

Projekt: Rupert  
  
Domainanalyse

Daunicht Nils (ndaunich@hsr.ch)

Fröhlich Adrian (a1froehl@hsr.ch)

Hochreutener Joel(jhochreu@hsr.ch)

Peisker Dominic(dpeisk@hsr.ch)

# Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 12.03.2015 | 0.1 | Erstellung | Dominic Peisker |
| 13.03.2015 | 1.0 | Überarbeiten/Korrektur, Zusammenfügen | Dominic Peisker, Nils Daunicht, Adrian Fröhlich |
| 17.04.2015 | 1.1 | Überarbeitungen / Aktualisierungen | Dominic Peisker |

# Inhalt

[Änderungsgeschichte 2](#_Toc417032040)

[Inhalt 3](#_Toc417032041)

[1. Einführung 4](#_Toc417032042)

[1.1 Zweck 4](#_Toc417032043)

[1.2 Gültigkeitsbereich 4](#_Toc417032044)

[1.3 Referenzen 4](#_Toc417032045)

[1.4 Übersicht 4](#_Toc417032046)

[2. Domain Modell 5](#_Toc417032047)

[2.1 Strukturdiagramm 5](#_Toc417032048)

[2.2 Konzeptbeschreibung 5](#_Toc417032049)

[3. Zustandsdiagramm 8](#_Toc417032050)

[4. Systemsequenzdiagramme 9](#_Toc417032051)

[4.1 UC 06: Runde durchführen 9](#_Toc417032052)

# Einführung

## Zweck

Dieses Dokument zeigt das Domänenmodell für das Projekt Rupert

## Gültigkeitsbereich

Das Dokument ist während des SE2 Projektes im Frühlingsemester 2015 und im Zusammenhang mit der Arbeit am Projekt Rupert gültig

## Referenzen

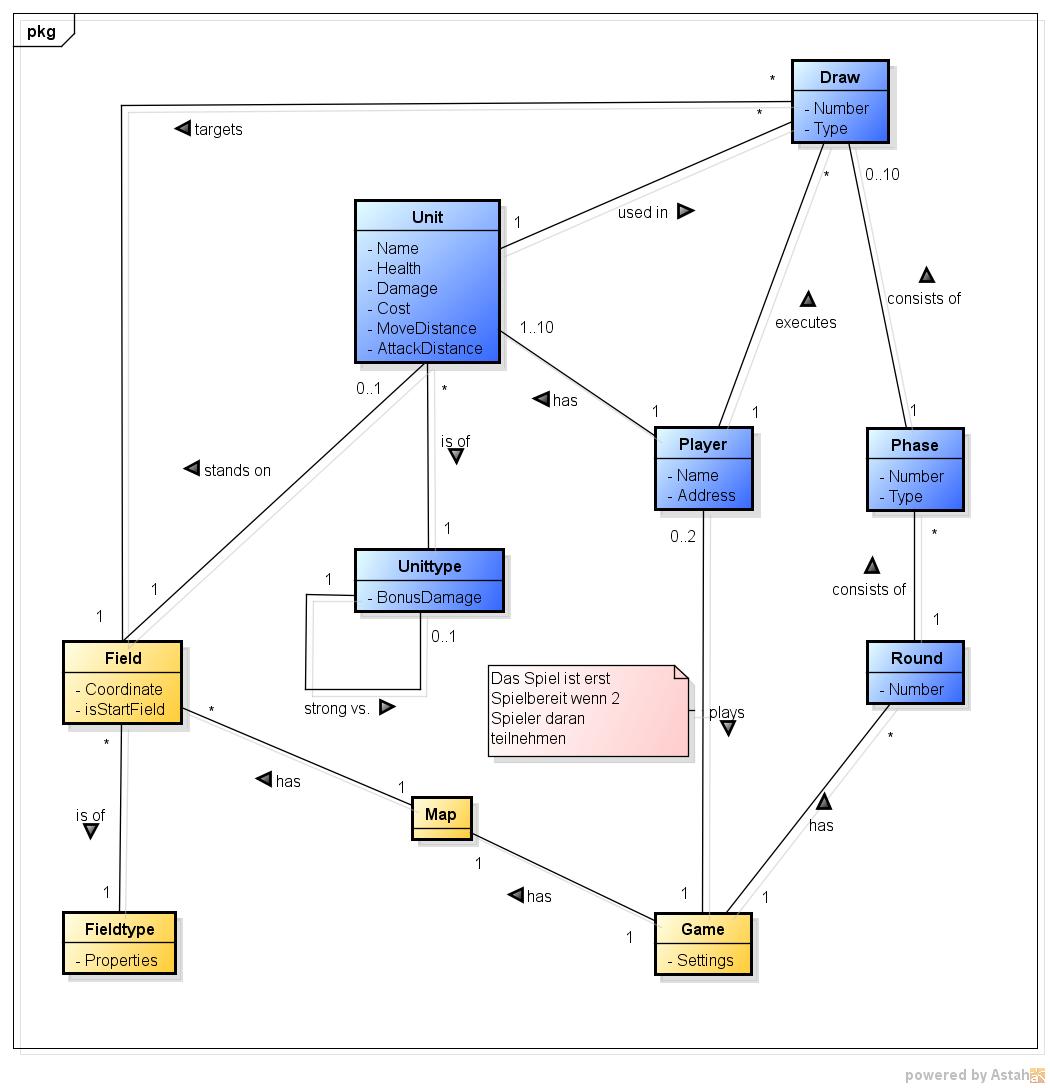
|  |  |
| --- | --- |
| Dokument | Inhalt |
| Projektplan | Beschreibt die Planung des Projekts |
| Anforderungsspezifikation | Use cases des Projekts |

## Übersicht

Zu Beginn des Dokuments wird das Domain Model eingeführt und die einzelnen Klassen der Domain detailliert beschrieben. Im Anschluss wird mittels eines Zustandsdiagrammes der Ablauf des Spiels verdeutlicht und zum Schluss die wichtigsten Use Cases mittels eines SSD dargestellt.

# Domain Modell

## Strukturdiagramm



## Konzeptbeschreibung

|  |  |  |
| --- | --- | --- |
| Game | | |
| Beschreibung | Stellt das Spiel dar und beinhaltet die Spielmechanik, Spiellogik und Spielregeln. | |
| Attribute | Settings | Eine noch nicht genau definierte Menge an Einstellungsmöglichkeiten für das Spiel. |
| Beziehung | Ein Spiel hat genau eine Karte. | |
| Ein Spiel hat 0 bis n Runden (Bis das Spiel zu Ende ist). | |
| Ein Spiel wird von zwei Spielern gespielt, wobei sich die beiden Spieler erst mit dem Spiel verbinden müssen. | |

|  |  |  |
| --- | --- | --- |
| Map | | |
| Beschreibung | Stellt die Karte (das Spielbrett) dar auf dem die Einheiten agieren. | |
| Attribute |  |  |
| Beziehung | Eine Karte pro Spiel. | |
| Eine Karte besteht aus mehreren Feldern. Es müssen genügend Felder vorhanden sein um zumindest jeder Einheit eine Startposition zu ermöglichen. | |

|  |  |  |
| --- | --- | --- |
| Field | | |
| Beschreibung | Stellt ein Feld auf der Karte dar. Auf einem Feld kann sich maximal eine Einheit befinden. | |
| Attribute | Coordinates | Die Koordinaten des Feldes bezüglich der Karte. |
| isStartField | Zeigt an ob es erlaubt ist auf diesem Feld eine Einheit während der Aufstellungsphase zu platzieren. |
| Beziehung | Ein Feld ist von einem Feldtyp. | |
| Auf einem Feld kann keine oder genau eine Einheit stehen. | |
| Ein Feld kann das Ziel von keinem oder mehren Zügen sein. | |

|  |  |  |
| --- | --- | --- |
| Fieldtype | | |
| Beschreibung | Stellt sozusagen das Terrain dar. | |
| Attribute | Properties | Eigenschafften eines Feldtyps. Dies kann besonderes Ausehen oder Bonusse resp. Malusse sein. |
| Beziehung | Ein Feldtype kann bei keinem oder mehreren Feldern vorhanden sein. | |

|  |  |  |
| --- | --- | --- |
| Player | | |
| Beschreibung | Stellt den Spieler dar. Auch der Host wird als Spieler bezeichnet. | |
| Attribute | Name | Der Alias des Spielers. |
| Address | Die Adresse an dem der Spieler zu erreichen ist (Einzigartiger Identifizierer). |
| Beziehung | Bis zu zwei Spieler sind an einem Spiel beteiligt (Zwei Spieler benötigt um das spiel zu starten). | |
| Ein Spieler hat eine oder bis zu zehn Einheiten zur Verfügung. | |
| Ein Spieler führt mehrere Züge pro Phase aus.. | |

|  |  |  |
| --- | --- | --- |
| Round | | |
| Beschreibung | Stellt eine abgeschlossene Runde im Spielt dar. Es werden solange Runde gespielt bis das Spiel zu Ende ist. | |
| Attribute | Number | Nummerierung der Runde. |
| Beziehung | Eine Runde besteht aus mehreren Phasen. | |
| Eine Runde gehört genau zu einem Spiel. | |

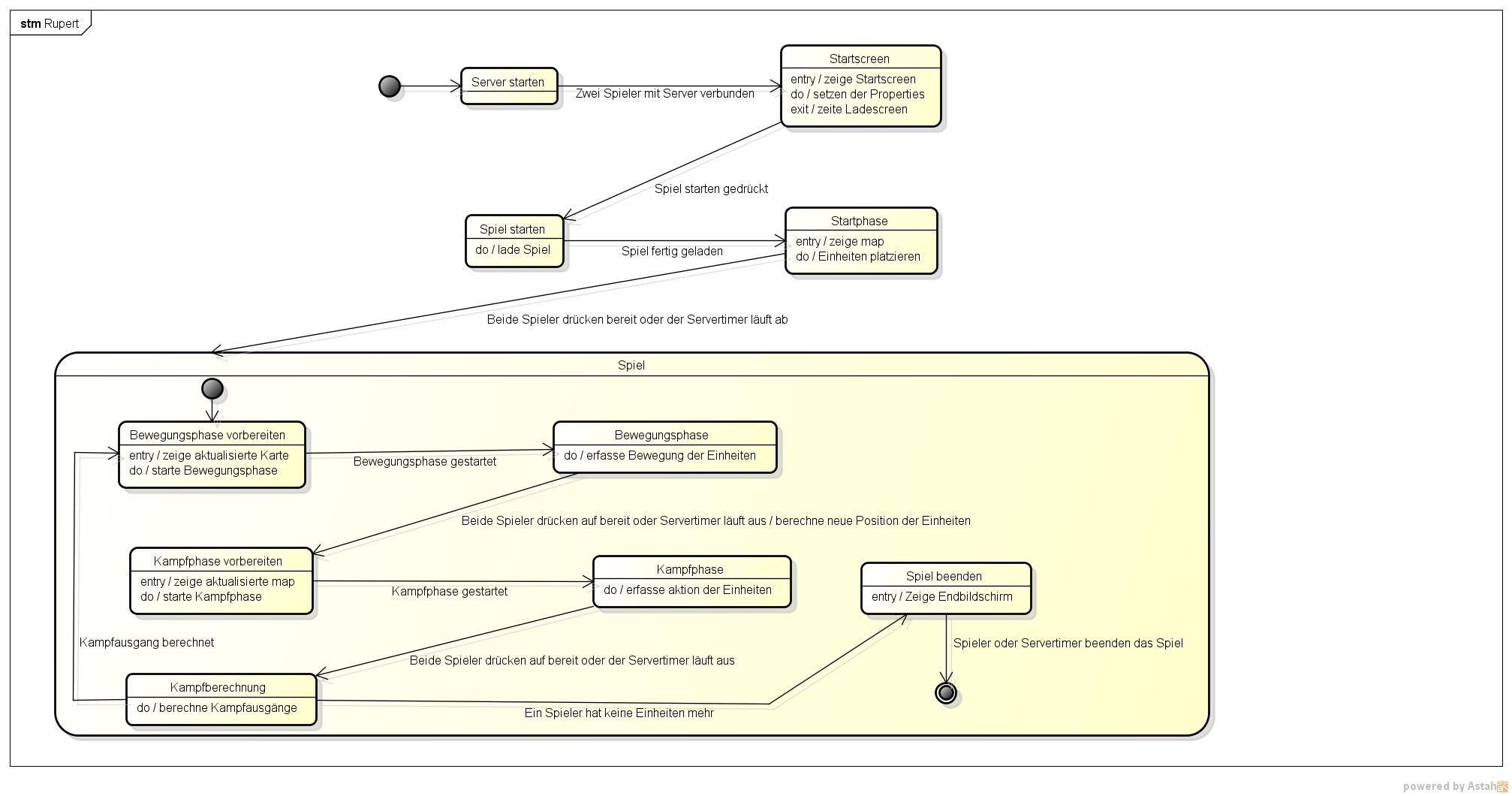
|  |  |  |
| --- | --- | --- |
| Phase | | |
| Beschreibung | Stellt ein Aktionsphase dar. Beinhaltet entweder Bewegungs- oder Kampfzüge. | |
| Attribute | Number | Nummer der Phase in der Runde. |
| Type | Der Typ der Runde. |
| Beziehung | Eine Phase gehört zu einer Runde. | |
| Eine Phase beinhaltet keinen bis zu zehn Züge. Es kann maximal ein Zug pro Einheit in einer Phase erfasst werden. | |

|  |  |  |
| --- | --- | --- |
| Draw | | |
| Beschreibung | Stellt die Aktion der Einheit in einer Phase dar. | |
| Attribute | Number | Die Nummer des Zuges in der Phase. |
| Type | Die Art der Aktion des Zuges. |
| Beziehung | Ein Zug ermöglich genau einer Einheit eine Aktion durchzuführen. | |
| Ein Zug gehört zu einer Phase. | |
| Ein Zug wird von einem Spieler durchgeführt. | |
| Ein Zug hat immer ein Feld als Ziel. | |

|  |  |  |
| --- | --- | --- |
| Unit | | |
| Beschreibung | Stellt eine Einheit im Spiel dar. | |
| Attribute | Name | Name bzw. Bezeichnung der Einheit. |
| Health | Die Anzahl Lebenspunkte der Einheit. Wenn die Lebenspunkte unter 0 fallen ist die Einheit tot. |
| Damge | Der Schaden den die Eineit dem Ziel zufügt. |
| Cost | Die Kosten der Einheit um sie im Spiel benützen zu können. |
| MoveDistance | Die Anzahl Felder, welche die Einheit während einer Bewegungsphase zurücklegen kann. |
| AttackDistance | Die Anzahl Felder, welche die Einheit zur Verfügung hat um Anzugreifen. |
| Beziehung | Eine Einheit hat genau einen Einheitentyp. | |
| Eine Einheit steht genau auf einem Feld. | |
| Eine Einheit wird von einem Spieler gesteuert. | |
| Eine Einheit kann keinmal oder genau einmal pro Zug benützt werden. | |

|  |  |  |
| --- | --- | --- |
| Unittype | | |
| Beschreibung | Stellt den Typ der Einheit. | |
| Attribute | BonusDamage | Der Schadensbonus gegen andere Einheitentypen. |
| Beziehung | Es kann keine oder mehrere Einheiten des selben Typs geben. | |
| Ein Einheitentyp kann gegen keinen oder genau einen anderen Einheitentyp stark sein. | |

# Zustandsdiagramm



# Systemsequenzdiagramme

Im Folgenden sind die Systemsequenzdiagramme der Fully Dressed Use Cases aufgeführt.

## UC 06: Runde durchführen

