

Systemtest

Team: Team 1

Mitglied 1: (Michael Hauser ,01267565)

Mitglied 2: (Angela Todhri, 11815296)

Mitglied 3: (Ismail Üner, 11721981)

Mitglied 4: (Flaminia Anselmi, 11934695)

Mitglied 5: (Sebastian Hepp, 01015083)

Mitglied 6: (Maximilian Heine, 01317323)

Proseminargruppe: Gruppe 6

Datum: 18.06.2021

Hinweis: Dieses Dokument wird auf Basis des Testprotokolls (=ausgefüllte Version der Liste der Testfälle) erstellt.

1. Ergebnis

Der Systemtest wurde auf Basis des Testdrehbuchs mit Stand 16.06.2021 mit 49 Testfällen und 2 nicht funktionalen Tests durchgeführt. Bei Durchführung dieser Tests kam es zu keiner kritischen oder mittleren Abweichungen – somit erfüllt es die Aufgabenstellung sowie unser selbst erarbeitetes Softwarekonzept mit den dazu gelisteten Use-Cases.

2. Funktionalität

Die Testfälle wurden für alle Benutzergruppen erstellt, wobei diese für die Gruppen *Game Manager* sowie *Admin* erweitert wurden. Jegliche Benutzung in Hinblick auf Erstellung und Veränderung, einschließlich Löschen, wurde zur Gänze abgedeckt. Dabei wurde auch auf mögliche Fehlbedienung und invalide Eingaben geachtet. Die Use-Cases geben das Grundgerüst für die Testfälle an, wobei für manche Use-Cases bereits durch einen Testfall die Prüfung der Funktionalität gedeckt wurde, gab es andere welche iterativ abgearbeitet wurden.

Begonnen wurde mit dem Use-Case 1 „*Registrieren eines Users*“. Dabei wurden im Softwarekonzept Annahmen über erfolgreichen und nicht erfolgreichen Zustand getroffen. Sollten nicht zulässige Werte durch den Benutzer oder das System entstehen, wird durch Fehlermeldungen in der GUI über das auftretende Problem informiert.

In manchen Fällen war das Testen repetitiv, da das Erzeugen von einfachen Objekten immer in dasselbe Prinzip erfolgt, dennoch wurden so fehlende Eingabekontrollen, Fehlermeldungen oder Benutzungshinweise entdeckt. Im Falle des Use-Cases „*View Player Profile*“ wird eine Möglichkeit angeboten, für einen Benutzer seinen Raspberry zu ändern. Dazu wird vom Benutzer die IP-Adresse des Raspberries abgefragt, was beispielhaft für eine der Sicherheitsmaßnahmen des Systems ist. Ein besonders übergreifender Use-Case, welcher weitere vorhergehende Use-Cases miteinschließt ist der „*Create Game*“, da dabei *Users* und *Topics* mit *Terms* miteingeschlossen werden. Dieser Fall bildet die Vorbedingung für die Fälle des Erstellens der Teams eines Spieles. Danach gelangt der Nutzer zum eigentlichen Gameplay, bei dem das Würfeln mit dem *Timeflip* integriert ist. Hier ist zudem die Ausgabe der zu erratenden Begriffe an die richtigen Teams, sowie die Punktevergabe zu testen.

3. Performanz, Fehlertoleranz und Stabilität

Da keine spezifische Vorgabe in Bezug auf das Antwortverhalten mit dem Auftraggeber vereinbart wurde, entspricht die Performanz den derzeitigen Standards. Nach Überprüfung aller Testfälle, konnten keine spürbaren Verzögerungen für den Nutzer festgestellt werden – somit ist das Antwortverhalten angemessen.

Sollte der Würfel die Verbindung zum Raspberry verlieren, muss das System am Raspberry neu gestartet werden. Der Raspberry zeigt dabei an, ob eine Verbindung zu der Webanwendung aufgebaut werden konnte und ob eine Übertragung durch REST erfolgreich war oder nicht. Sollte während dem Spiel die Verbindung zur Webanwendung unterbrochen werden, beispielsweise durch unabsichtliches Schließen des Browsers, haben die Spieler eines Spieles die Möglichkeit durch die Schaltfläche „*Join Game*“ dem derzeit laufenden Spiel wieder beizutreten.

Zum Thema Fehlerbehandlung,

Wo der Benutzer durch Eingaben mit dem System interagieren kann, wurde in gewissen Fällen bereits im Frontend dafür gesorgt, dass nur valide Daten zur Verarbeitung im Backend weitergeleitet werden. Bei inkorrekten Eingaben werden entsprechende Fehlermeldung ausgegeben, die den Nutzer auf die erlaubten bzw. benötigten Eingaben hinweisen. In anderen Fällen wurden im Backend

eigene Validierungs-Methoden erstellt, die im Falle eines Fehlers ähnliche Fehlermeldungen ausgeben.

Während dem Spielbeitritt durch „Join Game“, wenn der Nutzer die Teammitspieler auswählt, besteht die Möglichkeit die Auswahl durch „reset“ für alle Teams zurückzusetzen, falls eine falsche Auswahl vorgenommen wurde oder es zu einem anders begründeten unerwarteten Zustand gekommen ist.

Beim Löschen eines Users, darf dieser nicht eingeloggt sein. Um fälschliches, versehentliches Löschen zu erschweren, muss zudem dieser vorher auf den Status „disabled“ gesetzt worden sein. Das Löschen ist ein schwerer und endgültiger Eingriff, da dadurch die zugehörigen Statistiken mitgelöscht werden und sich andere Statistiken dadurch ändern können. Der Eingriff kann somit auch für andere Nutzer sichtbar sein.

4. Datenschutz

Da in der Applikation zwischen drei Benutzergruppen unterschieden wird, wird das von Spring implementierte Security Konzept eingesetzt, mit welchem im Backend nur privilegierten Benutzern der Zugriff auf bestimmte Ressourcen gewährt wird. Des Weiteren werden dem Nutzer im Frontend nur die Elemente angezeigt, welche für dessen Nutzergruppe verfügbar sind.

Hinsichtlich Verschlüsselung des Passworts wurde die von Spring vorimplementierte Funktion „*BcryptPasswordEncoder*“ gewählt, damit keine Passwörter im Klartext in der Datenbank gespeichert werden.

Für die REST-Kommunikation wurde eine eigene Authentifizierung implementiert, welche eine Verbindung nur zulässt, wenn der Nutzer den richtigen API-Key besitzt.

5. Usability

Den Anforderungen entsprechend wurde das System möglichst selbsterklärend implementiert. Besonders die Einfachheit für einen Nutzer mit der einfachsten Autorisierung „*Player*“ wurde berücksichtigt. Dieser kann nur Statistiken in der Lobby und Spielerprofile einsehen, sein eigenes bearbeiten und einem bereits erstellten Spiel. Es wurde angenommen, dass nur der Produktkäufer die Rolle „*Game Manager*“ besitzt. Durch diese Rolle erhält der Nutzer das Privileg Spiele zu erstellen. Er erhält außerdem mehr Möglichkeiten für Modifikationen oder Einsichten, wie zum Beispiel die Verwaltung der Raspberries, mit welcher sich nur der Besitzer auskennen muss.

In den einigen Bereichen der GUI werden Schaltflächen angeboten, welche eine Hilfestellung zur Funktionalität und Verwendung des aktuellen Bereiches anzeigen. Beim Beispiel des Bereichs der Raspberry-Verwaltung werden hier Informationen zum Hinzufügen eines Raspberries und zur Bedeutung von IP-Adresse und API-Key gegeben.

6. Weitere Auffälligkeiten

In einigen Bereichen kann die Funktionalität noch weiter ausgebaut werden. So wäre das Löschen mehrerer Terms gleichzeitig nach Auswahl durch beispielsweise Checkboxes eine Erleichterung. Der Manager Hub könnte durch verschiedenste (eventuell benutzerdefinierte) Auswertungen mehr Möglichkeiten für Game Manager bieten. Manche seltenen Fehler beim Beitritt eines Spiels konnten nicht ausgeschlossen werden, da die Bedingungen dafür relativ komplex sind. Das Gameplay könnte noch aufwendiger gestaltet werden. Das Konfigurieren des Timeflips, das zur Zeit über ein Kommandozeilen-Skript durchgeführt wird, könnte durch eine Funktion innerhalb der GUI der Webapplikation aufgewertet werden.