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.

# Ergebnis

*Fassen Sie das Ergebnis Ihres Systemtests zusammen – wann wurde der Test mit wie vielen Testfällen durchgeführt, wie viele Fehler wurden in welcher Kritikalität gefunden?*

Der Systemtest wurde mit dem Stand vom Testdrehbuch 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 Usecases.

# Funktionalität

* *Welche Überdeckung weisen Ihre Testfälle auf? – Setzen Sie dazu Ihre Testfälle mit den Use Cases in Beziehung.*
* *Sind Schnittstellen zu anderen Systemen mit Testfällen abgedeckt? Ergebnis?*

Die Testfälle wurden für alle Benutzergruppen erstellt, wobei diese für die Gruppen Manager sowie Administrator erweitert wurden. Jegliche Benutzung in Hinblick auf Erstellung und Veränderung, einschließlich löschen, wurde zur Gänze abgedeckt. Dabei wurde auch auf mögliche Fehlbedienung und Invalide Daten eingaben geachtet. Unsere Usecases gaben das Grundgerüst für unsere Testcases an und somit gibt es hierbei starke Überschneidungen, wobei wir für manche UseCases bereits mit einem Testfall die Prüfung der Funktionalität gedeckt haben, gab es andere welche Iterativ Schritt für Schritt abgearbeitet wurden.  
Beginnend mit unserem ersten UseCase vom Registrieren eines Users haben wir auch in unserem Systemtest mit diesem begonnen. Dabei wurden schon vorab in unserem Konzept Annahmen über einen erfolgreichen Zustand oder widerrum über einen nicht erfolgreichen Zustand getroffen. Sollte es zu nicht zulässigen Werten vom Benuter oder vom System kommen, weißen wir mittels Fehlermeldungen in der GUI, über das hierbei auftretende Problem auf.  
In manchen Fällen wirkte das Testen hinweg redundant, da das erzeugen von einfachen Objekten immer das selbe Prinzip verfolgte, aber trotzdem immer noch hilfreich war, da man so wieder miterlebte wie man bei manchen Objekten vergaß abzufragen ob der Input Valide ist.  
Im Falle des UseCases View Player Profile wird eine Möglichkeit angeboten, dass ein Benutzer seinen Raspberry ändert, hierbei funktioniert dies aber nur, wenn der Benutzer die IP Adresse des Raspberries kennt, was einen starken Zusammenhang aufzeigen soll wie das System miteinander arbeitet und welche Sicherheitsmaßnahmen getroffen wurden.   
Ein besonders übergreifender UseCase, welcher weitere zuvorstehende UseCases miteinschließt ist der Fall Create Game, da man dabei Users und Topics mit Terms miteinschließt. Dieser Fall bildet die Vorbedingung für die Fälle des erstellen eines Teams in einem Spiel, wonach man dann zum wirklichen Gameplay gelangt, welches dass würfeln des Timeflips integriert, die Ausgabe der zu erratenden Begriffe an die richtigen Teams steuert, sowie die Punktevergabe miteinschließt.

# Performanz, Fehlertoleranz und Stabilität

*Prüfen Sie zumindest folgende Aspekte:*

* *Ist das Antwortverhalten des Systems im vereinbarten Rahmen?*
* *Reagiert das System in angemessener Weise auf Fehlerzustände (z.B. Neustart des Minirechners, kurzfristiger Ausfall der Kommunikation zwischen Minirechner und zentralem Backend, etc.)?*
* *Reagiert Ihr System stabil auf fehlerhafte und attackierende Eingaben?*
* *Ist sichergestellt, dass gelöschte/inaktive Benutzeraccounts keine Auswirkungen auf vergangene Berichtszeiträume haben?*

Da keine spezifische Vorgabe im Bezug auf das Antworte verhalten mit dem Auftraggeber vereinbart war, wurde es nach unserem eigenem bemessen und den derzeitigen Standards festgesetzt und wir kamen nach Überprüfung aller Testfälle zu dem Ergebnis, dass unser System zu keinem Zeitpunkt dem User das gefühlt gibt zu langsam zu sein – somit ist das Antwortverhalten in einem derzeit völlig legitimen Rahmen.

Sollte der Würfel die Verbindung zum Raspberry verlieren, muss das System am Rasberry nocheinmal neu gestartet werden. Auf Seiten des Raspberry sieht man ob eine übertragung durch die REST erfolgreich geschehn ist oder nicht, sowie ob überhaupt eine Verbindung zu einer Webanwendung etabliert wurde. Sollte es während dem Spiel passieren das man die verbindung zur Webanwendung verliert, beispielsweise durch unabsichtliches schließen des Browsers, bieten wir es den beteiligten Spielern weiterhin an durch das klicken des Knopf „Join Game“ sich mit dem derzeit laufenden Spiel zu verbinden, insofern man für dieses Spiel eingeschrieben ist.

Zum Thema Fehlerbehandlung, wo der Benutzer mit dem System interagieren kann, sprich Daten erzeugen kann, wurde in gewissen Fällen bereits im Frontend dafür gesorgt das nur Valide Daten verarbeitet werden dürfen, wenn nicht wird dem Benutzer eine entsprechende Fehlermeldung gegeben, damit er seinen Fehler beheben kann. In anderen Fällen wurden im Backend eigens dafür geschriebene Validatoren erzeugt die im Falle eines Fehlers dem Benutzer auch wieder eine Information geben warum es dazu kam.  
Im Falle des erzeugen eines Teams geben wir dem Benutzer auch die Möglichkeit, das Team vollkommen zu reseten, sollte es zu einem unerwarteten Zustand gelangen.

Wir haben uns im Hinblick darauf einen User zu löschen geeinigt, dass dieser nicht eingelogt sein darf und weiters auch noch auf den Status disabled gesetzt werden muss um wirklich zu zeigen, dass es ein schwerwiegender eingriff in das System ist, da es Möglichkeiten in den Statistiken gibt wo dieser Spieler beteiligt ist und diese unwiderruflich gelöscht werden.

# Datenschutz

*Prüfen Sie zumindest folgende Aspekte:*

* *Sind Zugriffsrechte korrekt umgesetzt?*
* *Ist sichergestellt, dass spezifische Auswertungen und Auswertungszeiträume nur den dafür vorgesehenen Benutzergruppen zur Verfügung stehen?*

Da wir in Unserer App zwischen 3 Benutzergruppen unterscheiden, setzten wir das von Spring implementierte Security Konzept ein, mit welchem wir im Backend nur Priviligierten Benutzern den Zugriff auf bestimmte Ressourcen gewähren, sollte man diese nicht besitzen ist der Zugriff darauf nicht möglich. Des weiteren haben wir für die verschiedenen Gruppen auch das Frontend so angepasst, dass man auch nur die Elemente sieht welche man auch benutzen kann, da es ein trauriges gefühl vermittelt, 7 Seiten anzubieten, wobei man aber nur 3 benutzen kann, da auf die anderen 4 der Zugriff verweigert wird und einem nur eine Fehlermeldung gezeigt wird „Access Denied“.  
In hinsicht auf Kryptographie des Passworts entschieden wir uns für die von Spring vorimplementierte Funktion „BcryptPasswordEncoder“ um keine Klartext Passwörter in unserer Datenbank zu speichern.

Für unsere REST Kommunikation, schrieben wir eine eigene Authentifizierung, welche eine Verbindung nur zulässt sollte man den richtigen API-Key besitzen.

# Usability

*Wird das Gesamtkonzept des Systems den Bedürfnissen der adressierten Zielgruppe gerecht?*

Den Anforderungen entsprechend wurde das System stark selbsterklärend implementiert, besonders die Einfachheit für einen einfachen Benutzer – Spieler wurde Berücksichtigt, da dieser nur Statistiken in der Lobby, sowie seine eigenen einsehen kann und darüber hinaus nur noch einem Spiel beitreten kann und dieses nicht selbst erzeugen muss. Unsere Annahme dazu war, das der Softwareeigner – Manager das Privileg besitzt Spiele zu erzeugen und damit mehr Möglichkeiten besitzt Modifiaktionen vorzunehmen, was im großen und Ganzen auch ein einfacher Prozess ist, insofern man es 1 – 2 mal wiederholt hat. In den möglicherweise Kritischen Regionen wurden des weiteren auch noch Buttons hinzugefügt, welche einem nocheinmal erklären wie das System zusammen spielt und wie man es bedient. Im Beispiel des Raspberry und Timeflip, ein Leihe wird sich hierbei fragen warum muss er hierbei einen API-Key festlegen oder warum muss da etwas weiteres Konfiguriert werden, wenn man doch nur spielen möchte, somit wurde eine solche Konfiguration nur dem Manager und dem Admin eingeräumt, weil diese auch die Besitzer des Raspberrys und des Timeflip sein werden.

# Weitere Auffälligkeiten

*Führen Sie hier weitere Auffälligkeiten und noch zu leistende Arbeiten an.*