



**TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN**

AG Computergrafik & HCI
apl. Prof. Dr. Achim Ebert

SEP/MP 2020

Exploding Kittens

Pflichtenheft

5. Mai 2020

Gruppe 6

Siyu Zhang

Yirui Fu

Manuela Kuaguim

Florent Tandjune Tamoyem

Thiemo Lachmund

Alexander Nachtigall

Inhaltsverzeichnis

Inhaltsverzeichnis	2
1 Projekttreiber	3
1.1 Projektziel	3
1.2 Stakeholders	3
1.3 Aktuelle Lage	4
2 Projektbeschränkungen	5
2.1 Beschränkungen	5
2.2 Glossar	6
2.3 Relevante Fakten und Annahmen	7
3 Funktionale Anforderungen	9
3.1 Systemfunktionen	9
3.2 Systemgrenze (Use Case Diagramm)	11
3.3 Beschreibungen der Anwendungsfälle	11
3.4 Produktdaten	16
4 Nicht-funktionale Anforderungen	17
4.1 Softwarearchitektur	17
4.2 Benutzerfreundlichkeit	17
4.3 Leistungsanforderungen	18
4.4 Anforderungen an Einsatzkontext	18
4.5 Anforderungen an Wartung und Unterstützung	19
4.6 Sicherheitsanforderungen	20
4.7 Prüfungsbezogene Anforderungen	21
4.8 Kulturelle und politische Anforderungen	21
4.9 Rechtliche und standardsbezogene Anforderungen	21
5 Systemtestfälle	22
6 Warteraum	27

Kapitel 1

Projekttreiber

1.1 Projektziel

Im Rahmen des Software-Entwicklungs-Projekts/Modellierungspraktikums 2020 soll ein einfach zu bedienendes Client-Server-System zum Spielen von Exploding Kittens über ein Netzwerk implementiert werden. Die Benutzeroberfläche soll intuitiv bedienbar sein.

1.2 Stakeholders

/SH10/ **Name:** Spieler

Beschreibung: Menschliche Spieler.

Ziele/Aufgaben: Das Spiel zu spielen.

/SH20/ **Name:** Eltern

Beschreibung: Eltern minderjähriger Spieler.

Ziele/Aufgaben: Wollen Spielzeit ihrer Kinder begrenzen und Zugriff auf sensible Inhalte kontrollieren.

/SH30/ **Name:** Gesetzgeber

Beschreibung: Das Amt für Jugend und Familie.

Ziele/Aufgaben: Die Rechte der Spieler schützen und gewähren indem er Gesetze erstellt.

/SH40/ **Name:** Investoren (nur für Beispielszwecken)

Beschreibung: Parteien, die das Finanzmittel für die Entwicklung des Systems bereitstellen.

Ziele/Aufgaben: Gewinn zu erzielen, indem das System an Endverbraucher verkauft wird.

/SH50/ Name: Betreuer

Beschreibung: HiWis, die SEP/MP Projektgruppen betreuen.

Ziele/Aufgaben: Die Spielentwicklung durch die Studenten betreuen

/SH60/ Name: Prof. Dr. Achim Ebert

Beschreibung: Leiter des Software-Entwicklungsprojekt

Ziele/Aufgaben: Leitung der Lehrveranstaltung

1.3 Aktuelle Lage

Aktuell gibt es nur eine physische Form des Spiels und eine App für mobile Endgeräte. Da es zur Zeit keine Version des Spiels für Desktop-Computer gibt entsteht die Problematik, dass ein potenzieller Spieler ohne ein entsprechendes Handy keine Möglichkeit besitzt am Spiel teilzunehmen. Die physischen Spielkarten weisen außerdem das Problem auf, dass sie sich bei häufigem Gebrauch abnutzen oder kaputt gehen.

Dieses Projekt soll solchen Spielern ermöglichen ohne Handy online miteinander zu spielen und einer überhöhten Kartenabnutzung vorzubeugen. Es leistet somit auch einen Beitrag zur Müllvermeidung.

Eltern können das Spielverhalten ihrer Kinder in dieser Version genauer kontrollieren als bei den oben Genannten Varianten.

Investoren profitieren durch die Spielverkäufe an so neu gewonnene Spieler.

Kapitel 2

Projektbeschränkungen

2.1 Beschränkungen

/LB10/ **Name:** Selbstlehrende Bots

Beschreibung: Keine Selbstlehrfunktion von Bots wird implementiert.

Motivation: Die Funktionalität ist zu aufwändig zu implementieren und passt deshalb nicht in das Zeitbudget.

Erfüllungskriterium: Intelligenzalgorithmus von Bots ist so vorprogrammiert, dass sie Entscheidungen nur anhand des vorprogrammierten Wissens sowie des aktuellen Spielstands treffen, ohne dabei frühere Spiele zu berücksichtigen.

/LB20/ **Name:** Anwendungsbereich

Beschreibung: Das System ist ausschließlich für den privaten Bereich ausgelegt.

Motivation: Firmen oder kommerzielle Anwender haben wenig Verwendung für ein digitales Kartenspiel.

Erfüllungskriterium:

/LB30/ **Name:** Implementierungssprache

Beschreibung: Für die Implementierung ist ausschließlich Java 8 oder höher zu verwenden.

Motivation: Das optimiert die Betreuung von SEP/MP und koordiniert die Mitarbeit.

Erfüllungskriterium:

/LB40/ **Name:** GUI-Framework

Beschreibung: Die GUI ist mit JavaFX zu realisieren.

Motivation: Das optimiert die Betreuung von SEP/MP und koordiniert die Mitarbeit.

Erfüllungskriterium:

/LB50/ **Name:** Gitlab

Beschreibung: Für die Entwicklung ist das vorgegebene GitLab-Repository zu verwenden.

Motivation: Das optimiert die Betreuung von SEP/MP und koordiniert die Mitarbeit.

Erfüllungskriterium:

2.2 Glossar

Deutsch	Englisch	Bedeutung
Beispiele	Examples	Beispiele aus dem SEP letzter Jahren, welche angepasst werden müssen.
Explosion	explosion	Teilnehmer, die explodieren verlieren.
Karte	card	Spielkarte, die in einem Zug gespielt werden kann
Entschärfung	disable	Mit dieser Karte kann sich ein Spieler vor einer Explosion retten
Exploding Kitten	Exploding Kitten	Wer diese Karte zieht explodiert
Nö!	no!	Andere Karte außer Kraft setzen
Angriff	attack	Zug beenden ohne eine Karte zu ziehen, nächster Spieler muss zwei Karten ziehen
Hops!	Jump!	Zug beenden ohne Karte zu ziehen
Wunsch	wish	Mitspieler zwingen eine Karte rauszugeben
Mischen	mix	Spielstapel neu mischen
Blick in die Zukunft	View to the future	Oberste drei Karten des Spielstapels geheim ansehen

Deutsch	Englisch	Bedeutung
Katzen-Karten	Kitten Cards	Als Pärchen gespielt eine zufällige Karte vom Mitspieler stehlen, als Drilling Wunschkarte stehlen, als Fünfling beliebige Karte vom Ablagestapel nehmen
Passen	pass	Keine Karten ausspielen
Spielzug	move	Eine oder mehrere Karten offen auf den Stapel legen und Anweisungen befolgen.
Bot	bot	Spieler, dessen Spielaktionen vom Computer entschieden und durchgeführt werden
Kekse	Cookies	Offiziell keine gültige Maßnahme zur Bestechung der HiWis
Lobby	lobby	Virtueller Raum zum Betreten eines Spielraums
Spiel (Regelwerk)	game	Exploding Kittens
Spieler	player	Teilnehmer am Spielgeschehen
Spielraum	game room	Virtueller Raum, in dem ein Spiel stattfindet
Zug	turn	Zustand in dem ein Spieler eine Spielaktion ausführen muss

2.3 Relevante Fakten und Annahmen

Wichtige gekannte Fakten und getroffene Annahmen, die sich auf das Projekt direkt oder indirekt beziehen und dadurch auf die zukünftige Implementierungsentscheidungen Effekt haben können.

/FA10/ Name: Keine Fortentwicklung des App nach SEP/MP.

Beschreibung: Nach Ende des SEP/MP wird das Projekt nicht weiterentwickelt.

Motivation: Das Entwicklungsteam hat keine Lust darauf.

/FA20/ Name: Keine Lizenzen für Spielartefakte.

Beschreibung: Weder die TU Kaiserslautern noch das Spielwerk + die Freizeit GmbH gewahren dem Entwicklungsteam die Rechte für die Spielartefakte.

Motivation: Rechtliche Vorsorge.

/FA30/ **Name:** Keine bekannte Nachteile von Verwendung von Spielartefakten.

Beschreibung: Es ist nicht bekannt, dass die SEP/MP-Teilnehmer der letzten Jahre irgendwelche rechtlichen Probleme dadurch gehabt haben, dass sie die Spielartefakten vom Spielwerk + Freizeit GmbH im Rahmen ihrer SEP/MP eingesetzt haben.

Motivation: Rechtliche Vorsorge.

Kapitel 3

Funktionale Anforderungen

3.1 Systemfunktionen

/LF10/ **Name:** Spielverwaltung

Beschreibung: Das System verwaltet von mehreren Spielern geteilte Spiele in einem Spielraum. Spiele erfolgen nach den Spielregeln.

/LF20/ **Name:** Accountverwaltung

Beschreibung: Das System verwaltet die Accounts der Spieler in der Datenbank. Spieler können ihre Benutzernamen und Passwörter ändern.

/LF30/ **Name:** Zugriffsverwaltung

Beschreibung: Das System verwaltet den Zugang zum Spiel anhand von Benutzerdaten. Spieler können sich registrieren, anmelden, abmelden sowie ihre Kontos löschen.

/LF40/ **Name:** Verwaltung der Spielräume

Beschreibung: Das System verwaltet die Erstellung, Änderung und Löschung der Spielräume.

/LF50/ **Name:** Bestenliste

Beschreibung: Die Anzahl der gewonnenen Spiele aller Spieler anzeigen.

/LF60/ **Name:** Intelligente Bots

Beschreibung: Vom System gesteuerte Spieler, die möglichst gewinnbringende Strategien verfolgen. Eine leichtere und eine komplexere Version stehen zur Auswahl und können bei der Erstellung des Spielraums hinzugefügt werden.

/LF70/ **Name:** Chat

Beschreibung: Möglichkeit für Spieler in der Lobby miteinander zu kommunizieren und Spielpartner zu finden. Zusätzlicher Chat für Spielräume.

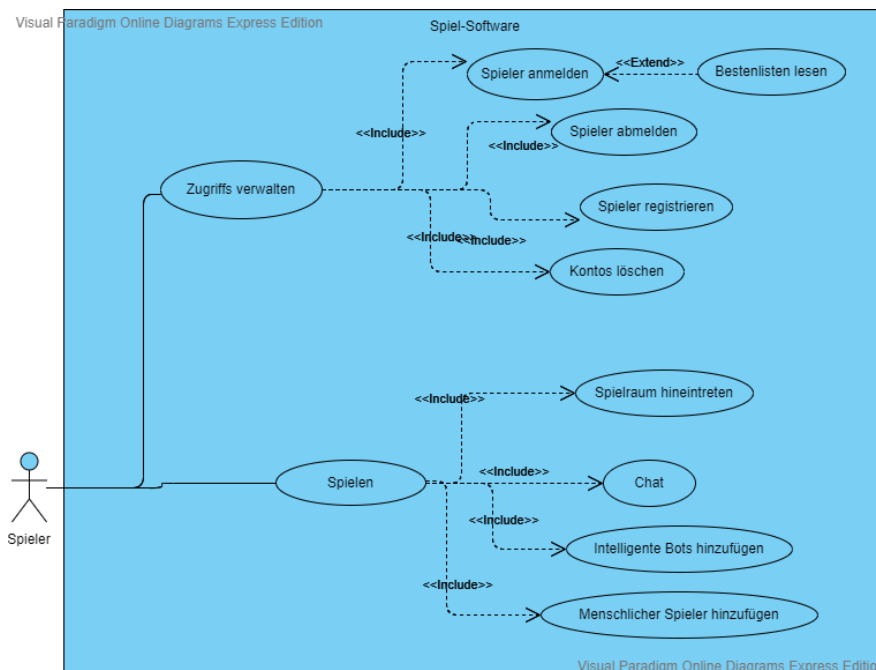


Abbildung 3.1: Use Case Diagramm

3.2 Systemgrenze (Use Case Diagramm)

Die Systemgrenze wird in der Abbildung 3.1 dargestellt.

3.3 Beschreibungen der Anwendungsfälle

/UC10/ **Name:** Zugriff verwalten.

Ziel: Zugriffsverwaltung.

Akteure: Das System.

Vorbedingungen: Falls vorhanden, Benutzername und Passwort.
Falls nicht vorhanden, keine.

Eingabedaten: Falls vorhanden, Benutzername /LD10/ und Passwort /LD20/. Falls nicht vorhanden, keine.

Beschreibung: Falls schon registriert, das Spieler gibt der Benutzername und das Passwort. Falls nicht registriert, das Spieler gibt Informationen zu registrieren. Richtigkeit vom Passwort.

Ausnahmen: Falls der Benutzername und dass Passwort passen nicht zusammen, das System gibt dem Spieler Hinweise.

Ergebnisse und Outputdaten: Das Spieler hat sich erfolgreich registriert oder angemeldet.

Systemfunktionen /LF30/.

/UC20/ **Name:** Spieler registieren.

Ziel: Spieler registiert sich.

Akteure: Spieler.

Vorbedingungen Keine.

Eingabedaten: Name/LD10/, Passwort/LD20/, Email??.

Beschreibung:

1. Spieler sendet das Formular ab.
2. Das System prüft ob der Benutzername schon vorhanden ist
3. Spieler erfolgreich registiert.

Ausnahmen:

Benutzername schon vorhanden: Das System zeigt eine Fehlermeldung an und Spieler sendet das Formular mit anderem Benutzername ab.

Ergebnisse und Outputdaten: Spieler erfolgreich registiert.

Systemfunktionen: /LF30/.

/UC30/ **Name:** Spieler anmelden.

Ziel: Spieler meldet sich im System an.

Akteure: Spieler.

Vorbedingungen Spieler ist erfolgreich angemeldet.

Eingabedaten: Zugriffsdaten /LD10/ /LD20/.

Beschreibung:

1. Spieler sendet das Formular ab.
2. Das System prüft die Gültigkeit von Zugangsdaten.

Ausnahmen:

Passwort oder Benutzername ist falsch: Das System zeigt eine Fehlermeldung an.

Ergebnisse und Outputdaten: Spieler erfolgreich angemeldet.

Systemfunktionen: /LF30/.

/UC40/ **Name:** Benutzername ändern.

Ziel: Spieler ändert seinen Benutzername.

Akteure: Spieler.

Vorbedingungen Spieler ist erfolgreich angemeldet.

Eingabedaten: Zugriffsdaten [/LD10/](#) [/LD20/](#).

Beschreibung:

1. Spieler sendet das Formular ab.
2. Das System prüft die Gültigkeit von Zugangsdaten.

Ausnahmen:

Password ist falsch: Das System zeigt eine Fehlermeldung an.

Ergebnisse und Outputdaten: Benutzername erfolgreich geändert.

Systemfunktionen: ??.

/UC50/ Name: Passwort ändern.

Ziel: Spieler ändert sein Passwort.

Akteure: Spieler.

Vorbedingungen Spieler ist erfolgreich angemeldet.

Eingabedaten: Passwort [/LD20/](#).

Beschreibung:

1. Spieler sendet das Formular ab.
2. Das System prüft die Gültigkeit des Passworts.

Ausnahmen:

Password ist falsch: Das System zeigt eine Fehlermeldung an.

Ergebnisse und Outputdaten: Passwort erfolgreich geändert.

Systemfunktionen: ??.

/UC60/ Name: Spieler löschen.

Ziel: Spieler entfernt seine Daten aus dem System.

Akteure: Spieler.

Vorbedingungen Spieler erfolgreich angemeldet.

Eingabedaten: Passwort [/LD20/](#).

Beschreibung:

1. Spieler sendet das Formular ab.
2. Das System prüft die Richtigkeit des Passworts und fragt Spieler noch ein mal, ob er sich wirklich aus dem System entfernen möchte.

3. Spieler bestätigt seine Intention.
4. Das System entfernt alle Daten des Spielers aus der Datenbank und bewegt Spieler in den Vorraum.

Ausnahmen:

Keine Löschung erwünscht: Schritt 4 wird nicht durchgeführt.

Ergebnisse und Outputdaten: Spielerkonto wurde gelöscht.

Systemfunktionen: /LF30/.

/UC70/ **Name:** Bestenliste lesen.

Ziel: Spieler liest die Bestenliste.

Akteure: Spieler.

Vorbedingungen Spieler erfolgreich angemeldet.

Eingabedaten: keine.

Beschreibung: Spieler liest die Bestenliste ??.

Ausnahmen: keine

Ergebnisse und Outputdaten: Spielerkonto wurde gelöscht.

Systemfunktionen: /LF50/.

/UC80/ **Name:** Spielen.

Ziel: Spieler spielt.

Akteure: Spieler.

Vorbedingungen Spieler erfolgreich angemeldet.

Eingabedaten: keine.

Beschreibung: Spieler kann den Spielraum hineintreten, mit anderen Benutzern chatten und spielen

Ausnahmen: keine.

Ergebnisse und Outputdaten: Spieler in einen Spielraum.

Systemfunktionen: /LF40/.

/UC90/ **Name:** Chatten.

Ziel: Spieler chat mit einander.

Akteure: Spieler.

Vorbedingungen Spieler erfolgreich angemeldet und steht in einem Spielraum.

Eingabedaten: keine.

Beschreibung: Spielern chatten mit einander.

Ausnahmen: keine.

Ergebnisse und Outputdaten: keine

Systemfunktionen: /LF70/.

/UC100/ **Name:** Menschlicher Spieler hinzufügen.

Ziel: Spieler fügt einen menschlichen Spieler zum Spielraum.

Akteure: Spieler.

Vorbedingungen Spieler steht in einem Spielraum.

Eingabedaten: keine.

Beschreibung: Spielern fügt einen menschlichen Spieler zum Spiel

Ausnahmen:

Keiner aktiver Spieler gefunden: warten für zwei Sekunden und versucht nochmal.

Ergebnisse und Outputdaten: Spieler erfolgreich gefügt.

Systemfunktionen: /LF70/.

/UC110/ **Name:** Intelligente Bots hinzufügen.

Ziel: Spieler fügt ein Bot Spieler zum Spielraum.

Akteure: Spieler.

Vorbedingungen Spieler steht in einem Spielraum.

Eingabedaten: keine.

Beschreibung: Spielern fügt ein Bot Spieler zum Spiel

Ausnahmen: keine

Ergebnisse und Outputdaten: Spieler erfolgreich gefügt.

Systemfunktionen: /LF60/.

3.4 Produktdaten

Hier sollen die Daten genannt werden, die im System verwendet werden.

/LD10/ Name: Benutzername*¹

Fachliche Beschreibung: Benutzername des Spielers

Relevante Systemfunktionen: /LF10/, /LF30/

/LD20/ Name: Passwort*

Fachliche Beschreibung: Passwort des Spielers

Relevante Systemfunktionen: /LF30/

¹ “*” bedeutet hier, dass die Daten in der Datenbank zu speichern sind

Kapitel 4

Nicht-funktionale Anforderungen

4.1 Softwarearchitektur

/NF10/ **Name:** Client-Server Anwendung

Beschreibung: Das verteilte Spiele-System ermöglicht das gemeinsame Spielen von verschiedenen Rechnern aus.

Motivation: Aufgabestellung v. SEP/MP.

Erfüllungskriterium: Das fertige System besteht aus Client- und Server-Teilen.

/NF20/ **Name:** Plattformunabhängigkeit

Beschreibung: Es soll sich um eine plattformunabhängige Anwendung handeln. Zumindest Windows- und Linuxsysteme sind zu unterstützen.

Motivation: Aufgabenstellung v. SEP/MP.

Erfüllungskriterium: Das Spiel funktioniert auf Linux- und Windows-Systemen

4.2 Benutzerfreundlichkeit

/NF30/ **Name:** Benutzeralter

Beschreibung: Das System ist für Benutzer geeignet, die älter als 5 Jahre sind.

Motivation: Jüngere Benutzer sind noch unfähig das Spiel zu spielen.

Erfüllungskriterium: In den AGBs steht ein entsprechender Hinweis.

/NF40/ **Name:** Technische Fähigkeiten

Beschreibung: Besondere technische Fähigkeiten sind von den Benutzern nicht zu erwarten.

Motivation: Auch die Menschen, die kaum etwas von Bedienung bzw. Programmierung von Rechnern verstehen, sollen fähig sein, das System zu verwenden.

Erfüllungskriterium: Intuitive GUI

4.3 Leistungsanforderungen

/NF50/ **Name:** Antwortzeit

Beschreibung: Maximale Antwortzeit für alle Systemprozesse.

Motivation: Das System muss immer brauchbar sein.

Erfüllungskriterium: Das System antwortet auf Benutzerhandlungen nie später als in 10 Sekunden.

4.4 Anforderungen an Einsatzkontext

Anforderungen an physische Umgebung

/NF60/ **Name:** Lauffähigkeit an SCI-Rechnern

Beschreibung: Das Produkt muss auf einem eigenem Gerät lauffähig sein, welches zur Präsentation am Ende des SEPs genutzt werden muss. Falls keine eigenen Rechner vorhanden sind, stehen auch die SCI-Terminals zur Verfügung.

Motivation: Optimierung von Betreuung und Abnahme des SEP/MP

Erfüllungskriterium: Spiel ist lauffähig auf SCI-Rechner

Absatz- sowie Installationsbezogene Anforderungen

/NF70/ **Name:** Installationsanleitung

Beschreibung: Falls die Installation nicht lediglich das Öffnen einer Datei voraussetzt, muss der genaue Installations- und Startvorgang schriftlich für Benutzer zur Verfügung gestellt werden.

Motivation: Spezifikation

Erfüllungskriterium:

Anforderungen an Versionierung

/NF80/ **Name:** Keine weitere Versionen

Beschreibung: Nach Version 1.0 ist keine weitere Entwicklung vorgesehen.

Motivation: Das ist nur das SEP/MP, kein Geschäftsprojekt, siehe [/FA10/](#)

Erfüllungskriterium:

4.5 Anforderungen an Wartung und Unterstützung

Wartungsanforderungen

/NF90/ **Name:** Dokumentation

Beschreibung: Der Quellcode muss ausführlich dokumentiert werden.

Motivation: Quellcode soll von jedem verstanden werden, nicht nur vom Ersteller.

Erfüllungskriterium: JavaDoc

/NF100/ **Name:** Testen

Beschreibung: Der Quellcode außer GUI muss gut getestet werden.

Motivation: Das Spiel sollte möglichst keine Fehler enthalten

Erfüllungskriterium: Von Unit-Tests muss mindestens 70% des Quellcodes bedeckt werden. GUI-Klassen sind aus der Anforderung ausgenommen.

Anforderungen an technische und fachliche Unterstützung

/NF110/ **Name:** Unterstützung

Beschreibung: Es ist keine technische und fachliche Unterstützung des Systems geplant.

Motivation: Siehe [/FA10/](#).

Erfüllungskriterium: Nicht anwendbar.

Anforderungen an technische Kompatibilität

/NF120/ **Name:** Kompatibilität

Beschreibung: Siehe [/NF20/](#)

Motivation:

Erfüllungskriterium:

4.6 Sicherheitsanforderungen

Zugang

/NF130/ **Name:** Spielerzugriff

Beschreibung: Spieler haben nur Zugriff auf das Konto auf dem sie angemeldet sind und können nur an diesem Änderungen vornehmen.

Motivation: Schutz vor Diebstahl von Konten.

Erfüllungskriterium:

Integrität

/NF140/ **Name:** Regelwerk

Beschreibung: Die Spielregeln dürfen von keinem Spieler umgangen werden.

Motivation: Verhindern von unfairen Vorteilen.

Erfüllungskriterium:

Datenschutz/Privatsphäre

/NF150/ **Name:** Passwörter

Beschreibung: Nur der Server und kein Spieler hat Zugriff auf Passwörter.

Motivation: Schutz vor Diebstahl von Daten.

Erfüllungskriterium:

Virenschutz

/NF160/ **Name:** Beispiel

Beschreibung:

Motivation:

Erfüllungskriterium:

4.7 Prüfungsbezogene Anforderungen

Anforderungen, die sich auf die Prüfung/Audit vom System von SEP/MP-Tutoren oder von weiteren Instanzen beziehen.

/NF170/ **Name:** Formate der Systemdokumentation

Beschreibung: Systemdokumentation muss in 2 Formen geführt werden (wenn anwendbar): Die Ausgangsdateien (\LaTeX , Dateien der Diagrammerstellungssoftware, Dateien der Grafiksoftware usw.) und PDFs.

Motivation: Optimierung der SEP/MP-Betreuung.

Erfüllungskriterium: Siehe Beschreibung.

4.8 Kulturelle und politische Anforderungen

/NF180/ **Name:** Systemsprache

Beschreibung: Die Interfacesprache ist Deutsch.

Motivation: Synchronisation des Verständnisses von Teammitgliedern mit unterschiedlichen kulturellen Hintergründen.

Erfüllungskriterium:

4.9 Rechtliche und standardsbezogene Anforderungen

/NF190/ **Name:** Nicht rechtliche Anforderungen

Beschreibung: Keine relevanten rechtlichen Anforderungen bekannt.

Motivation: Siehe [/FA10/](#).

Erfüllungskriterium: Nicht anwendbar.

Kapitel 5

Systemtestfälle

Hier sollen verschiedene Szenarien beschrieben werden, mithilfe deren Sie später Systemtests ausführen und die erwarteten Ergebnisse darstellen.

/TF10/ Name: Spieler anmelden.

Motivation: Testet, ob die Anmeldung im System korrekt funktioniert.

Szenarien:

1. *Zugriffsdaten sind vorhanden und richtig*
⇒ Spieler wird in die Lobby bewegt.
2. *Benutzername ist registriert, Passwort ist falsch*
⇒ Fehlermeldung wird angezeigt.
3. *Benutzername ist nicht registriert*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: [/LF30/](#)

Relevante Use Cases: [/UC30/](#)

/TF20/ Name: Spieler registrieren.

Motivation: Testet, ob die Registrierung in dem System korrekt funktioniert.

Szenarien:

1. *Benutzername sind vorhanden und richtig*
⇒ Ein Account mit den eingetragenen Daten wird erzeugt und der Spieler wird in die Lobby bewegt.
2. *Benutzername ist schon genutzt*
⇒ Fehlermeldung wird angezeigt.
3. *Email ist schon genutzt*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: [/LF30/](#)

Relevante Use Cases: /UC30/

/TF30/ **Name:** Account bearbeiten.

Motivation: Testet, ob das Bearbeiten des Accounts in dem System korrekt funktioniert.

Scenarien:

1. *Neuer Benutzername nicht vorhanden*
⇒ Benutzername wird erfolgreich geändert.
2. *Neuer Benutzername bereits vorhanden*
⇒ Fehlermeldung wird angezeigt.
3. *Neues Passwort bearbeiten, altes richtig*
⇒ Passwort wird erfolgreich geändert.
4. *Neues Passwort bearbeiten, altes falsch*
⇒ Fehlermeldung wird angezeigt.
5. *Profil Bild bearbeiten*
⇒ Profil Bild wird erfolgreich geändert.
6. *Profil Bild bearbeiten funktioniert nicht*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF40/ **Name:** Account löschen.

Motivation: Testet, ob das Löschen des Accounts in dem System korrekt funktioniert.

Scenarien:

1. *Passwort richtig*
⇒ Spieler wird in den Anmeldebildschirm bewegt, Account wird gelöscht.
2. *Passwort falsch*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF50/ **Name:** Spielraum erstellen, bearbeiten oder löschen.

Motivation: Testet, ob das Erstellen, das Bearbeiten oder das Löschen eines Spielraums im System korrekt funktioniert.

Scenarien:

1. *Spielraum erstellt*
⇒ Spieler wird im erstellte Spielraum bewegt.

2. *Spielraum erstellt funktioniert nicht*
⇒ Fehlermeldung wird angezeigt und kein Spielraum wird erstellt.
3. *Spielraum bearbeiten*
⇒ Spielraum Eigenschaften werden geändert.
4. *Spielraum bearbeiten funktioniert nicht*
⇒ Fehlermeldung wird angezeigt.
5. *Spielraum löschen*
⇒ Spieler soll das Löschen bestätigen.
⇒ Spieler wird im Spiel Menü bewegt.
⇒ Spielraum wird gelöscht.
6. *Spielraum löschen funktioniert nicht*
⇒ Fehlermeldung wird angezeigt.
⇒ Spielraum wird nicht gelöst.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF60/ **Name:** Bots.

Motivation: Testet, ob das Hinzufügen bzw. das Entfernen der Bots korrekt funktioniert.

Scenarien:

1. *Bots hinzufügen, Platz vorhanden*
⇒ Bots werden im Spielraum bewegt.
2. *Bots hinzufügen, kein Platz vorhanden*
⇒ Fehlermeldung wird angezeigt.
3. *Bots entfernen, Bot im Raum*
⇒ Bots werden entfernt.
4. *Bots entfernen, kein Bot im Raum*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF70/ **Name:** Spiel.

Motivation: Testet, ob das Starten bzw. das Ende des Spiels korrekt funktioniert.

Scenarien:

1. *Spiel starten, genug Spieler*
⇒ Spiel starten.
⇒ Spieler werden im Spielraum bewegt.

2. *Spiel starten, nicht genug Spieler*
⇒ Fehlermeldung wird angezeigt.
3. *Spiel beenden funktioniert*
⇒ Spiel wird beendet.
⇒ Spieler werden in Lobby bewegt.
4. *Spiel beenden funktioniert nicht*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF80/ **Name:** Bestenliste.

Motivation: Testet, ob das Anzeigen der Bestenliste korrekt funktioniert.

Scenarien:

1. *Bestenliste anzeigen, mindestens ein Spiel abgeschlossen*
⇒ Bestenliste anzeigen.
2. *Bestenliste anzeigen, bisher niemand gespielt*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF90/ **Name:** Chatroom.

Motivation: Testet, ob das Empfangen bzw. das Senden von Nachrichten korrekt funktioniert.

Scenarien:

1. *Nachricht schicken, mit Inhalt*
⇒ Nachricht wird verschickt und erscheint im Chatfenster aller Spieler.
2. *Nachricht schicken, kein Inhalt*
⇒ Fehlermeldung wird angezeigt.
⇒ Keine Nachricht wird verschickt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF100/ **Name:** :Spieler einladen.

Motivation: Testet, ob man Spieler einladen kann.

Scenarien:

1. *Spieler wird zum Spiel eingeladen, Spieler existiert*
⇒ Spiel starten mit Freunden.

2. *Spieler wird zum Spiel eingeladen, Spieler existiert nicht*
⇒ Fehlermeldung wird angezeigt.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

/TF110/ **Name:** Spieler Punkte.

Motivation: Testet, ob die Spielers Punkte übertragen werden.

Scenarien:

1. *Das Spiel ist fertig. Kein Spielabbruch.*
⇒ Verdiente Punkte werden zu den vorherigen Punkte addiert.
2. *Das Spiel ist fertig. Spielabbruch.*
⇒ Keine Punkte werden zu den vorherigen Punkte addiert.

Relevante Systemfunktionen: /LF30/

Relevante Use Cases: /UC30/

Kapitel 6

Warteraum

Hier werden Anforderungen spezifiziert die den sogenannten “Warteraum” darstellen. Hier gehören alle Anforderungen, die “Wünschkriterien” sind, das heißt, sie sind zwar erwünscht, aber werden nur dann in aktuelle Anforderungen übernommen, wenn dafür genügendes Zeitbudget vorhanden ist und werden am wahrscheinlichsten in der Zukunft (und nicht jetzt) implementiert (oder in den kommenden Sprints beim SCRUM-Prozessmodell).

/WR10/ Name: Hintergrundmusik

Beschreibung: Für die Spieler soll eine Auswahl zur Verfügung stehen, mit der die Hintergrundmusik beim Spielen ausgewählt werden kann.

Motivation: Höhere Zufriedenheit der Benutzer.

Erfüllungskriterium: Spieler können zu jedem Zeitpunkt (außer im Vorraum) die Musik ausschalten oder ein anderes Lied auswählen.