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 hierbei passendes Beispiel um zu zeigen wie die Modelle zusammenarbeiten wird im

# 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?*

# 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?*

# Usability

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

# Weitere Auffälligkeiten

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