
Der Admin sieht die Ansicht „User-Verwaltung“.
Der User scheint nicht mehr in der User-Liste auf.
Der gelöschte User kann sich nicht mehr anmelden.

beobachtete Abweichung

Wenn ein User nicht gelöscht werden kann passiert nichts, es wird auch keine Meldung angezeigt.

☐ OK ☐ kosmetische Abweichungen ☐ mittlere Abweichungen ☐ große Abweichungen

☐ System unbenutzbar

TC Auth8.7: User löschen - eingeloggt

Use Case

Administrator

Ausgangszustand:

Der Nutzer ist als Admin eingeloggt.

Aktion

1. Der Admin klickt auf die Registerkarte „Admin Submenu“
2. Der Admin klickt auf „Manage Players“
3. Der Admin sieht die Ansicht User-Verwaltung (Liste der registrierten User + Button „Create Player“)
4. Der Admin klickt neben dem User, den er löschen möchte, auf das Abfalleimer-Symbol
5. Ein Pop-Up Fenster erscheint, ob der User wirklich gelöscht werden soll
6. Der Admin klickt auf „Yes“

| | |
|---|--|
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich. Der Admin sieht die Ansicht „User-Verwaltung“. Der User scheint nicht mehr in der User-Liste auf. Der gelöschte User kann sich nicht mehr anmelden. Der User hat keine Berechtigungen mehr.</p> |
| beobachtete Abweichung | <p>Wenn der User nicht gelöscht werden kann passiert nichts, es wird auch keine Meldung angezeigt.</p> |
| <input type="checkbox"/> OK <input type="checkbox"/> kosmetische Abweichungen <input type="checkbox"/> mittlere Abweichungen <input type="checkbox"/> große Abweichungen <input type="checkbox"/> System unbenutzbar | |

| TC Auth8.8: Spielraum löschen | |
|---|--|
| Use Case | Administrator |
| Ausgangszustand: | Der Nutzer ist als Admin eingeloggt. |
| Aktion | <ol style="list-style-type: none"> 1. Der Admin klickt auf die Registerkarte „Admin Subemenu“ und dort auf „Delete games“. 2. Der Admin sieht eine Liste mit den aktuellen Spielerräumen mit jeweils einem Abfalleimer-Symbol 3. Der Admin klickt neben dem Spielerraum, den er löschen möchte auf das Abfalleimer-Symbol 4. Ein Pop-Up Fenster erscheint, ob der Spielerraum wirklich gelöscht werden soll 5. Der Admin klickt auf „löschen“ |
| erwarteter Ergebniszustand: | <p>Das Pop-Up Fenster schließt sich Der Admin sieht die Spielelobby Der Spielerraum scheint noch so lange in der Liste auf, bis alle Spieler den Spielende Bildschirm verlassen haben Die Spieler, die sich im Raum befunden haben, wurden herausgeworfen Das Spiel, das im gelöschten Spielerraum stattgefunden hat, wird nicht zu den Statistiken der Spieler gezählt</p> |
| beobachtete Abweichung | <p>Admin sieht nicht Spielelobby nach Löschen des Spiels, Spielerraum bleibt nicht in Spielelobby in der Liste (in „Delete Games“ schon)</p> |
| <input 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. Weitere nichtfunktionale Testfälle

- Das Programm läuft auch bei der Verwendung von mehreren RaspberryPi/TimeFlip-Würfeln.

- Es kann auf die Nutzeroberfläche zugegriffen werden, obwohl ein Spiel läuft.
- Ausreichend Daten zur Testung sind vorhanden.

4. Anhang

4.1. Referenzierte Dokument

- Konzeptbeschreibung (13.05.2021)