



- Team 16 -

Cornelius Huber

Enrique Rempp

Philipp Letschka

René Müller

Pflichtenheft v21.4.28

Inhaltsverzeichnis:

Pflichtenheft v21.4.28	1
Inhaltsverzeichnis:	2
Meilenstein 1	4
Kontextanalyse	4
Fachwissen	5
Domänenmodell	12
Anforderungsdefinitionen	13
Akteure:	13
Funktionale Anforderungen:	14
Nichtfunktionale Anforderungen:	32
Meilenstein 2	36
Anwendungsfälle:	36
Main Menu:	36
Spiel beitreten:	38
Spielablauf (Serverside):	39
Spieler am Zug:	42
Editor:	43
Abläufe im System:	44
Zustandsdiagramme:	44
Player am Zug:	44
Hauptmenü:	46
Sequenzdiagramme:	47
Meilenstein 3	51
Dialogstruktur	51
Graphische Gestaltung und Nutzungskonzept	53
Hauptmenü:	53
Join Game:	55
Create Game:	55
Game:	57
Editor:	58
Character Editor:	58
Map Editor:	59
Szenario Editor:	60
Dialoge:	61
Alert Dialog:	61
Info Dialog:	62
Rückmeldung:	63
Server und KI Client:	64
Meilenstein 5:	65
Komponentendiagramm:	65
Artefakte:	68

Kontextanalyse

In diesem Softwaregrundprojekt geht es um die Realisierung eines rundenbasiertes Taktik-Spiels welches sich an dem Marvel Universum orientiert. Das Spiel trägt den Namen "Marvelous Mashup" und lässt den Spieler in eine Rolle eines Marvel Helden steigen und gegen andere Spieler oder eine Künstliche Intelligenz spielen, die von Zuschauern im Nachhinein verfolgt werden kann.

Die Versionierung der Projektdateien wird durch Git gestützt, die Architektur basiert auf der Client-Server-Architektur, das bedeutet, dass die Spieler Clients nicht untereinander kommunizieren, sondern lediglich über den Server.

Es werden von verschiedenen Parteien die essentiellen Module der Spiele Software entwickelt (KI-Client, Server, Benutzer-Client, Editor) die nach Fertigstellung zusammengesetzt werden können. Durch diese Modularität können verschiedene Modulare Realisierungen ausgetauscht werden.

Das fertige System umfasst eine spielbare Version von Marvelous Mashup, bei dem der Spieler das Ziel hat, die sechs Infinity Stones anzueignen und das Universum zu retten - und das bevor der Gegner dies getan hat.

Nach der Umsetzung des hiesigen Projekts lassen sich zahlreiche weitere Zusatzfunktionen realisieren, wie das erweitern der Heldenauswahl, der Ausreifung der KI durch weitere Lernalgorithmen oder dem Hinzufügen weiterer Modi wie beispielsweise ein Kompetitives Ranking-System, dass derzeitigen Esport-Standards gerecht wird.

Durch die Freiheit eines modularen System lassen sich dementsprechend auch Lastenheft Fremde Erweiterungen einfach implementieren, wie beispielsweise ein Chat oder Ähnliches, die das System ausschmücken.

Fachwissen

Begriff	Charakter
Beschreibung	Eine Figur aus allseits beliebten Comic-Heften, die durch eine Einheit auf dem Raster der Spielfelder repräsentiert wird.
Ist-ein	Bewegliche Einheit
Kann-sein	Player Character (PC), Non-Player Character (NPC)
Aspekt	Name, zugehöriger Spieler oder NPC, Eigenschaften, Health Points, Inventar, Position auf dem Spielfeld
Bemerkung	Der Begriff "Charakter" bezeichnet einen NPC der auf dem Spielfeld herumläuft
Beispiele	Wade Winston Wilson aka. Wade "Wichslurch" Wilson aka. Deadpool Gehört zur Heldengruppe von Benutzer Billy Healthpoints: 80 Movement Points: 2 Action Points: 2 Inventar: empty Position {x: 3, y: 4} etc.

Begriff	Player Character
Beschreibung	Ein Charakter der von einem Menschen gesteuert wird
Ist-ein	Charakter
Kann-sein	
Aspekt	Der Player Charakter bringt das Spielgeschehen durch Aktionen voran
Bemerkung	
Beispiele	Deadpool, Thanos

Begriff	Non Player Character
Beschreibung	Ein Programmgesteuerter Charakter. Er wird nicht von einem Spieler gesteuert, daher kann sein handeln nicht gesteuert werden.
Ist-ein	Programmgesteuerter Verhaltens Algorithmus
Kann-sein	
Aspekt	Non Player Characters (NPCs) beeinflussen das Spielgeschehen auf Basis eines Algorithmus
Bemerkung	Diese Charaktere können nicht manuell vom Spieler gesteuert werden
Beispiele	Stan Lee, Goose

Begriff	Spieler
Beschreibung	Eine Person oder künstliche Intelligenz der oder die Aktionen von den Charakteren (wie Spielzüge) ausführt.
Ist-ein	Aktiver Teilnehmer des Spiels
Kann-sein	Mensch, künstliche Intelligenz
Aspekt	Ein Spieler besitzt 6 Charaktere, welche in der Spielvorbereitung gewählt werden.
Bemerkung	Es sind immer 2 Spieler pro Spiel notwendig
Beispiele	

Begriff	Künstliche Intelligenz / KI
Beschreibung	Ersetzt einen oder mehrere Spieler, um das Spiel auch alleine spielen zu können oder falls Soll an Haben der Mitspieler nicht erfüllt ist
Ist-ein	Ersatz für einen Spieler
Kann-sein	Unterschiedlicher Schweregrad
Aspekt	Schwierigkeit, IP und Port des Servers
Bemerkung	Kann aktiviert werden falls es nicht genug Spieler gibt.
Beispiele	

Begriff	Wahlphase
Beschreibung	Zu Beginn muss der Spieler eine Heldengruppe aus 6 Charakteren zusammenstellen. Die Herausforderung hier ist, die Charaktere aus dem Vorrat der in der Charakter-Konfiguration definierten Figuren so zu wählen, dass eine möglichst gute Kombination entsteht.
Ist-ein	Spielvorbereitung
Kann-sein	
Aspekt	
Bemerkung	Der Ablauf der Wahlphase aus Sicht eines Spielers ist so: 1. Der Server nimmt aus der Liste aller Charaktere der geladenen Charakter-Konfiguration 12 zufällige Charaktere, die dem Spieler angezeigt werden. 2. Der Spieler wählt 6 davon aus.
Beispiele	Spieler A wählt Thor, Hulk, Mantis, Rocket, Spider Man und Captain Marvel

Begriff	Highscore
Beschreibung	Zeigt den höchsten erreichten Punktestand an
Ist-ein	Vergleichswert den man übertreffen möchte
Kann-sein	Orientierung eines Höchstwertes, den es zu knacken gilt (Motivation)
Aspekt	Punktzahl
Bemerkung	
Beispiele	Spieler A hat einen Punktestand von 420.000 und damit einen Highscore, da es der Höchstwert ist

Begriff	Marvel
Beschreibung	Marvel ist eine US-Amerikanische Comic und Film Produktionsfirma. Sie hat u.a. Das marvel Cinematic Universe erschaffen, zusammen mit einer großen Anzahl an Superhelden.
Ist-ein	US-amerikanische Produktionsfirma für Film- und Fernsehen
Kann-sein	
Aspekt	Hat die Charaktere erfunden welche in unserem Spiel spielbar sein werden.
Bemerkung	
Beispiele	

Begriff	Infinity Stones
Beschreibung	Steine verschiedener Farbe mit unterschiedlichen Effekten, 6 Stück
Ist-ein	Item auf dem Spielfeld oder getragen von einem Charakter
Kann-sein	Verstärker eines Charakters wenn dieser ihn trägt indem es ihm einen zusätzlichen Effekt oder eine Fähigkeit gibt
Aspekt	Effekt (individuell)
Bemerkung	Jede Farbe hat einen bestimmten Effekt, ein Charakter kann mehrere Steine besitzen und so mehrere Effekte nutzen, Ziel des Spiel ist es alle 6 Steine auf einem Charakter zu haben.
Beispiele	Space Stone (blau): Der Träger kann sich mit einer Aktion auf ein beliebiges freies feld teleportieren.

Begriff	Effekte
Beschreibung	Jeder Infinity Stone hat einen eigenständigen
Ist-ein	Ein Effekt ist eine weitere Aktion die vom Spieler ausgeführt werden kann
Kann-sein	Eine Verstärkung für den Charakter
Aspekt	Cooldown Zeit (individuell)
Bemerkung	Jeder Effekt hat eine eigene Farbe
Beispiele	Space Stone (blau): Der Träger kann sich mit einer Aktion auf ein beliebiges freies feld teleportieren.

Begriff	Spielzug
Beschreibung	Die Charaktere kommen in einer zufälligen Reihenfolge hintereinander dran. Wenn ein Charakter an der Reihe ist kann dieser seine Movement Points und Action Points nutzen.
Ist-ein	Moment in dem ein Charakter eine oder mehrere Handlungen ausführen kann.
Kann-sein	
Aspekt	
Bemerkung	Alle Action und Movement Points werden zu Beginn des Spielzugs von einem Charakter zurückgesetzt.
Beispiele	Rocket Raccoon ist am Zug. Er hat z.B. laut der Charakter-Konfiguration 2 MP und 2 AP. Als erstes benutzt der Spieler einen AP um Rocket einen Fernkampf-Angriff auf Nebula machen zu lassen, die in Reichweite und Sichtlinie steht. Dann benutzt er einen MP, um den Space Stone auf einem Nachbarfeld aufzusammeln. Mit dem zweiten AP benutzt Rocket den Space Stone um sich neben den ausgeknockten Starlord zu teleportieren, damit er ihn im nächsten Zug (wenn er wieder APs hat) mit dem Soul Stone in seinem Inventar wiederbeleben kann.

Begriff	Movement-Points (MP)
Beschreibung	Diese Punkte geben die Anzahl an möglichen Bewegungen an, die ein Spieler oder die KI während ihrer Runde tätigen können
Ist-ein	Indikator der die Anzahl an Bewegungen einschränkt
Kann-sein	Eine Meta die Taktik implementiert und zum Überlegen anreizt, da die Bewegungen limitiert werden
Aspekt	Anzahl der freien Bewegungen pro Runde
Bemerkung	Jeder Spieler / Jede KI hat zu ihrer Rundenzeit eine verfügbare MP pro Charakter
Beispiele	Thanos hat 5 MP in der Runde, somit kann er 5 Bewegungen durchführen und sich auf dem Spielfeld bewegen

Begriff	Health-Points (HP)
Beschreibung	Diese Punkte indizieren die Lebenspunkte die ein Charakter zur Verfügung hat. Fällt dieser Wert auf 0, ist der Charakter außer Gefecht gesetzt und steht dem Spieler / der KI nicht mehr zur Verfügung (außer ein bestimmter Effekt wird im weiter Spielverlauf aktiviert).
Ist-ein	Indikator für die Lebenspunkte eines Charakters
Kann-sein	Motivation, den Charakter taktisch klug einzusetzen, damit er zum Sieg verhilft
Aspekt	Anzahl der Lebenspunkte
Bemerkung	Jeder Charakter besitzt Lebenspunkte, die im Verlauf des Spiels zu oder abnehmen
Beispiele	

Begriff	Meta
Beschreibung	Die Meta ist im Esports die "Most effective strategy available". Durch sorgfältiges balancieren von Werten und Fähigkeiten kann diese angepasst werden.
Ist-ein	Spielregeln, die das Spiel interessanter / kompetitiver gestalten
Kann-sein	Ansporn nachzudenken, da diese Spielregeln das Spiel meist taktisch schwieriger gestalten
Aspekt	
Bemerkung	Die Meta ist nicht essentiell für ein Spiel notwendig, ein Spiel kann theoretisch auch ohne Meta Elemente gespielt werden, allerdings gibt die Meta dem Spiel meist erst einen Spielwert, da sie das Projekt interessanter gestaltet für den Spieler, da dieser sich mit dieser vertraut machen muss und somit durch überlegen das Spiel gewinnen kann. Hiermit wird also auch die geistige Fähigkeit der Spieler unter Beweis gestellt.
Beispiele	Movement-Points, Health-Points

Begriff	Aktion
Beschreibung	Aktionen werden von Charakteren ausgeführt und bewirken verschiedene Effekte und Zustände, sowie Inkrementierung oder Dekrementierung von HP sowie MP und sind ein essentieller Bestandteil von <i>Marvelous Mashup</i>
Ist-ein	Essentielles Spielelement, dass es den Spielern / der KI erlaubt den Charakter bestimmte Dinge ausführen zu lassen
Kann-sein	
Aspekt	Art der Aktion, Aktion und Reaktion auf vorhergehende Einflüsse
Bemerkung	Durch Aktionen wird der Spielverlauf beeinflusst
Beispiele	Ein Charakter möchte sich bewegen- Dazu muss er die Bewegungsaktion anwenden um sich über das Spielfeld zu bewegen; Dies kostet in MP, also wird hierbei pro Bewegungsaktion ein MP dekrementiert

Begriff	Savanne von Wakanda
Beschreibung	Die Savanne von Wakanda ist ein fiktiver Ort auf dem Kontinent Afrika südlich der Sahara, welcher von Marvel erschaffen wurde.
Ist-ein	Sie dient als Ort des Spiels und wird damit auch zum Ort des Spielfelds.
Kann-sein	
Aspekt	Beinhaltet das Spielfeld
Bemerkung	Der Hintergrund im Spiel sollte an die Savanne von Wakanda erinnern.
Beispiele	

Begriff	Spielfeld
Beschreibung	Ein x auf y großes rechteckiges Raster aus Feldern
Ist-ein	
Kann-sein	Benutzerdefiniert angepasst
Aspekt	Beinhaltet alle Felder
Bemerkung	
Beispiele	Ein 10 * 10 großes Feld mit einer Mischung aus verschiedenen Feldern

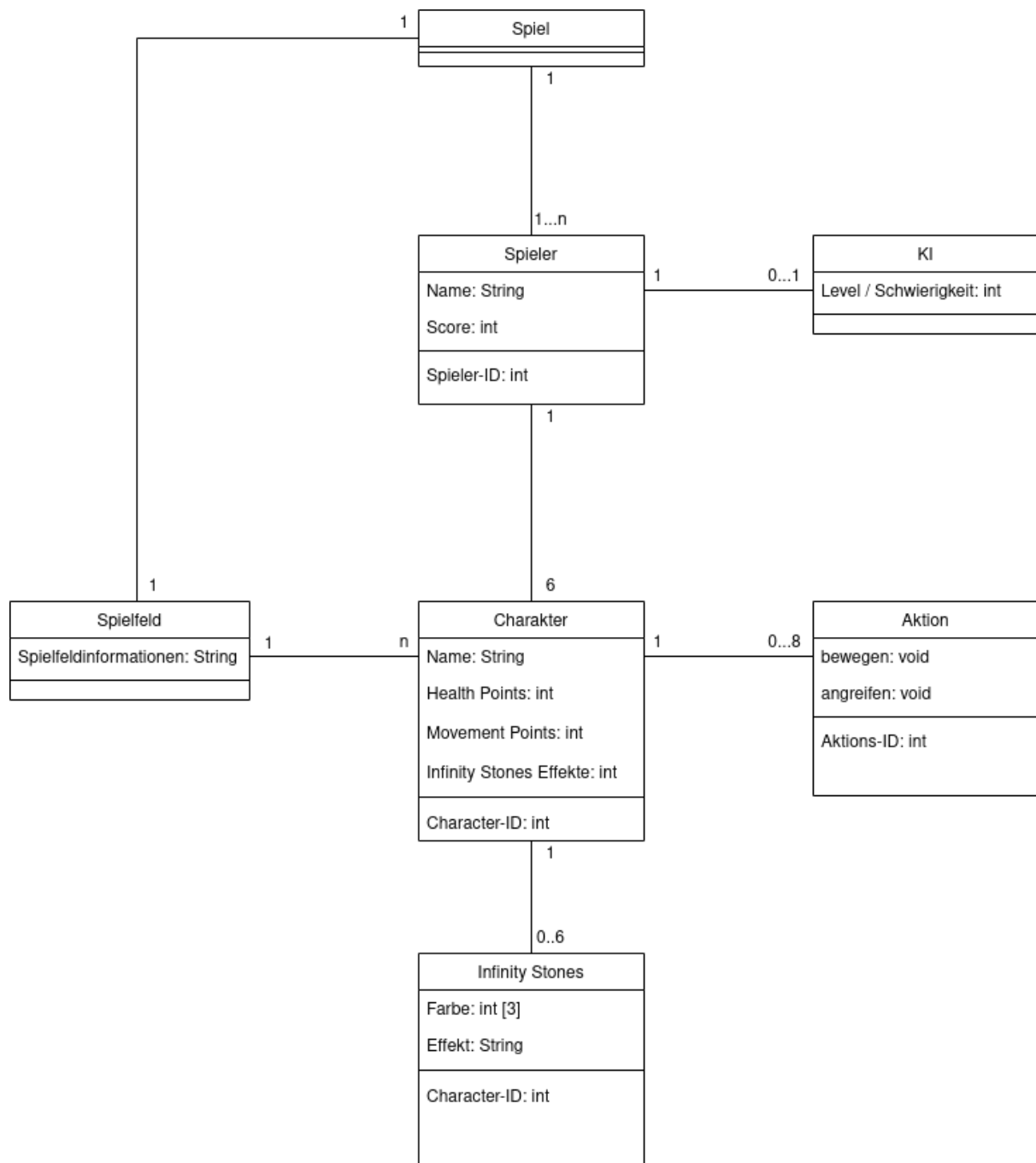
Begriff	Feld
Beschreibung	Felder des Spielbretts können freier Raum sein, oder durch Hindernisse oder Objekte besetzt sein. Sie können betretbar sein oder nicht, und die Sichtlinie durchlassen, oder blockieren.
Ist-ein	Ort auf dem ein Charakter stehen kann
Kann-sein	Gras, Gras mit Fels, Gras mit Charakter, Gras mit Infinity Stone.
Aspekt	Position auf dem Spielfeld
Bemerkung	
Beispiele	Das Feld an der rechten unteren Ecke ist ein Grasfeld ohne weitere besetzung.

Begriff	Goose
Beschreibung	Goose ist eine (besondere) Katze (Flerken) aus dem Marvel Cinematic Universe. Sie hat alle Infinity Stones verschluckt und verteilt nun mit der Zeit alle Steine auf dem Spielfeld.
Ist-ein	Flerken
Kann-sein	
Aspekt	
Bemerkung	Zu Beginn der Runden 1 bis 6 (also bevor die Charaktere ihre Züge machen) erscheint Goose auf einem zufälligen freien Feld, kotzt einen zufälligen der sich noch in ihrem Magen befindenden Infinity Stones auf diesem Feld aus, und verschwindet dann wieder.
Beispiele	Runde 1: Goose hat noch alle Steine, daher wird zufällig der Space Stone gewählt und auf Feld {3,4} platziert.

Begriff	Stan Lee
Beschreibung	Tritt vor Runde 7 auf und erscheint neben den schwächsten Charakter. Er heilt daraufhin alle die sich in seiner Sichtweite befinden.
Ist-ein	Erfinder der (meisten) Superhelden von Marvel und Comicautor der Marvel Comics. NPC
Kann-sein	
Aspekt	
Bemerkung	Stan ruft Excelsior um alle wiederzubeleben.
Beispiele	

Begriff	User-Client/Benutzer-Client
Beschreibung	Software, die im Gegensatz zum Server auf dem Rechner des Anwenders läuft.
Ist-ein	Die graphische Anwendung des Spiels.
Kann-sein	
Aspekt	
Bemerkung	
Beispiele	

Domänenmodell



Anforderungsdefinitionen

Akteure:

Entwickler:

Die Entwickler sind wir. Wir erstellen das Spiel komplett. Dabei können wir in unterschiedlichen Bereichen arbeiten.

Server:

Alle Aktionen die von dem Server ausgeführt werden.

Spieler:

Die Person, die das Spiel an der Client App spielt.

KI:

KI steht für künstliche Intelligenz. Dies ist Code, der einen Spieler im Spiel ersetzen soll, indem dieser sich wie ein Mensch im Spiel verhält und Entscheidungen trifft.

NPC:

NPC steht für Non-Player-Character. Ein NPC stellt einen Charakter im Spiel dar, der nicht von Menschen gespielt werden kann.

Thanos:

Thanos ist ein NPC. Dieser NPC ist der Antagonist im Spiel.

Stan Lee:

Stan Lee ist ein NPC. Er ist der Erfinder bzw. Autor von Marvel.

Goose:

Goose ist ein NPC. Goose erscheint vor jeder Runde auf einem zufälligen und freien Feld und legt einen Infinity Stone ab.

Funktionale Anforderungen:

ID:	FA1
Titel:	Architektur
Beschreibung:	Das zu entwickelnde Produkt ist eine verteilte Anwendung mit Client-Server-Architektur. Die Clients kommunizieren dabei nur mit dem Server, aber nicht untereinander.
Begründung:	Das Spiel ist ein Online Spiel das von mehreren Spielern gleichzeitig gespielt wird und von einem Server abhängig ist.
Abhängigkeiten:	FA2 - FA8, FA33, FA34
Akteure:	Entwickler

ID:	FA2
Titel:	Server
Beschreibung:	Der Server beinhaltet die Anwendungslogik zur Verwaltung von Clients und Partien sowie die Umsetzung der Spielregeln. Spiele können von beliebig vielen Zuschauern passiv mitverfolgt werden.
Begründung:	Es ist eine zentrale Software benötigt, die (möglichst nicht manipulierbar) das Spielgeschehen steuert.
Abhängigkeiten:	FA1, FA33, FA34
Akteure:	Server, Entwickler

ID:	FA3
Titel:	User-Client
Beschreibung:	Ein Benutzer-Client mit grafischer Oberfläche, der es einem einzelnen menschlichen Benutzer ermöglicht, entweder aktiv als Mitspieler an einer Partie teilnehmen, oder als passiver Zuschauer im Zuschauermodus eine Partie zwischen zwei anderen Clients zu verfolgen.
Begründung:	Das Spiel muss auch gespielt werden können
Abhängigkeiten:	FA2, FA8, FA33, FA34
Akteure:	Entwickler, Spieler

ID:	FA4
Titel:	KI-Client
Beschreibung:	KI-Clients werden autonom von einer Künstlichen Intelligenz gesteuert und können als Mitspieler an einer Partie teilnehmen.
Begründung:	Damit können auch Personen ohne Freunde gegen jemanden spielen.
Abhängigkeiten:	FA2, FA3, FA33, FA34
Akteure:	KI, Entwickler

ID:	FA5
Titel:	Editor
Beschreibung:	Ein Editor, mit dem Content und Konfigurationen für das Spiel erstellt werden können, wie Szenarios, Partie Konfigurationen und Charakter-Konfigurationen.
Begründung:	Somit können die Spiele nach eigenem Ermessen beliebig gestaltet werden.
Abhängigkeiten:	FA35
Akteure:	Entwickler

ID:	FA8
Titel:	Kommunikation möglich
Beschreibung:	Über die WebSocket-Verbindung werden Textstrings im UTF-8 Encoding ausgetauscht, die im Format JSON repräsentierte Nachrichten enthalten.
Begründung:	Damit ist die Nachricht für sowohl Client-Anwendungen oder Server lesbar und interpretierbar.
Abhängigkeiten:	FA2, FA3, FA4, FA5, FA7, FA33, FA34
Akteure:	Entwickler, Server

ID:	FA9
Titel:	Charakterinfo anzeigen
Beschreibung:	Der Spieler kann nach belieben einsehen welche Werte alle Charaktere haben.
Begründung:	Das Spiel ist ein Spiel mit vollständiger Information.
Abhängigkeiten:	FA3, FA11, FA21
Akteure:	Spieler

ID:	FA10
Titel:	Spielfeld anzeigen
Beschreibung:	Das Spielfeld ist ein rechteckiges kartesisches Raster aus $x * y$ Feldern.
Begründung:	Der Spieler muss den momentanen Zustand des Spiels sehen.
Abhängigkeiten:	FA3, FA17, FA20, FA21, FA22, FA23, FA24
Akteure:	Spieler

ID:	FA11
Titel:	Charakter Info verwalten
Beschreibung:	<p>Charaktere haben bestimmte Werte, die in der Charakter-Konfiguration definiert sind:</p> <ul style="list-style-type: none">• Name: Superhelden haben einen coolen Superhelden-Namen.• Movement Points (MP) und Action Points (AP): Jeder Charakter bekommt pro Zug eine bestimmte Anzahl an Punkten, die er jeweils für Bewegungsschritte und Aktionen nutzen kann.• Health Points (HP): Stellen den Gesundheitszustand des Charakters dar. Wenn die HPs auf 0 sinken, ist der Charakter ausgeknockt, und liegt einfach passiv am Boden. Er kann keine Züge mehr machen, bis er ggf. wiederbelebt wird.• Inventar: Ein Charakter kann Infinity Stones in seinem Inventar tragen.• Position: Die Position auf dem Spielfeld <p>Diese Informationen müssen richtig gespeichert, verwaltet, und übertragen werden.</p>
Begründung:	Damit die Charaktere korrekt gesteuert werden können müssen die Werte von ihnen richtig gespeichert und verwaltet werden.
Abhängigkeiten:	FA17
Akteure:	Spieler, Server

ID:	FA14
Titel:	Infinity Stones
Beschreibung:	<p>Es gibt sechs Infinity Stones. Jeder Stein hat eine Farbe und erlaubt dem Charakter, der ihn im Inventar hat, eine spezielle Aktion auszuführen. Nachdem diese Aktion ausgeführt wurde, kann der jeweilige Stein n Runden lang nicht mehr benutzt werden, um diese Aktion auszuführen. Auch wenn der Stein von jemand anderem übernommen wird, bleibt sie wirksam.</p> <p>Steine:</p> <ul style="list-style-type: none"> • Space Stone (blau): Der Träger kann sich mit einer Aktion auf ein beliebiges freies Feld teleportieren. • Mind Stone (gelb): Der Träger kann seine mentale Energie in einen Strahl fokussieren, und einen Fernkampf Angriff auf jedes Ziel in Sichtlinie ausführen. • Reality Stone (rot): Der Träger kann mit einer Aktion auf einem benachbarten Feld einen Felsen verschwinden lassen, oder auf einem benachbarten freien Feld einen Felsen erscheinen lassen. • Power Stone (violett): Der Träger kann mit einer Aktion einen Nahkampfangriff ausführen, der den doppelten Schaden seines normalen Nahkampf-Angriffs macht. Die HPs des Trägers bei der Aktion um 10% seiner maximalen HPs reduziert. • Time Stone (grün): Der Träger kann mit einer Aktion die "Zeit zurückspulen", und seine MPs und APs auf den Anfangswert zurücksetzen. • Soul Stone (orange): Der Träger kann mit einer Aktion einen ausgeknockten Charakter auf einem benachbarten Feld "wiederbeleben".
Begründung:	Zusätzliche Fähigkeiten und Gegenstand der Begierde.
Abhängigkeiten:	FA15
Akteure:	Goose, Spieler, KI

ID:	FA15
Titel:	Felder
Beschreibung:	Jedes einzelne Feld auf dem Spielbrett hat 3 Wahrheitswerte, ob es frei ist, ob es betretbar ist und ob es die Sicht blockiert, diese hängen von dem ab was sich auf dem Feld befindet
Begründung:	
Abhängigkeiten:	FA2, FA10, FA14
Akteure:	Spieler

ID:	FA16
Titel:	Spielstart
Beschreibung:	Vor der ersten Runde platziert der Server alle Charaktere auf zufälligen freien Felder auf dem Spielbrett.
Begründung:	Die Charaktere brauchen einen Ort an dem sie sich am Anfang befinden.
Abhängigkeiten:	FA2, FA15
Akteure:	Server, Spieler, KI

ID:	FA17
Titel:	Aktuellen Charakter anzeigen
Beschreibung:	Der Charakter, welcher gerade an der Reihe ist, wird gehighlightet.
Begründung:	Damit wird dem Spieler gezeigt, welcher Charakter am Zug ist.
Abhängigkeiten:	FA2, FA9, FA15
Akteure:	Server, Spieler

ID:	FA18
Titel:	Thanos
Beschreibung:	Nach einer zuvor definierten Anzahl an Runden erscheint Thanos auf einem zufälligem freiem Feld. Er bewegt sich auf das nächstgelegene Feld mit Infinity Stones zu und verwandelt auf dem Weg dazu alle Hindernisse zu Gras. Thanos erhält jede Runde einen zusätzlichen MP und ist immun gegen Angriffe.
Begründung:	Allzulange Partien sollen durch eine spieleigene Mechanik verhindert werden.
Abhängigkeiten:	FA21
Akteure:	Server, Thanos, NPC

ID:	FA19
Titel:	Sieges-Bildschirm anzeigen
Beschreibung:	Sobald ein Charakter am Ende einer Zugphase alle sechs Infinity Stones in seinem Inventar hat, endet die Partie sofort, und der Spieler dieses Superhelden hat gewonnen.
Begründung:	Ein Spieler hat die Gewinnbedingung erreicht und gewinnt das Spiel.
Abhängigkeiten:	FA2
Akteure:	Spieler

ID:	FA20
Titel:	Hintergrund anzeigen
Beschreibung:	Das Spiel ist thematisch in der Savanne Wakandas angesiedelt. Entsprechend muss der Hintergrund texturiert sein.
Begründung:	Der Hintergrund gibt einen Kontext für das Spiel
Abhängigkeiten:	FA10, FA3
Akteure:	Spieler

ID:	FA21
Titel:	Figuren anzeigen
Beschreibung:	Spieler und NPCs haben Charaktere auf dem Feld, diese werden angezeigt.
Begründung:	Um einen Überblick über das Spielgeschehen zu gewährleisten, muss die UI die Positionen der Charaktere anzeigen.
Abhängigkeiten:	FA9, FA11, FA17
Akteure:	NPC, Spieler

ID:	FA22
Titel:	Hindernisse anzeigen
Beschreibung:	Hindernisse können über das Spielfeld verteilt sein. Über Hindernisse kann nicht gegangen werden.
Begründung:	Für eine Routenplanung muss die Topologie der Karte bekannt sein.
Abhängigkeiten:	FA10
Akteure:	Spieler

ID:	FA23
Titel:	HP der Felsen anzeigen
Beschreibung:	Felsen können zerstört werden. Am Anfang haben sie 100 HP, diese können durch angriffe reduziert werden. Bei 0 werden sie zerstört.
Begründung:	Diese Anzeige ist für taktische Entscheidungen wichtig.
Abhängigkeiten:	FA10, FA22
Akteure:	Spieler

ID:	FA24
Titel:	Animationen
Beschreibung:	Möglichst alle Handlungen die auf dem Spielfeld geschehen sollen animiert werden.
Begründung:	Damit kann der Spieler eine Veränderung wahrnehmen (und es sieht gut aus)
Abhängigkeiten:	FA3
Akteure:	Spieler

ID:	FA25
Titel:	Helden auswählen
Beschreibung:	Der Spieler wählt seine 6 Charaktere aus 12 vom Server vorgegebenen.
Begründung:	Der Spieler soll seine Charaktere selber bestimmen können.
Abhängigkeiten:	FA2, FA3
Akteure:	Spieler, Server

ID:	FA26
Titel:	Reihenfolge anzeigen
Beschreibung:	Die Reihenfolge der nächsten Charaktere wird als UI angezeigt.
Begründung:	Damit kann der Spieler im Voraus planen welche Züge er machen will.
Abhängigkeiten:	FA3, FA17
Akteure:	Spieler

ID:	FA27
Titel:	Movement und Action Points anzeigen
Beschreibung:	Ein Charakter hat bei seinem Zug eine bestimmte Anzahl an Movement und Action Points. Ihm soll nun angezeigt werden wie viele ihm zur Verfügung stehen
Begründung:	Damit kann der Spieler besser planen was er noch tun kann.
Abhängigkeiten:	FA3, FA11, FA28, FA29
Akteure:	Spieler

ID:	FA28
Titel:	Movement und Action Points benutzen
Beschreibung:	Der Spieler steuert über ein UI wie er Action und Movement Points einsetzt und wie sich der Charakter dementsprechend verhält.
Begründung:	Der Spieler muss die Möglichkeit haben seine Movement und Action Points einzusetzen.
Abhängigkeiten:	FA3, FA11, FA27, FA29
Akteure:	Spieler

ID:	FA29
Titel:	Möglichkeiten einschränken
Beschreibung:	Die Benutzung von Action und Movement Points wird eingeschränkt sobald keine mehr vorhanden sind oder er nicht an der Reihe ist.
Begründung:	Da der Spieler nicht unbegrenzt Action und Movement Points besitzt und nicht beliebig einsetzen darf, muss seine Steuerung eingeschränkt werden.
Abhängigkeiten:	FA3, FA11, FA27, FA28
Akteure:	Spieler

ID:	FA30
Titel:	Goose
Beschreibung:	Zu Beginn von Runden 1 bis 6 übergibt sich dieser Charakter auf ein zufälliges freies Feld und erbricht einen der Infinity Stones, der noch nicht bereits im Spiel ist.
Begründung:	Die Infinity Stones sollten nicht direkt, sondern mit der Zeit auf dem Feld platziert werden.
Abhängigkeiten:	FA9, FA10, FA14, FA15, FA21
Akteure:	Server, Goose, NPC

ID:	FA31
Titel:	Stan Lee
Beschreibung:	Stan Lee ist der Kopf hinter Marvel. Er erscheint zu Beginn von Runde sieben neben einem zufälligen ausgeknockten Charakter und setzt die Anzahl seiner Leben wieder auf die maximale Anzahl zurück. Falls es keine ausgeknockten Charakter gibt, dann wird der mit den geringsten HP geheilt. Anschließend ruft Stan "Excelsior" und verschwindet.
Begründung:	Forderung des Lastenhefts.
Abhängigkeiten:	FA10, FA15, FA21
Akteure:	Server, Stan Lee, NPC

ID:	FA32
Titel:	Spiel Informationen anzeigen
Beschreibung:	Generelle Informationen werden als UI angezeigt. Zb: <ul style="list-style-type: none">• Rundenummer• Spielernamen• Charakternamen
Begründung:	Damit bekommt der Spieler weitere Informationen die er nutzen kann um seine Züge zu planen.
Abhängigkeiten:	FA3, FA9, FA11
Akteure:	Spieler

ID:	FA33
Titel:	Nachrichten lesen
Beschreibung:	Ankommende Nachrichten vom Server werden richtig gelesen und interpretiert.
Begründung:	Der Server sendet Nachrichten, welche für die Spielbarkeit des Spiels notwendig sind. Dazu zählen unter anderem Handlungen des anderen Spielers oder Handlungen von NPCs.
Abhängigkeiten:	FA2, FA3, FA4
Akteure:	Server

ID:	FA34
Titel:	Nachrichten an den Server senden.
Beschreibung:	Jede Handlung des Spielers wird als Nachricht an den Server übertragen.
Begründung:	Da die Kommunikation der beiden Spieler über den Server läuft, muss jede Handlung des Spielers als Nachricht an den Server übertragen werden.
Abhängigkeiten:	FA2, FA3, FA4
Akteure:	Spieler, Server

ID:	FA35
Titel:	Hauptmenu UI
Beschreibung:	Das Spiel soll ein Hauptmenü mit allen notwendigen Funktionen des Spiels haben. Funktionen sind u.a: <ul style="list-style-type: none">• Spiel beitreten• Spiel erstellen• Editor öffnen• Anwendung beenden
Begründung:	Damit kann der Spieler entscheiden was er tun will.
Abhängigkeiten:	FA3
Akteure:	Spieler

ID:	FA36
Titel:	Spiel erstellen
Beschreibung:	Der Spieler kann ein benutzerdefiniertes Spiel erstellen.
Begründung:	Möglichkeit ein Spiel zu starten.
Abhängigkeiten:	FA3, FA35
Akteure:	Spieler

ID:	FA37
Titel:	Spiel beitreten
Beschreibung:	Ein Spieler kann einem Spiel aus einer Liste von verfügbaren Spielen beitreten.
Begründung:	Damit finden sich 2 Spieler für ein Spiel
Abhängigkeiten:	FA3, FA35
Akteure:	Spieler, Server

ID:	FA38
Titel:	Anwendung beenden
Beschreibung:	Eine saubere Möglichkeit die Anwendung zu beenden
Begründung:	Damit ist es für den Spieler einfach die Anwendung auch wieder zu beenden.
Abhängigkeiten:	FA3, FA35
Akteure:	Spieler

ID:	FA39
Titel:	Spiel pausieren
Beschreibung:	Falls ein Spieler eine Pausierung der Partie wünscht, bis ein Spieler anzeigt, dass er weiterspielen will. KI-Clients können keine Pause einlegen.
Begründung:	Verbindliche Anforderung im Lastenheft
Abhängigkeiten:	FA43, FA3
Akteure:	Spieler

ID:	FA40
Titel:	Spiel personalisieren
Beschreibung:	Der Spieler kann bei der Erstellung eines Spiels entscheiden wie das Spiel aussehen soll. Dabei kann er eine im Editor erstellte Konfiguration auswählen.
Begründung:	Bringt mehr Abwechslung
Abhängigkeiten:	FA3, FA5, FA35
Akteure:	Spieler

ID:	FA41
Titel:	Spieler Timeout
Beschreibung:	Wenn einer der Spieler wegen eines Verbindungsabbruchs nicht mehr mitspielen kann wird erstmal gewartet ob der Spieler wieder zurückkehrt.
Begründung:	Dadurch wird ein Spiel nicht direkt abgebrochen.
Abhängigkeiten:	FA39, FA42
Akteure:	Spieler

ID:	FA42
Titel:	Spieler reconnect
Beschreibung:	Wenn ein Spieler die Verbindung verloren hat, während er noch in einem Spiel war kann er dem Spiel wieder beitreten.
Begründung:	Dadurch wird ein Spiel nicht direkt abgebrochen.
Abhängigkeiten:	FA39, FA41
Akteure:	Spieler, Server

ID:	FA43
Titel:	Pause Button
Beschreibung:	Falls ein Spieler eine Pausierung der Partie wünscht, bis ein Spieler anzeigt, dass er weiterspielen will. Hier wird das passende UI Element geliefert.
Begründung:	Verbindliche Anforderung im Lastenheft
Abhängigkeiten:	FA39
Akteure:	Spieler

ID:	FA44
Titel:	Spiel auswählen
Beschreibung:	Auswahl eines Spiels in der Aufbauphase aus einer Liste aller erstellten noch nicht begonnenen Spiele.
Begründung:	Somit können Spieler gezielt Spielen beitreten.
Abhängigkeiten:	FA36
Akteure:	Spieler

ID:	FA45
Titel:	Spiel zusehen
Beschreibung:	Zuschauer sollten dem Spiel beitreten können
Begründung:	Wenn das Spiel bereits zu voll ist, kann zugesehen werden.
Abhängigkeiten:	FA10, FA36
Akteure:	Spieler

Nichtfunktionale Anforderungen:

ID:	QA1
Titel:	Berechnungszeit
Beschreibung:	Spieler sollten nicht länger als eine Sekunde auf das Resultat ihrer Handlung warten müssen, daher sollte der Server möglichst effizient implementiert sein.
Begründung:	Übermäßige Wartezeiten verwirren und desorientieren Spieler.
Abhängigkeiten:	
Akteure:	
Priorität:	5

ID:	QA2
Titel:	Crashes
Beschreibung:	Nur jedes 100. Spiel sollte crashen.
Begründung:	Durch dieses QA wird sichergestellt, dass Spieler nicht an Fehlern die Lust am Spielen verlieren.
Akteure:	
Priorität:	3

ID:	QA3
Titel:	Zuschauer
Beschreibung:	Mindestens 10 Zuschauer sollten zusehen können.
Begründung:	Dies ist nötig, um eine Skalierbarkeit gewährt zu leisten.
Akteure:	
Priorität:	2

ID:	QA4
Titel:	Kommentar Sprache
Beschreibung:	Als Kommentar Sprache sollte Englisch verwendet werden.
Begründung:	Die Kommentare im Code sollten von jedem Informatiker verstanden werden.
Abhängigkeiten:	
Akteure:	Entwickler
Priorität:	4

ID:	QA5
Titel:	Dokumentensprache
Beschreibung:	Dokumente können in Deutsch oder in Englisch geschrieben werden.
Begründung:	Da die Dokumente nur von einer begrenzten Anzahl an Personen gelesen werden. Außerdem ist von diesen Personen zu erwarten das sie deutsch sprechen können.
Abhängigkeiten:	
Akteure:	Entwickler
Priorität:	7

ID:	QA6
Titel:	UI Sprache
Beschreibung:	Die von der UI verwendeten Sprache kann auch Deutsch oder Englisch sein.
Begründung:	Da das Spiel auch nur von einer begrenzten Anzahl an Personen gespielt werden wird, kann die Sprache auch deutsch sein. Außerdem ist von diesen Personen zu erwarten das sie deutsch sprechen können.
Abhängigkeiten:	
Akteure:	Entwickler
Priorität:	7

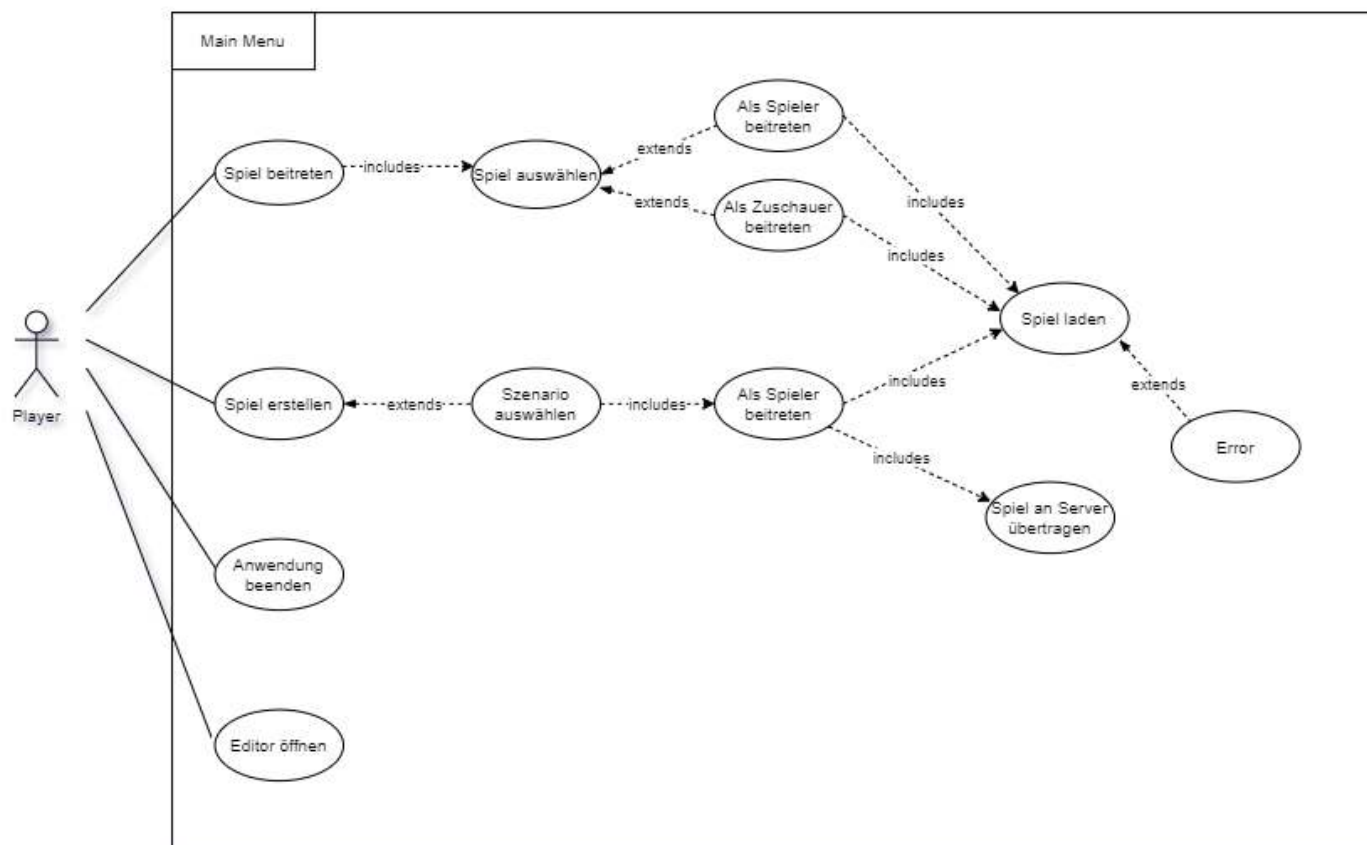
ID:	QA7
Titel:	Plattform
Beschreibung:	<p>Die Komponenten Benutzer-Client und Editor müssen jeweils auf mindestens einer der folgenden Plattformen laufen:</p> <ul style="list-style-type: none"> • Eine der verbreiteten, aktuellen Linux-Distributionen • Eine aktuelle Version von Microsoft Windows • Basierend auf Webtechnologien, in einem aktuellen, standardkonformen Browser <p>Die Komponenten KI-Client und Server müssen als Docker-Container lauffähig sein.</p>
Begründung:	Dadurch kann der Server-Code auf dem passenden Server laufen. Außerdem hat ein großer Teil der Personen, welche das Spiel spielen wollen die Möglichkeit dazu
Abhängigkeiten:	
Akteure:	Entwickler, Spieler
Priorität:	7

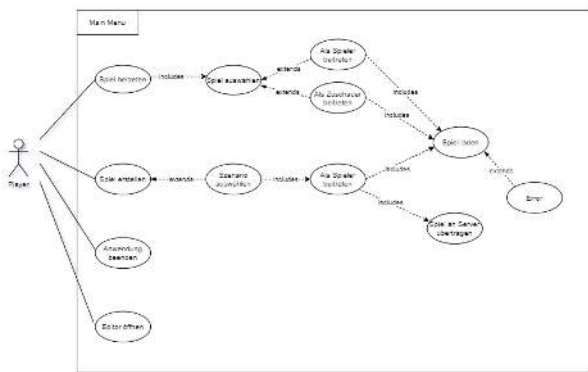
ID:	QA8
Titel:	Sicherheit / Datenschutz
Beschreibung:	Der Austausch von Informationen soll gegen mögliche Angriffe geschützt sein.
Begründung:	Dadurch könnten die Aktionen verändert und ein unfaires Spiel erzeugt werden.
Abhängigkeiten:	
Akteure:	Entwickler
Priorität:	7

ID:	QA9
Titel:	Wartbarkeit / Portabilität / Änderbarkeit / Testbarkeit / Verständlichkeit
Beschreibung:	Der Code soll so geschrieben sein, das einfach Änderungen an ihm vorgenommen werden können. Dafür muss er verständlich geschrieben sein. Tests sollen möglich sein und daraus folgend sollte es gut möglich sein den Code anzupassen.
Begründung:	Der Code muss einfach zu verändern sein um ein Gelingen des Projekts sicher zu stellen.
Abhängigkeiten:	
Akteure:	Entwickler
Priorität:	1

Anwendungsfälle:

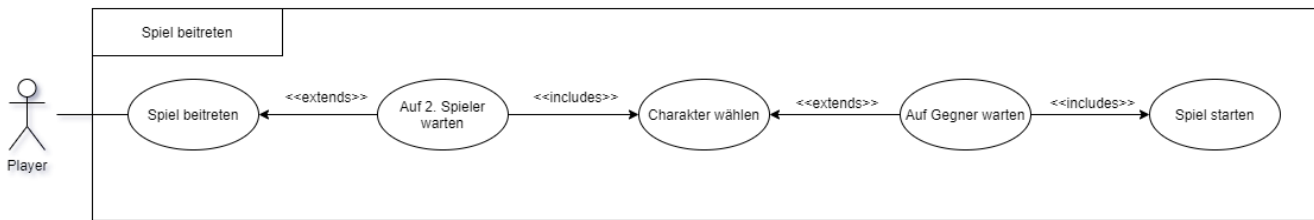
Main Menu:





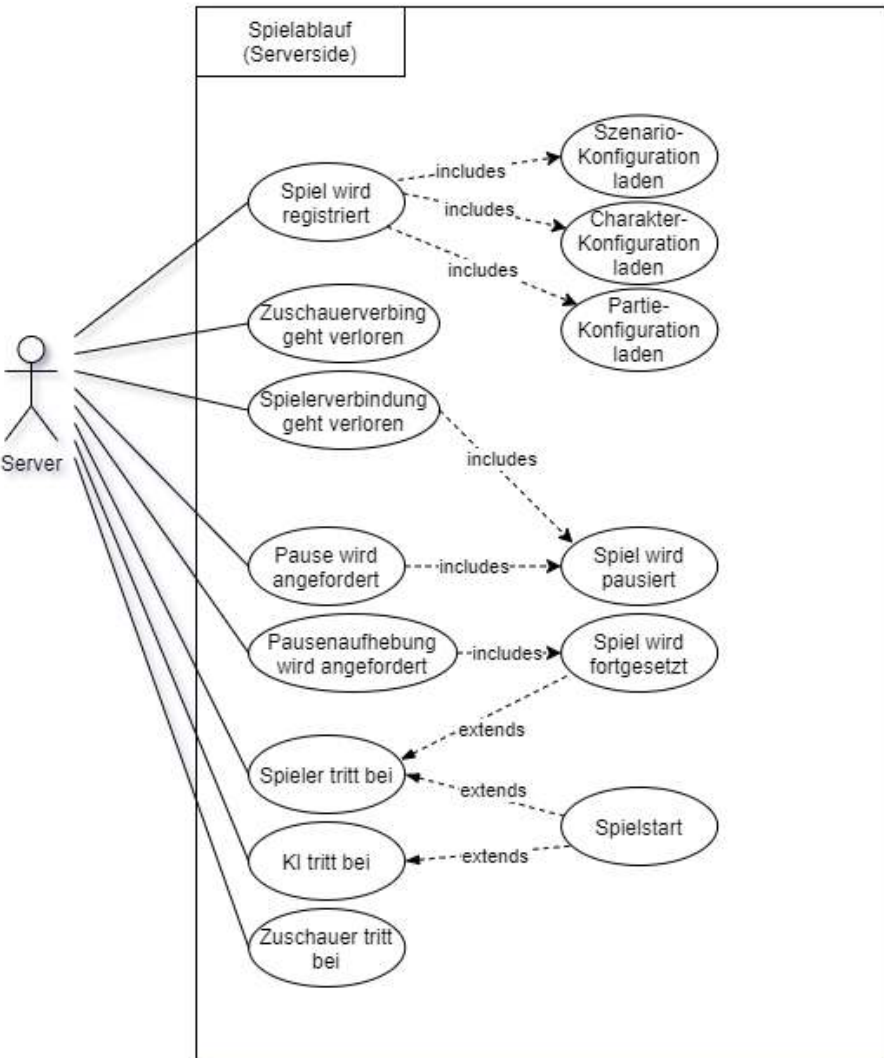
Anwendungsfall	Beschreibung	FAs
Spiel beitreten	Der Spieler/Zuschauer kann einem bereits erstellten noch nicht laufendem Spiel beitreten, ohne jedoch Eistellungen zu dem Spiel tätigen zu können (Anzahl der Spieler, Schwierigkeit etc)	FA37
Spiel auswählen	Auswahl eines Spiels in der Aufbauphase aus einer Liste aller erstellten noch nicht begonnenen Spiele	FA44
Als Spieler beitreten	Beitritt zu einem Spiel als aktiver Akteur und Spieler, wobei man berechtigt ist Züge und Aktionen auf dem Spielfeld mit Helden auszuführen	FA37, FA44
Als Zuschauer beitreten	Beitritt zu einem Spiel als passiver Akteur und Zuschauer, wobei man keine Berechtigung hat ins Spielgeschehen einzugreifen, dieses aber mitzuverfolgen	FA1, FA2, FA3
Spiel laden	Lädt den aktuellen Fortschritt eines Spiels, zu Beginn verwendet um ein neues Spiel zu laden	FA16
Error	Fehlermeldung aufgrund eines Fehlers im Spiel und/oder eine nicht zugelassene Aktion	FA2, FA33, FA34
Spiel erstellen	Der Spieler erstellt ein neues Spiel, wobei nur er allein die Rechte hat gewisse Werte des Spiels festzulegen, wie Anzahl der Spieler oder der Schwierigkeitsgrad	FA36
Szenario auswählen	Auswahl zwischen mehreren Spielmöglichkeiten	FA40
Spiel an Server übertragen	Überträgt den aktuellen Spielfortschritt an den Server	FA2, FA34
Anwendung beenden	Beendet das Spiel und schließt das Fenster	FA38
Editor öffnen	Öffnet den Editor Screen	FA5

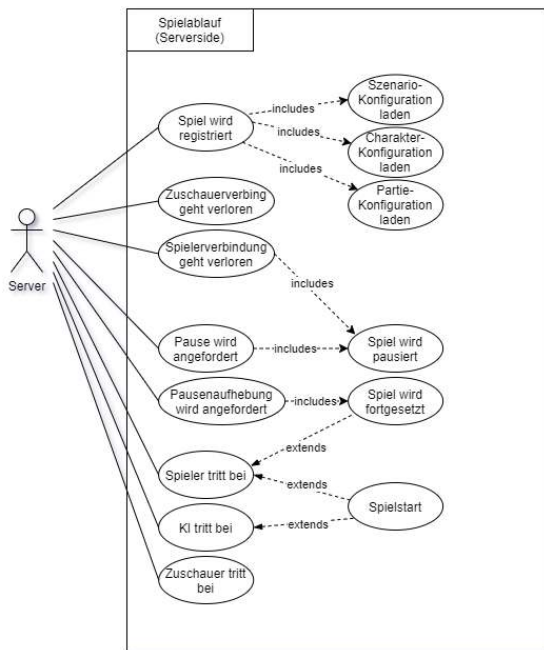
Spiel beitreten:



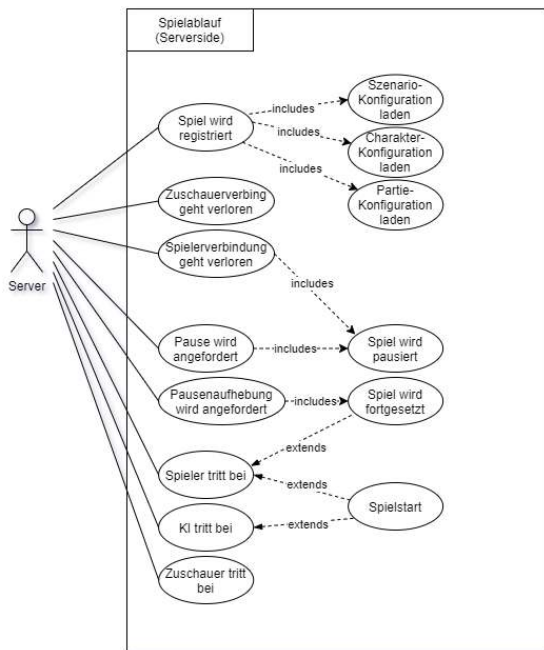
Anwendungsfall	Beschreibung	FAs
Auf 2. Spieler warten	Falls der Spieler bis jetzt der einzige spielende Spieler (Zuschauer ausgeschlossen) ist, wird auf einen weiteren Spieler gewartet. Der Spieler hat hier zusätzlich die Möglichkeit eine KI als Gegner hinzuzufügen.	FA36, FA37
Charakter wählen	Auswahl von 6 Helden (gilt universell, da jeder Held nur einmal existiert und nicht mehrmals in einem Team oder in mehreren Teams gleichzeitig kämpfen kann). Der Spieler hat hierbei außerdem nur die Wahl aus 12 Helden.	FA25
Auf Gegner warten	Wenn die 6 Helden fertig gewählt wurden muss evtl noch auf den anderen Spieler gewartet werden.	FA36, FA37
Spiel starten	Das Spiel kann gestartet werden.	FA1, FA10, FA16, FA20, FA21, FA22, FA23, FA25, FA26, FA32

Spielablauf (Serverside):



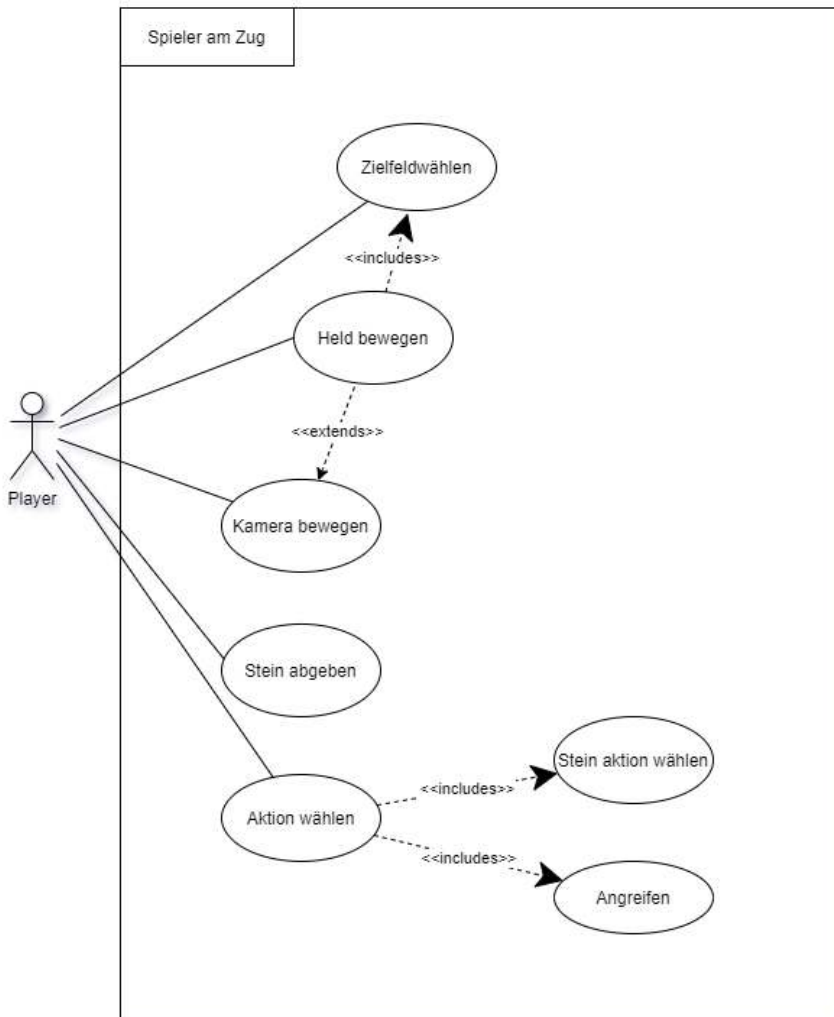


Anwendungsfall	Beschreibung	FAs
Spiel wird registriert	Ein neues Spiel wird erstellt. Die Konfig Dateien werden empfangen und werden nun interpretiert. Das Spiel wird nun zur Verfügung gestellt.	FA1, FA2, FA4, FA5, FA14, FA16
Szenario-Konfiguration laden	Hierbei werden generelle Einstellungen geladen.	FA1, FA2, FA4, FA5, FA14, FA16
Charakter-Konfiguration laden	Die Helden werden mit ihren Werten geladen.	FA11
Partie-Konfiguration laden	Die Karte wird geladen.	FA5
Zuschauer-Verbindung geht verloren	Ein Zuschauer beendet die Verbindung (beabsichtigt oder wegen einem Verbindungsfehler)	
Spieler-Verbindung geht verloren	Ein Spieler verliert die Verbindung. Hierbei wird das Spiel pausiert.	FA41
Pause wird angefordert	Einer der beiden Spieler pausiert das Spiel.	FA39, FA43
Pausen Aufhebung wird angefordert	Einer der beiden Spieler kann nun die Pause wieder aufheben. Das muss hierbei nicht der Spieler sein der pausiert hat.	FA39, FA43



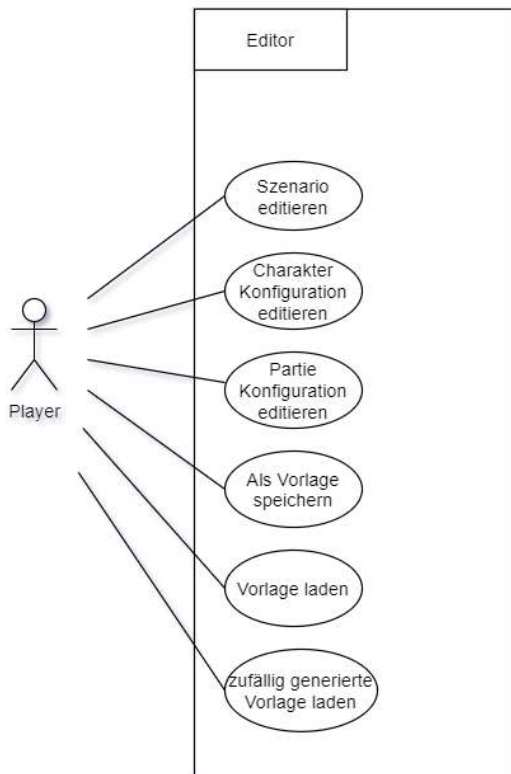
Anwendungsfall	Beschreibung	FAs
Spiel wird pausiert	Der Spielablauf wird pausiert. Das wird mit einem Dialog angezeigt, dies ist nur Benutzer-Clients möglich, im Gegensatz zu KI-Clients.	FA41
Spiel wird fortgesetzt	Das Spiel wird fortgesetzt.	FA42
Spieler tritt bei	Ein Spieler tritt dem Spiel bei. Hierbei wird nun entweder auf einen 2. Spieler oder Bot gewartet oder das Spiel gestartet.	FA3
KI tritt bei	Ein Bot (KI) tritt dem Spiel bei. Hierbei wird nun entweder auf einen 2. Spieler oder Bot gewartet oder das Spiel gestartet.	FA4
Zuschauer tritt bei	Ein Zuschauer tritt bei. Das kann zu jedem Zeitpunkt im Spiel passieren.	FA37
Spielstart	Wenn 2 Spieler oder Bots beigetreten sind wird das Spiel gestartet.	FA1, FA10, FA16, FA20, FA21, FA22, FA23, FA25, FA26, FA32

Spieler am Zug:



Anwendungsfall	Beschreibung	FAs
Held bewegen	Der Held nutzt einen oder mehrere Movement Points um sich auf einem anderen Feld zu platzieren.	FA27, FA28, FA29
Zielfeld wählen	Das Feld auf das sich der Held bewegen soll wird gewählt.	FA15
Kamera bewegen	Die Kamera wird bewegt, um einen bestimmten Teil des Spielfelds anzuschauen oder zu fokussieren.	FA24
Stein abgeben	Ein Stein der in dem Besitz des aktuellen Charakters ist wird an einen anderen Charakter abgegeben.	FA14, FA25
Aktion wählen	Der Spieler wählt was er mit einem Action Point machen will.	FA28
Stein Aktion wählen	Wenn der Charakter Infinity Steine besitzt kann er wählen ob und welchen er benutzen möchte.	FA14
Angreifen	Er kann einen Nah- oder Fernkampf Angriff machen, der Angriff und der am Ziel zugefügte Schaden wird visualisiert, sodass sichtbar ist, welches Ziel angegriffen wurde.	FA28

Editor:

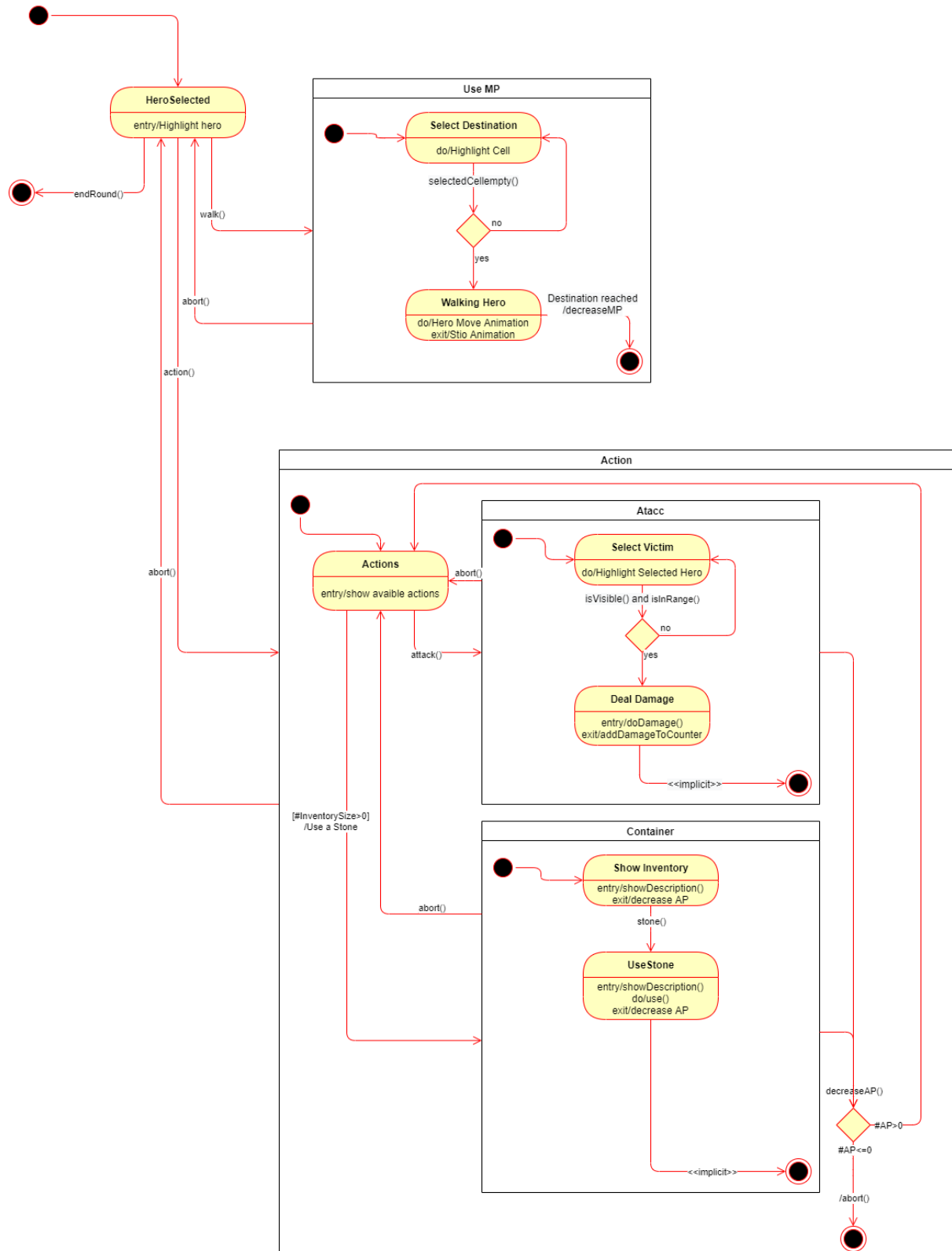


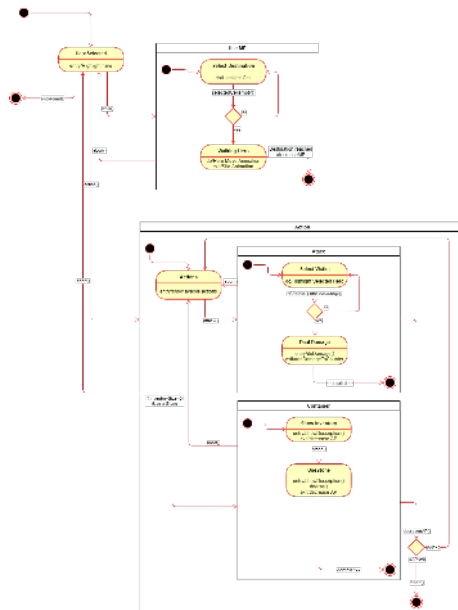
Anwendungsfall	Beschreibung	FAs
Szenario editieren	Die Map kann hier angepasst werden. Es kann sowohl Größe als auch aussehen eingestellt werden.	FA5
Charakter Konfiguration editieren	Hier können die Werte der einzelnen Charakter angepasst werden.	FA5, FA11
Partie Konfiguration editieren	Die generellen Spieleinstellungen werden hier eingestellt.	FA5
Als Vorlage speichern	Hier kann eine Konfiguration gespeichert werden. Sie kann ab sofort bei der Erstellung eines neuen Spiels verwendet werden.	FA2, FA34
Vorlage laden	Eine Konfiguration wird geladen und kann nun weiter bearbeitet werden.	FA2, FA33
zufällig generierte Vorlage laden	Eine zufällige Konfiguration wird gesetzt und kann nun noch bearbeitet werden.	FA2, FA33

Abläufe im System:

Zustandsdiagramme:

Player am Zug:

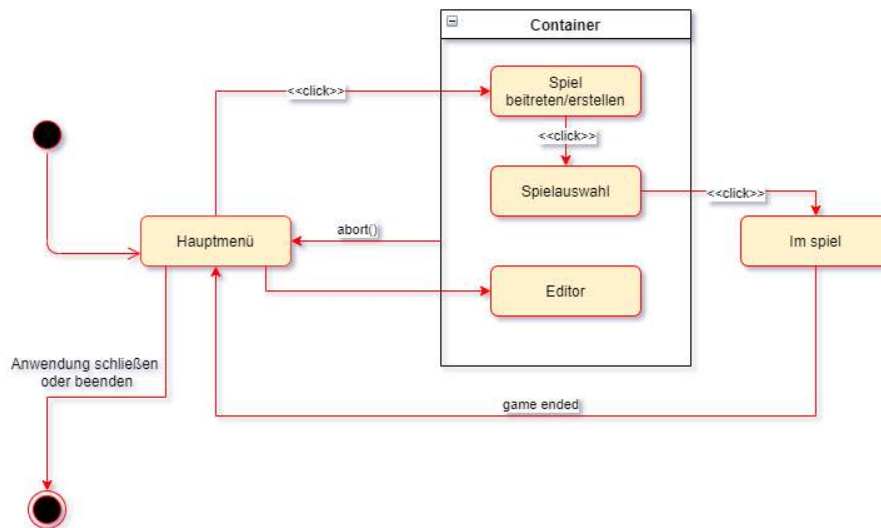




Der Spieler hat nach der Auswahl eines Charakters die Wahl, sich zu bewegen, oder eine Aktion durchzuführen. Wenn der Spieler sich bewegen will, wird überprüft, ob das angegebene Spielfeld leer ist. Ist dies der Fall, bewegt sich der entsprechende Charakter zum angegebenen Spielfeld. Entscheidet der Spieler sich zu einer Aktion, kann er wählen zwischen dem Attackieren eines anderen Charakters oder dem Benutzen eines Infinity Stones. Im Falle dessen, dass der Spieler erstes wählt, kann ein Opfer gewählt werden, dass wenn sichtbar und in Reichweite Schaden bekommt. Entscheidet der Spieler sich hingegen für zweiteres, so muss er in einem Dialog einen der Infinity stones in seinem Besitz auswählen und benutzen. Egal, was passiert, am Ende werden die Action Points reduziert.

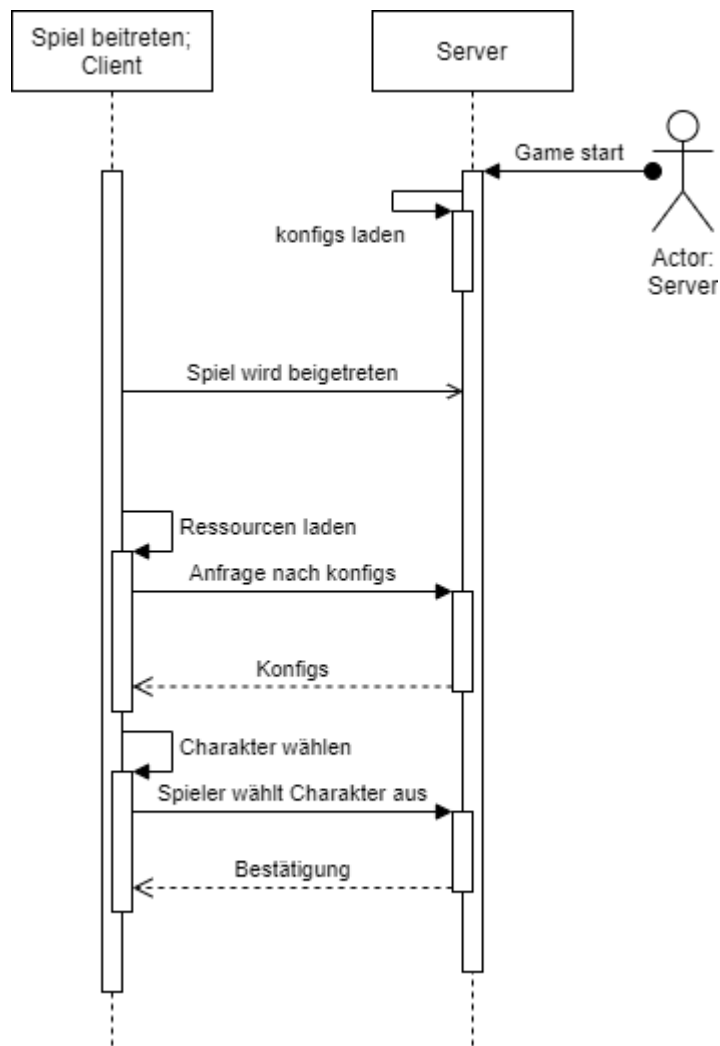
Nach der Ausführung von Sowohl eines Movement Points als auch eines Action Points hat der Spieler die Möglichkeit weitere Action und Movement Points zu verwenden.

Hauptmenü:



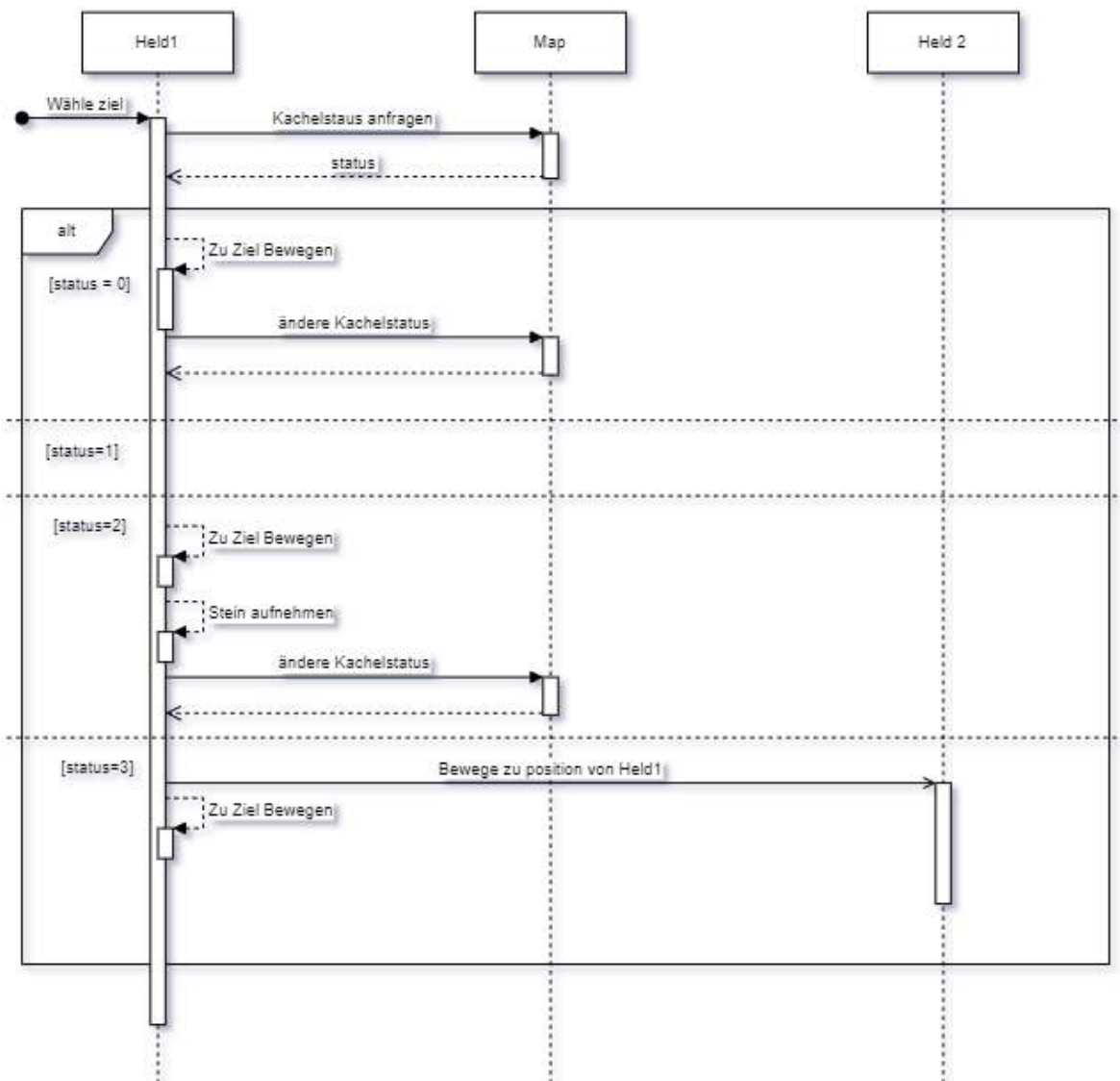
Im Hauptmenü hat der Spieler durch klicken der linken Maustaste die Möglichkeit auszuwählen zwischen den Optionen ein Spiel zu erstellen oder zu erstellen, ein Spiel auszuwählen, oder in den Editor zu wechseln.

Sequenzdiagramme:



Der Spieler tritt dem Spiel bei und überträgt seinen Namen in diesem Zug an den Server. Wenn ein Spieler einem bereits laufenden Spiel beitrifft, sind alle Konfigurationsdateien bereits geladen. Der Client empfängt diese dann über das Netzwerk vom Server und lädt sie selbst. Anschließend kann vom Spieler ein Charakter ausgewählt werden, diese Auswahl wird vom Client an den Server weitergegeben, welcher dann die Auswahl bestätigt, sollte sie valide sein.

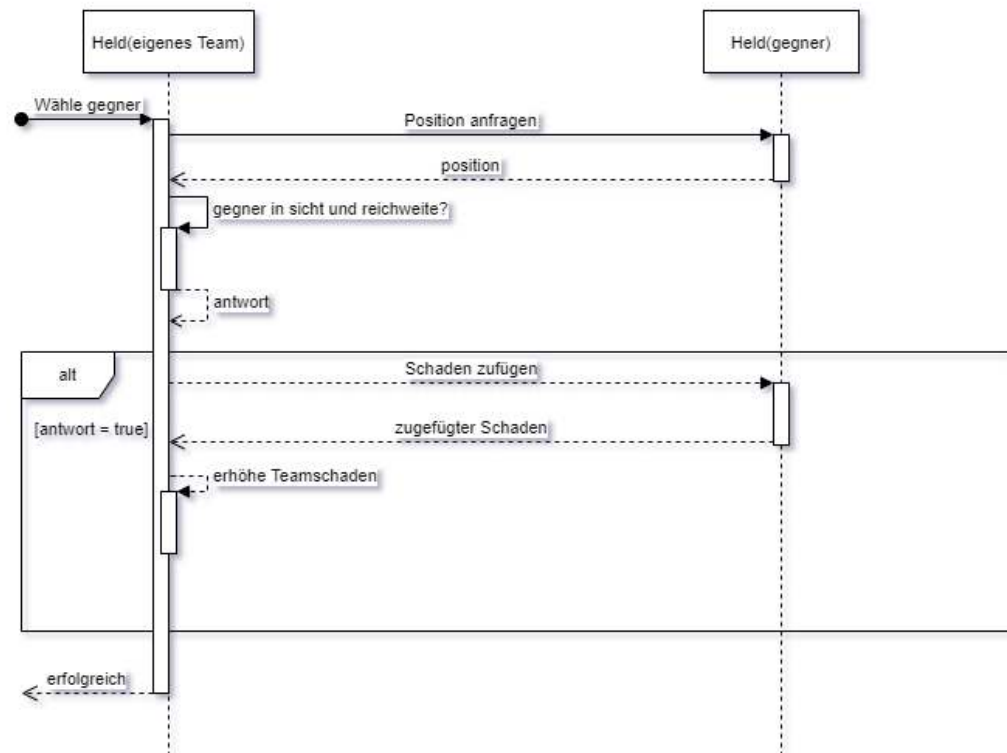
Player bewegt sich



status	Status ist ein ganzzahliger wert der eindeutig beschreibt was sich auf dem Feld befindet:	0 = nichts 1 = Stein 2 = Infinity Stone 3 = Held
---------------	---	---

Ein Spieler wählt eine Kachel aus und es wird nun bestimmt was sich in dieser Kachel (oder auch Feld genannt) befindet und dementsprechend wird entschieden was nun passiert.

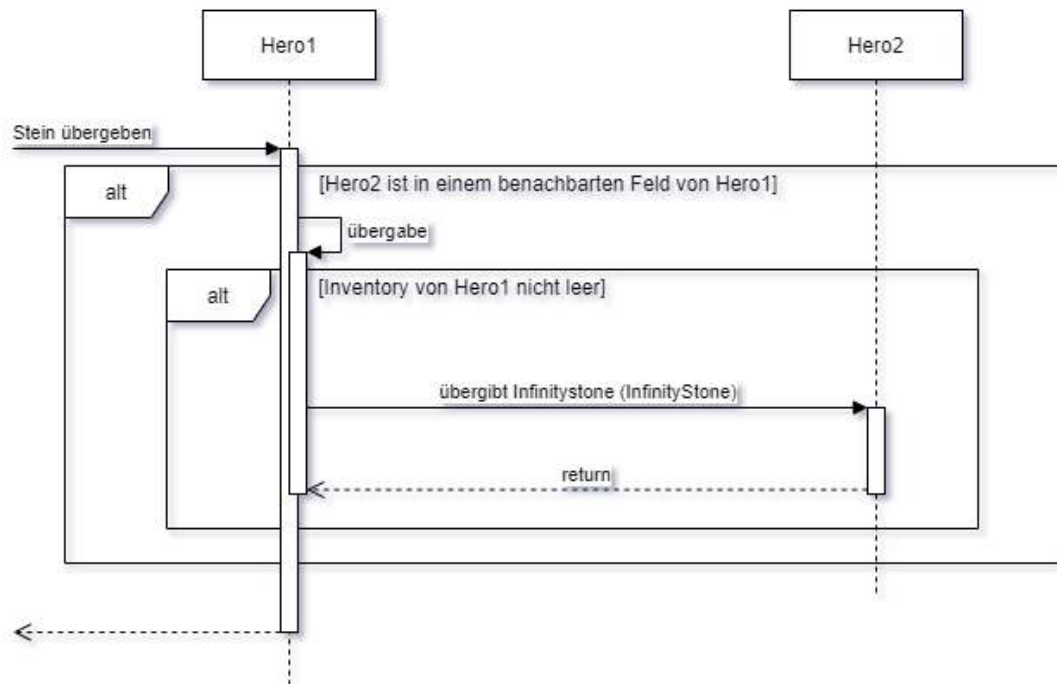
Player attacks enemy

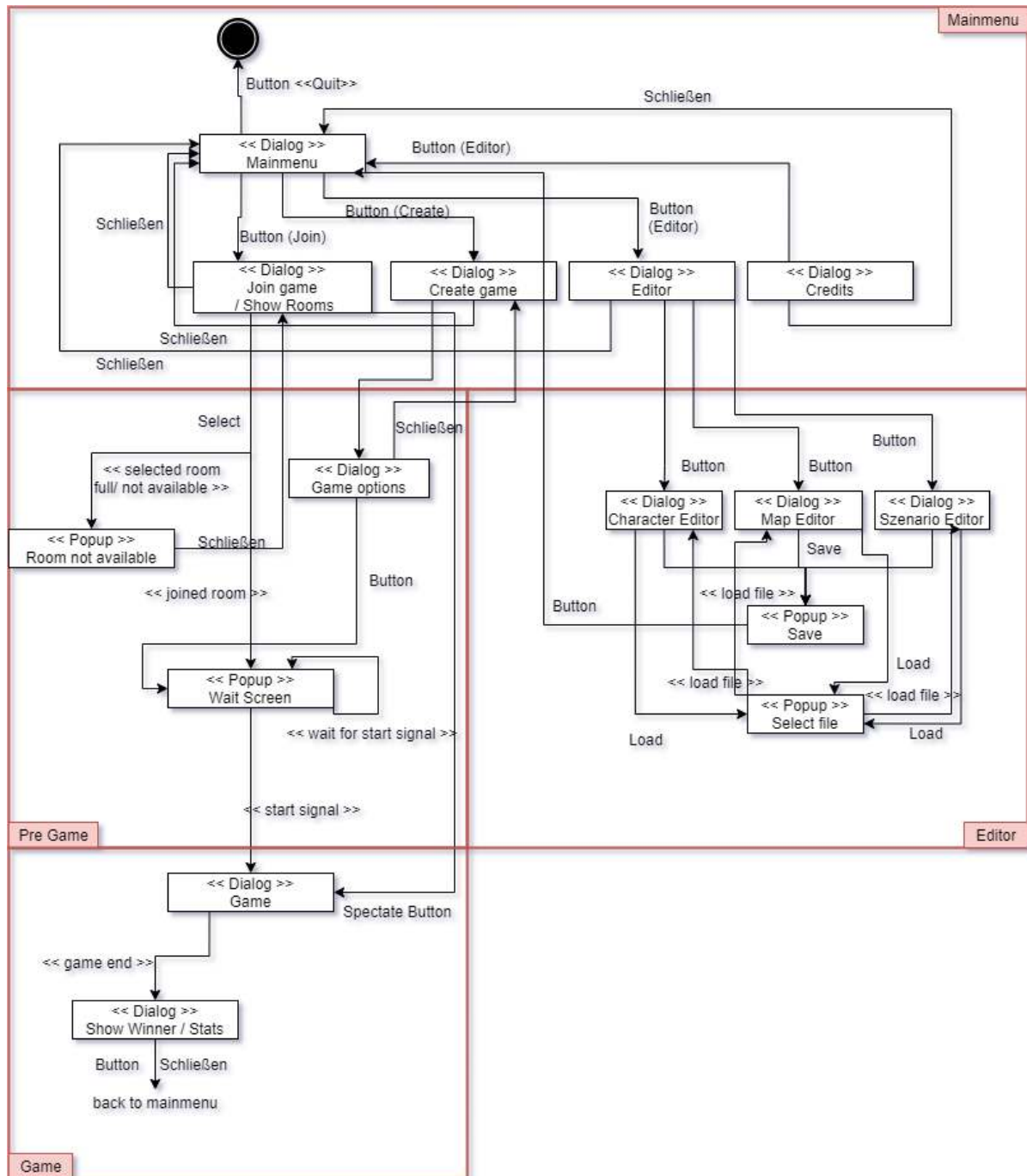


Teamschaden	Es wird gespeichert wieviel Schade der Spieler am ende der Partie insgesamt zugefügt hat, dieser wert wird auch als tiebreaker verwendet	
return	Der Wert ob der Angriff erfolgreich war wird returned	true = Angriff war erfolgreich false = Angriff ist fehlgeschlagen

Ein Player wählt mit einem Charakter einen Gegner aus. Wenn dieser in Reichweite ist, wird ihm Schaden zugefügt und der Teamschaden wird erhöht.

Held übergibt Infinitystone





Mainmenu

Das Hauptmenü navigiert den Nutzer durch die verschiedenen Menüeinträge. Durch die Einträge "Join Game", "Create Game" und "Editor" gelangt man in eine tiefere Auswahlebene des ausgewählten Eintrags. Credits zeigt eine Auflistung der Personen an, die an dem Projekt mitgewirkt haben.

Pregame

Der Pregame Abschnitt beschreibt das Dialogverhalten bis zum Beitritt in das Spiel. Dorthin gelangt man durch das Erstellen eines neuen Raumes der zuerst konfiguriert werden muss, oder durch das Auswählen eines existierenden Raumes. Vom Raum Auswahldialog kann auch die "Spectate" Taste gedrückt werden, wodurch man ein laufendes Spiel beobachten kann. Erst wenn genügend Spieler (2 Spieler, oder 1 Spieler, 1 KI) verbunden sind und das Startsignal erteilt wurde, gelangt man zum Spielszenario, bei welchem das eigentliche Spiel angezeigt wird und gespielt werden kann.

Editor

Im Editor Abschnitt hat man die Möglichkeit Character- Map und Szenario Editoren zu öffnen in welchen die jeweiligen Konfigurationsmöglichkeiten angezeigt werden, mit denen der Nutzer die Aspekte anpassen kann. Die Schaltflächen "Load" und "Save" ermöglichen ein abspeichern und bearbeiten von Konfigurationen.

Game

Im Spielszenario kann der Nutzer mit dem Hauptspiel interagieren und seinem Charakter Anweisungen geben. Am Ende des Spiel wird der Gewinner angezeigt.

Graphische Gestaltung und Nutzungskonzept

Hauptmenü:



Farben:

- | | |
|----------------------|---------|
| • Rot (Marvel Logo): | #F0131E |
| • Gelb: | #FECB02 |
| • Schwarz: | #FFFFFF |
| • Weiß: | #000000 |



Das **Hauptmenü** navigiert den Nutzer über die verschiedenen Menüeinträge in die verschiedenen Bereiche des Spiels.

Die Einträge "Join Game", "Create Game" und "Editor" sind die zentralen Bereiche und sollen priorisiert werden gegenüber den anderen Einträgen. Daher sind diese Buttons größer als die restlichen Buttons.

Durch die Einträge "Join Game", "Create Game" gelangt man zu weiteren Darstellungen, welche es ermöglichen ein Spiel zu spielen.

Der Eintrag "Editor" öffnet den Editor, welcher auch ein sehr zentraler Aspekt des Spiels ist.

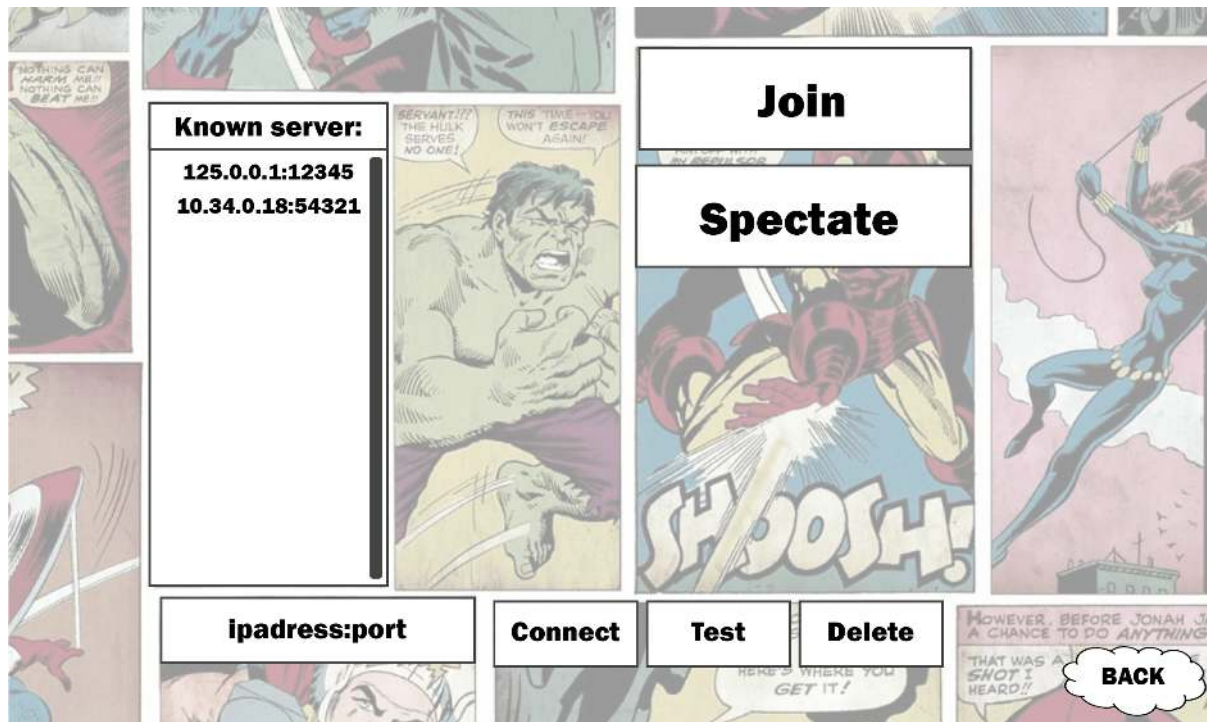
Über "Settings" kommt man in die Einstellungen des Spiels. Es gibt hier noch kein Konzept, welche Einstellungen angepasst werden können. "Credits" zeigt eine Auflistung der Personen an, die an dem Projekt mitgewirkt haben. "Quit" beendet die Anwendung.

Das Design ist sehr einfach gehalten um den Nutzer nicht zu überfordern. Oben in der Mitte ist das Logo zu sehen. Das Logo ist im Stil einer Action bubble aus Comics übernommen. Das Rot der Schrift ist die Farbe des Marvel Logos.

Die Auswahlknöpfe sind auch in diesem Rot.

Der Hintergrund zeigt schwach ein Comic an. Das macht das Hauptmenü weniger langweilig.

Join Game:



Im **Join Game** Menü kann der Nutzer einem Spiel beitreten. Hier kann eine Kombination aus IP-Adresse und Portnummer entweder manuell eingegeben werden, oder aus der Liste der bekannten Server eingefügt werden.

Der Nutzer kann sich nun zu dieser IP über den Button “Connect” verbinden. Er kann auch die Verbindung überprüfen mit dem Button “Test”. Dieser verbindet den Client mit dem Server und baut daraufhin die Verbindung wieder ab. Mit dem Button “Delete” wird die IP aus der Liste gelöscht.

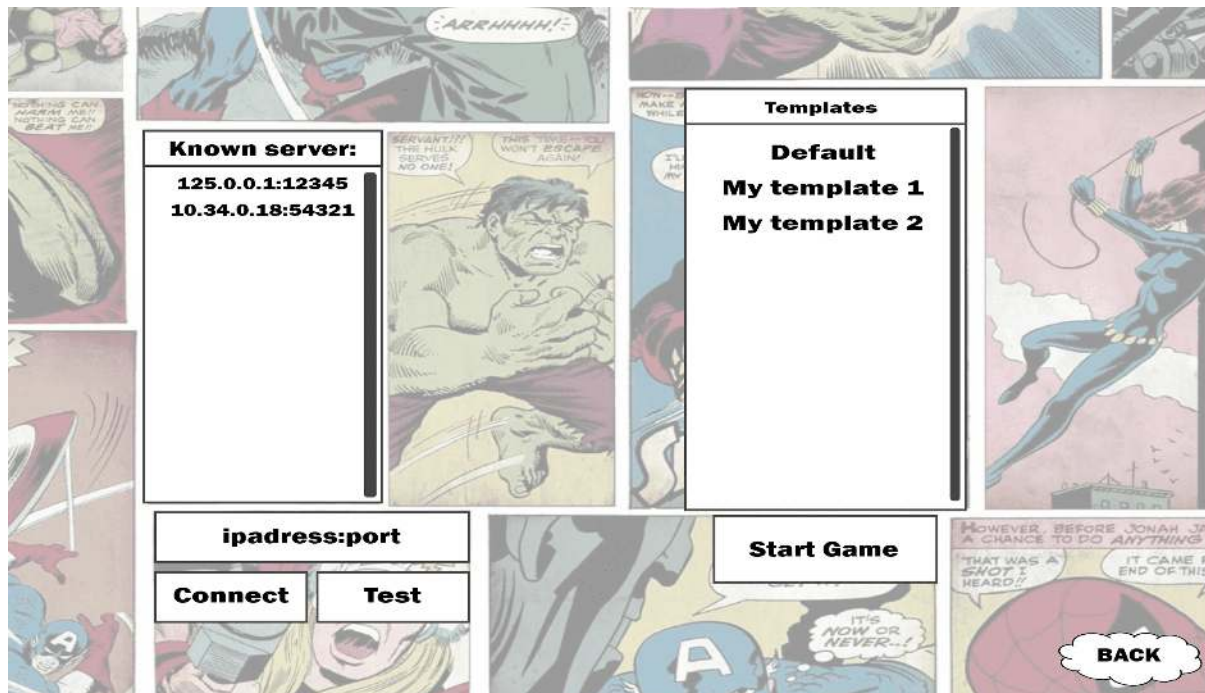
Der Button “Back” navigiert den Nutzer wieder zurück ins Hauptmenü.

Sobald der Client mit einem Server verbunden ist, kann der Nutzer nun über die beiden verbleibenden Buttons “Join” und “Spectate” dem Spiel beitreten oder zuschauen.

Im Hintergrund befindet sich weiterhin ein Auszug aus einem Marvel Comic.

Die einzelnen Buttons, Rechtecke und Fenster sind an die Konturen des Comics angepasst um sich in den Hintergrund einzufügen. Der Ausschnitt des Comic soll nicht der selbe sein wie der im Hauptmenü.

Create Game:



Im **Create Game** Menü kann der Nutzer ein Spiel erstellen. Hier kann eine Kombination aus IP-Adresse und Portnummer entweder manuell eingegeben werden, oder aus der Liste der bekannten Server eingefügt werden um sich mit dem Server, welcher das Spiel hosten soll zu verbinden.

Der Nutzer kann sich nun zu dieser IP über den Button "Connect" verbinden. Er kann auch die Verbindung überprüfen mit dem Button "Test". Dieser verbindet den Client mit dem Server und baut daraufhin die Verbindung wieder ab.

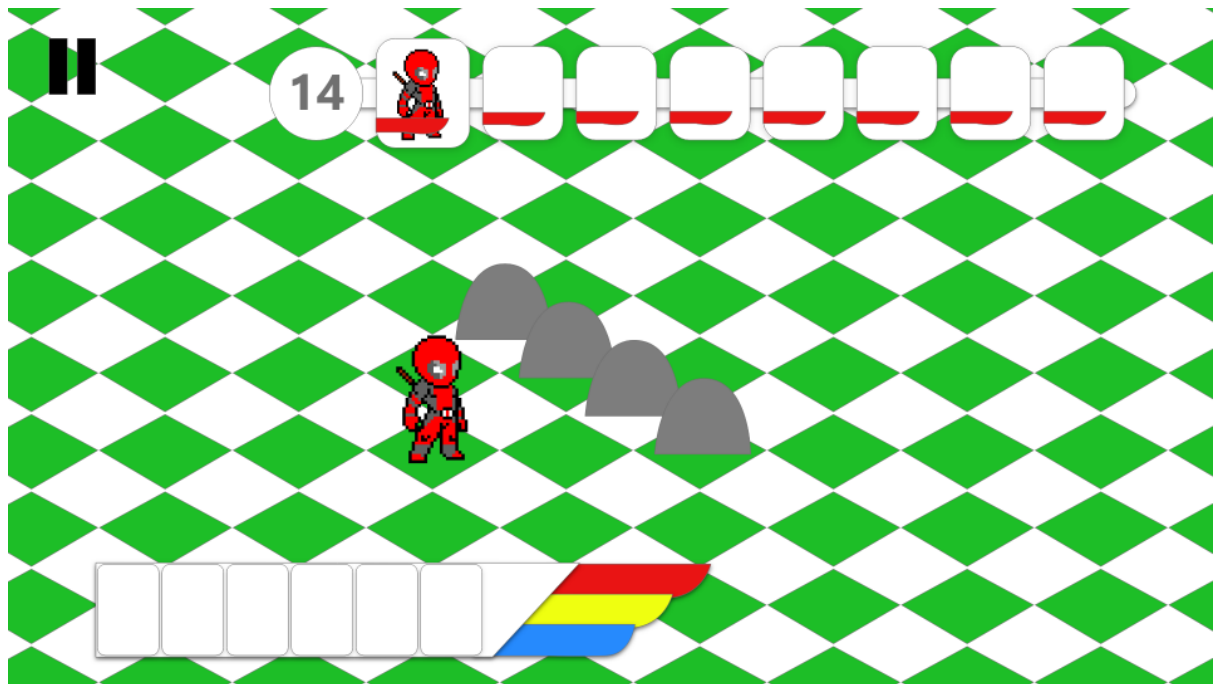
Der Button "Back" navigiert den Nutzer wieder zurück ins Hauptmenü.

Sobald der Client mit einem Server verbunden ist, kann der Nutzer nun über die die Liste an Vorlagen eine Konfiguration für das Spiel auswählen. Hit dem "Start Game" Button wird die Konfiguration bestätigt und das Spiel wird im Server gestartet.

Im Hintergrund befindet sich weiterhin ein Auszug aus einem Marvel Comic.

Die einzelnen Buttons, Rechtecke und Fenster sind an die Konturen des Comics angepasst um sich in den Hintergrund einzufügen. Der Ausschnitt des Comic soll nicht der selbe sein wie der im Hauptmenü.

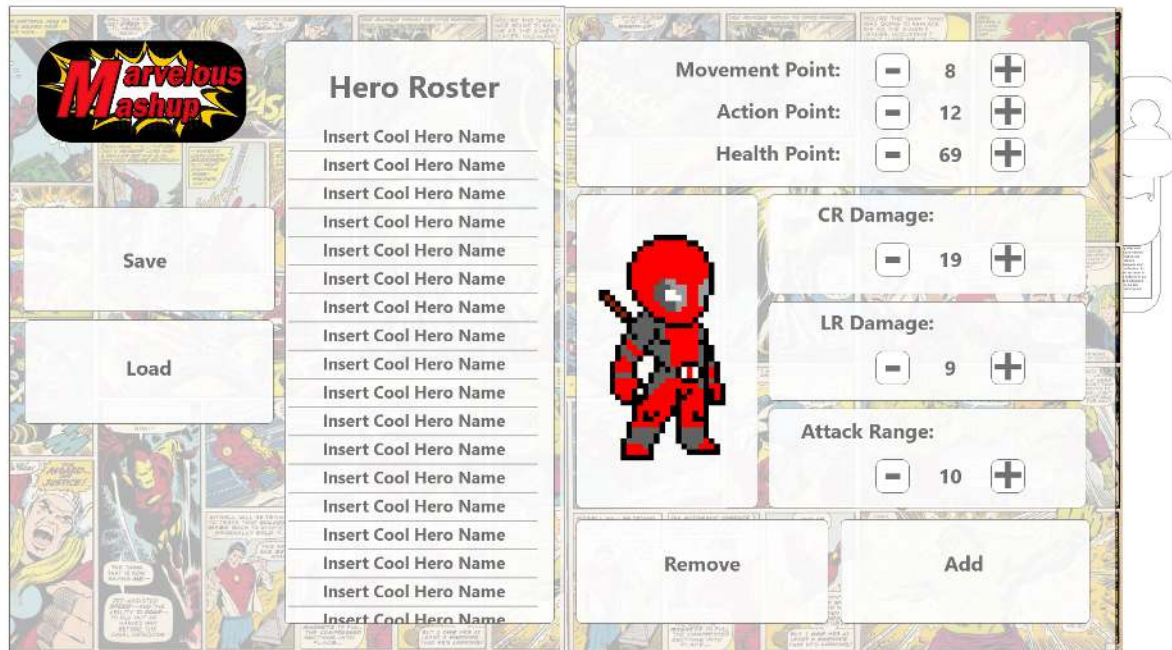
Game:



Im **Spiefenster** werden dem Spieler alle nötigen Informationen gezeigt, um das Spiel zu spielen. Der Bildschirm wird von der Spielfläche, der vom Szenario bestimmten Map, ausgefüllt, auf der die Charaktere sich durch simples Klicken bewegen können. Am oberen Rand wird dem Spieler präsentiert, wieviele HP der momentan aktive Charakter hat und welche Charaktere als nächstes ihren Zug haben und wieviele HP diese haben. Links an der "Zeitleiste" ist die Anzahl des momentan Zugs. Unten links wird dem Spieler das Inventar des momentan aktiven Charakters angezeigt, sowie 3 Balken: der rote Balken stellt die HP dar, der gelbe die AP und der blaue die MP dieses Charakters für diese Runde. Oben links ist noch ein Menüknopf.

Editor:

Character Editor:

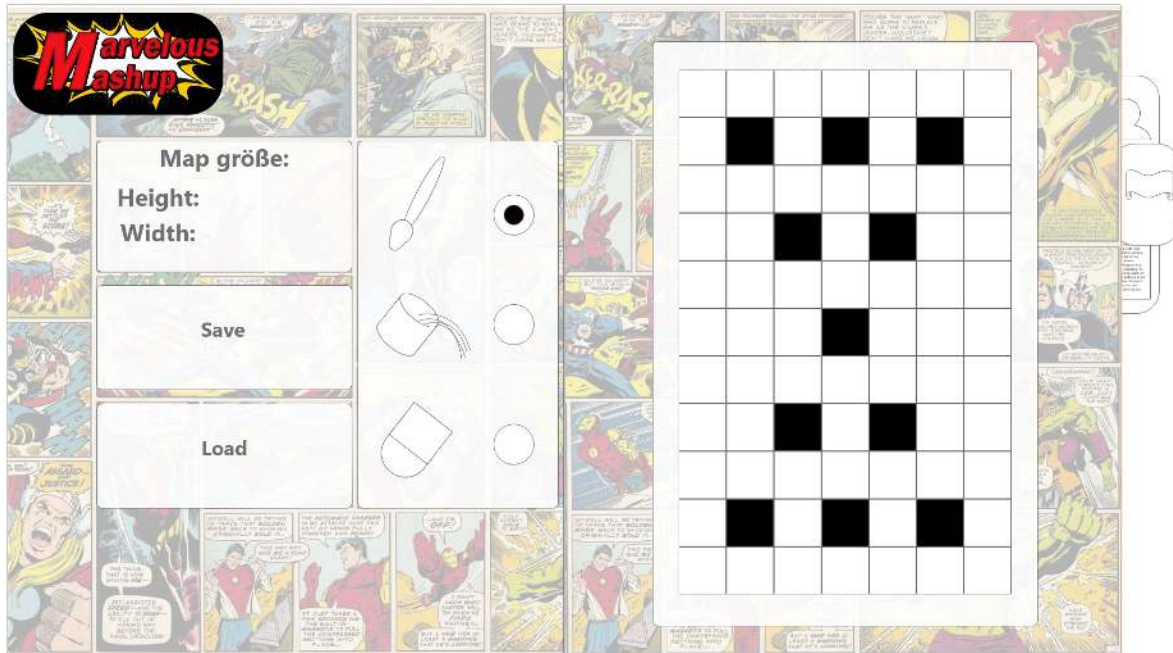


Im **Character Editor** kann der Nutzer eine ganze Reihe an Charakteren erstellen und editieren welche später spielbar sein werden. Der Nutzer kann auch einzelne Charaktere, welche zuvor hinterlegt wurden, in die Liste an, in diesem Szenario verfügbaren, Helden/Charakteren laden.

Das Design soll an ein Comic Buch erinnern. Die einzelnen Fenster der UI sind an die Rechtecke des Comics angelehnt. Die linke Seite listet alle möglichen Charaktere auf, die auf der rechten Seite nach eigenen Wünschen angepasst werden können.

Außerdem ist an der Seite mit einer Markierung die Seiten markiert die zu den anderen Bestandteilen des Editors führen. Diese können mit einem Klick ausgewählt werden.

Map Editor:

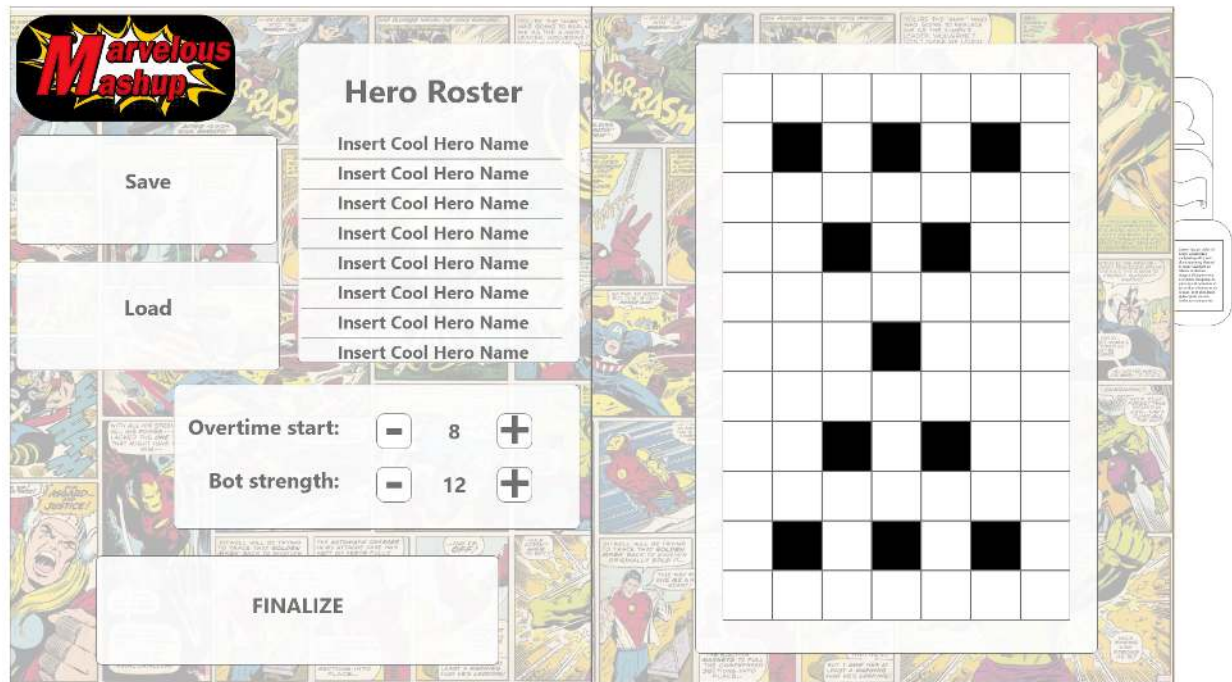


Im **Map Editor** kann der Nutzer Felsen auf das Spielfeld zeichnen und entfernen und somit eine eigene Map erstellen auf welcher dann im Szenario gespielt wird. Es wird dem Nutzer auch freigestellt vorherig gespeicherte Maps zu laden oder die neu von ihm erstellte zu hinterlegen.

Das Design soll an ein Comic Buch erinnern. Die einzelnen Fenster der UI sind an die Rechtecke des Comics angelehnt. Die rechte Seite dient hier als Vorschau für das Spielfeld und die linke Seite ist für die Interaktion gedacht.

Außerdem ist an der Seite mit einer Markierung die Seiten markiert, die zu den anderen Bestandteilen des Editors führen. Diese können mit einem Klick ausgewählt werden.

Szenario Editor:



Im **Szenario Editor** kann der Nutzer sich die Liste an Verfügbaren helden und die Map nochmal ansehen sowie die Overtime der runden und die Stärke der eventuellen KI einstellen. Dem Nutzer steht es auch frei Komplettes Szenario zu laden das zuvor Hinterlegt wurde und kann das nun von ihm erstellte auch hinterlegen.

Das Design soll an ein Comic Buch erinnern. Die einzelnen Fenster der UI sind an die Rechtecke des Comics angelehnt. Die Rechte Seite dient hier als Vorschau für das Spielfeld und die linke Seite lässt generelle Spielinformationen einstellen.

Außerdem ist an der Seite mit einer Markierung die Seiten markiert die zu den anderen Bestandteilen des Editors führen. Diese können mit einem Klick ausgewählt werden.

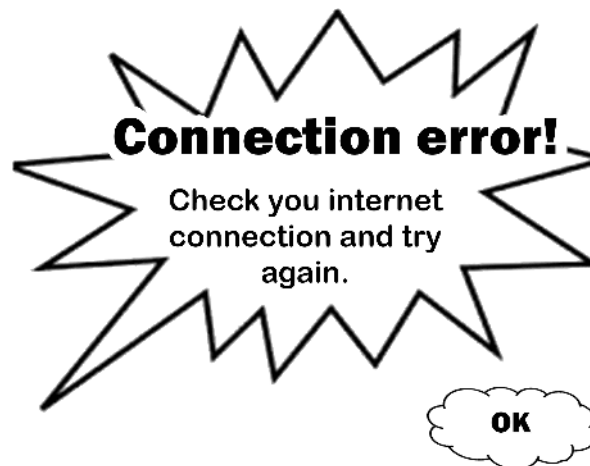
Dialoge:

Die **Dialoge** teilen sich in 2 verschiedene Arten von Dialogen auf.

Es gibt Alert Dialoge die den Nutzer auf etwas hinweisen. Diese Dialoge haben nur einen Knopf um den Dialog wieder zu schließen. Diese Dialoge sind meistens Error Dialoge. Das bedeutet, der Spieler wird auf einen Fehler aufmerksam gemacht.

Die andere Art sind Auswahldialoge. Bei diesen Dialogen hat der Nutzer mehrere Optionen aus denen er auswählen kann.

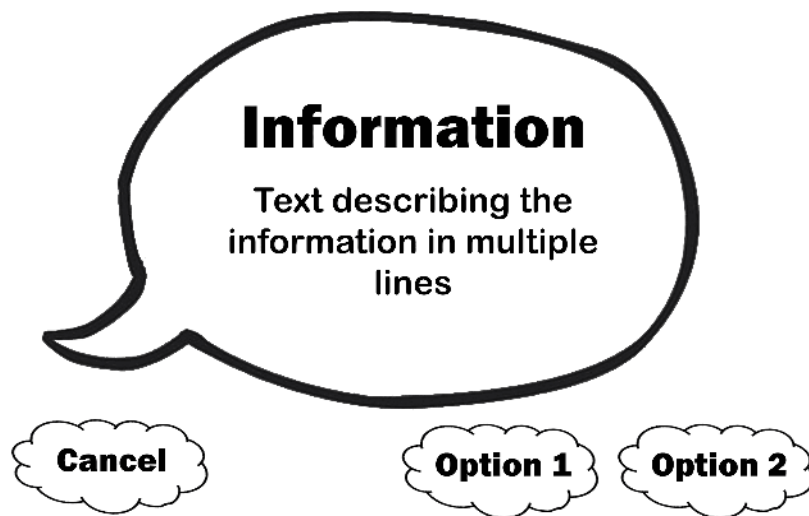
Alert Dialog:



Der **Alert Dialog** besteht aus einer Action Bubble mit einem Titel und einer Beschreibung. Titel und beschreibung befinden sich innerhalb der Action Bubble und nennen den Grund für den Dialog. Diese dürfen auch über die Bubble hinausgehen.

Der Button um den Dialog wieder zu schliessen befindet sich unterhalb der Action Bubble und besteht aus einer eigenen Thinking Cloud mit dem Text "OK"

Info Dialog:



Der **Info Dialog** lässt dem Nutzer eine Auswahl wie er fortfahren möchte.

Er besteht aus einer Speech Bubble die den Text beinhaltet. Der Text hat auch eine Überschrift und eine Beschreibung. Beide dürfen nicht über die Speech Bubble hinausgehen.

Unter der Bubble befinden sich weitere Thinking Clouds über die der Nutzer auswählen kann wie er fortfahren will. Standardmäßig gibt es eine zum abrechnen und 2 für weitere Optionen.

Rückmeldung:

Um dem Nutzer **Rückmeldung** zu geben, wenn auf etwas gewartet wird, gibt es unten rechts im Eck eine Anzeige dafür. Dadurch ist der Nutzer nicht verwirrt wenn eine Aktion ungewöhnlich lange dauert.

Es gibt hier 2 verschieden Arten von Wartezeiten. Wenn der Client auf eine Nachricht von einem Server wartet, und wenn eine Berechnung ungewöhnlich lange dauert.

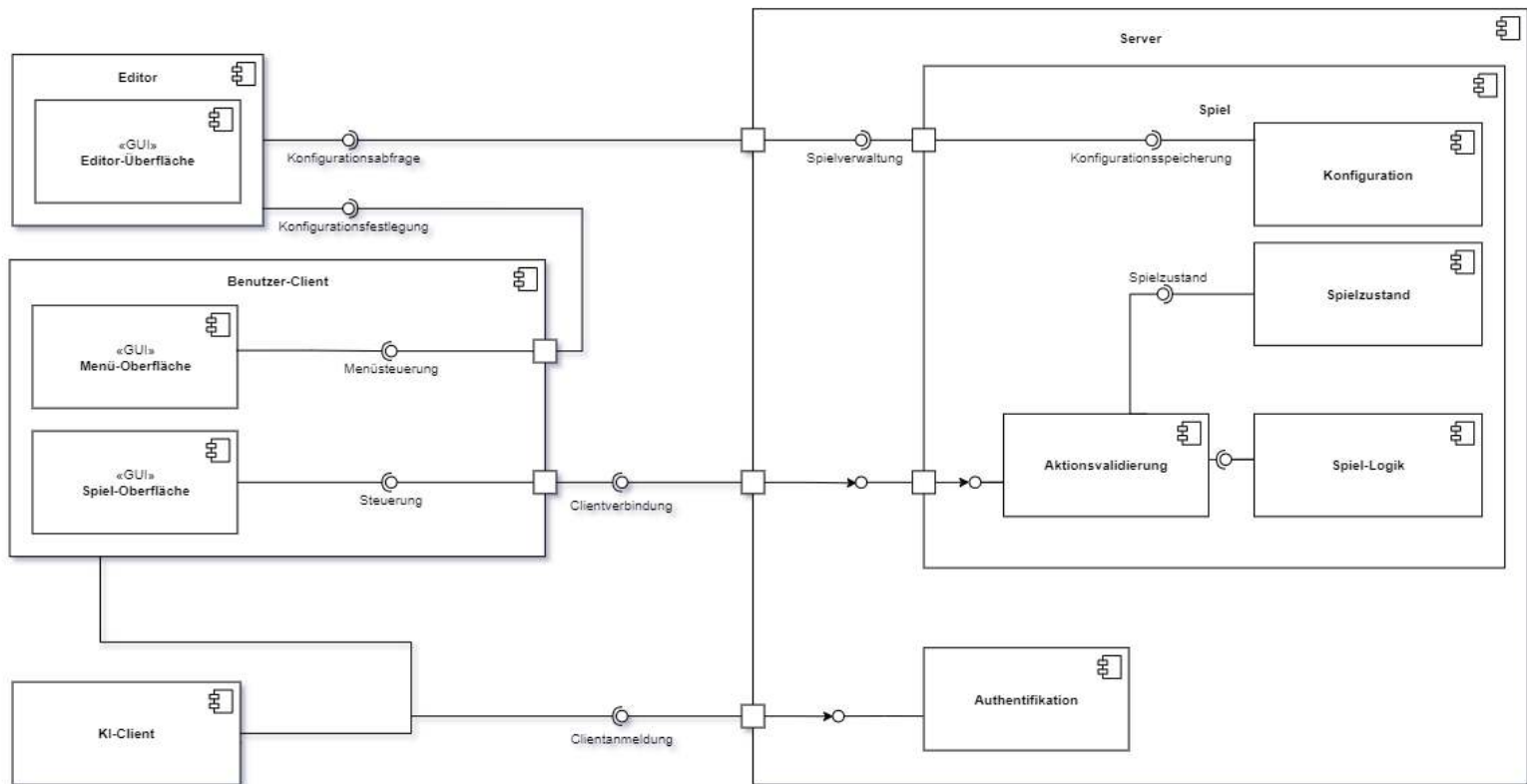


Server und KI Client:

Im Gegensatz zum Client bekommt der Nutzer selbst das Serverprogramm und den KI-Client nicht zu Gesicht. Um hier Ressourcen bei der Entwicklung zu sparen, wird der KI-Client keine graphische Oberfläche haben. Aus demselben Grund, und der Tatsache, dass Docker keine graphische Oberfläche in diesem Sinne unterstützt, wird auch der Server über die Kommandozeile gesteuert werden.

Meilenstein 5:

Komponentendiagramm:



Editor:

Die Komponente ist essentiell um individuelle Konfigurationen zu erstellen, die vom Spiel interpretiert werden. Diese können vom Server angefordert werden.

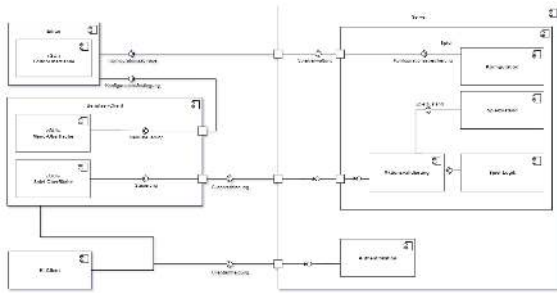
FAs: FA5

Editor-Oberfläche:

Die Oberfläche ist der Interaktionsbereich des Nutzers um den Editor zu steuern. Er hat eine eigene Komponente, da die Daten vom Nutzer eingegeben werden.

Die an der Oberfläche getätigten Eingaben werden vom Editor verarbeitet und bei einer Konfigurationsabfrage an den Server geschickt, wo die Dateien weiter interpretiert werden.

FAs: FA5



Benutzer-Client:

An dieser Komponente spielt sich die Eingabe des Spielers ab, heisst der Nutzer steuert was gemacht werden soll. Bei einer Front- & Backend Architektur handelt es sich hierbei um das Frontend.

FAs: FA1, FA3, FA11, FA34

Menü-Oberfläche:

Hier steuert der Nutzer über einzelne Menüpunkte. Er kann sich durch Menüs bewegen, den Editor starten und Menüpunkte auswählen.

FAs: FA25, FA35, FA36, FA37, FA38, FA40, FA44, FA45

Spiel-Oberfläche:

Der Spieler steuert hier verschiedene Aktionen des Spiels. Die Anweisungen werden verarbeitet und dann als Aktionen an den Server geschickt.

FAs: FA9, FA10, FA15, FA17, FA19, FA20, FA21, FA22, FA23, FA24, FA 26, FA27, FA28, FA29, FA32, FA33, FA39, FA43

KI-Client:

Client Anwendung die über einen Docker-Container läuft und Computergesteuert auch Anweisungen an den Server schickt.

FAs: FA4

Server:

Der Server hostet ein Spiel. Er authentifiziert dabei Clients. Außerdem kann er danach auch über Verbindungen Nachrichten entgegennehmen und an das Spiel weiterleiten. Hier werden die Verbindungen auch verwaltet.

FAs: FA1, FA2, FA8, FA16, FA18, FA30, FA31, FA32, FA41, FA42

Spiel:

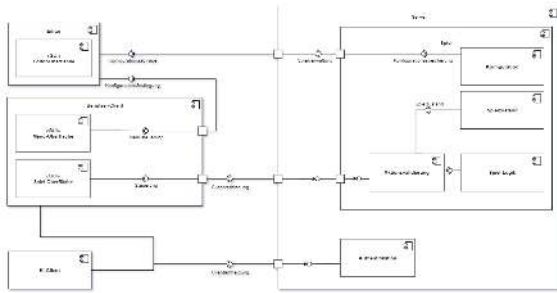
Das Spiel stellt die zentrale Spielsteuerung da. Hierbei werden alle Informationen verwaltet, die mit dem Spiel selber zu tun haben. Client Nachrichten und beliebige Konfigurationsdateien werden entgegengenommen und entsprechend verarbeitet.

FAs: FA14, FA15

Authentifikation:

Hier verwaltet der server anmeldungen der Spieler und KI clients. Angemeldete Spieler werden mit mit einer bestimmten ID versehen um sie eindeutig unterscheiden zu können.

FAs: FA37, FA41, FA42, FA45



Konfiguration:

Umfasst die 3 Jasons welch den Spielverlauf und die Helden definieren. Diese werden im Editor erstellt und bearbeitet. Sie werden vom Server geladen und können danach beliebig gelesen werden.

FAs: FA15

Spielzustand:

Der Spielzustand ist eine Speicherung des aktuellen Spiels und allen zugehörigen Werten. Er kann erst nach einer Authentifizierung der Nachricht verändert werden.

FAs: FA15

Aktionsvalidierung:

Jede einkommende Nachricht wird hier auf Korrektheit überprüft. Dadurch kann keine unmögliche, invalide Aktion ausgeführt werden. Wenn eine Nachricht als OK eingestuft wird, wird der Spielzustand dementsprechend verändert und eine Nachricht an alle Clients geschickt.

FAs: FA32

Spiel-Logik:

Die Logik umfasst alle im spiel vorhandenen vorgänge wie zum Beispiel den Helden bewegen oder Schaden zufügen. Jede von CLienten Aktion wird von der Spiellogik erfasst und ausgeführt.

FAs: FA32

Artefakte:

Charakter Konfiguration:

characterConfig.json

Partie Konfiguration:

partie-konfig.json

Szenario Konfiguration:

szenario-name.scenario, "*szenario-name*" muss ersetzt werden