

## *vires*: Pflichtenheft

Marc Huisinga

Vincent Lehmann

Steffen Wißmann

2. September 2015

# Inhaltsverzeichnis

<b>1</b>	<b>Spielerklärung</b>	<b>1</b>
<b>2</b>	<b>Zielbestimmung</b>	<b>2</b>
2.1	Musskriterien . . . . .	2
2.2	Wunschkriterien . . . . .	3
2.3	Abgrenzungskriterien . . . . .	3
<b>3</b>	<b>Produkteinsatz</b>	<b>3</b>
3.1	Anwendungsbereiche . . . . .	3
3.2	Zielgruppen . . . . .	3
3.3	Betriebsbedingungen . . . . .	4
<b>4</b>	<b>Produktumgebung</b>	<b>4</b>
4.1	Software . . . . .	4
4.2	Hardware . . . . .	4
4.3	Orgware . . . . .	4
<b>5</b>	<b>Produktfunktionen</b>	<b>5</b>
5.1	Benutzerfunktionen . . . . .	5
5.2	Administratorfunktionen . . . . .	7
<b>6</b>	<b>Produktdaten</b>	<b>7</b>
<b>7</b>	<b>Produktleistungen</b>	<b>8</b>
<b>8</b>	<b>Benutzeroberfläche</b>	<b>8</b>
<b>9</b>	<b>Qualitätszielbestimmung</b>	<b>9</b>
<b>10</b>	<b>Testszenarien</b>	<b>9</b>
<b>11</b>	<b>Entwicklungsumgebung</b>	<b>9</b>
<b>12</b>	<b>Glossar</b>	<b>10</b>
<b>13</b>	<b>Meilensteine</b>	<b>12</b>
<b>14</b>	<b>Layoutdesign</b>	<b>12</b>

### **Zusammenfassung**

Dieses Pflichtenheft beschreibt die Anforderungen an die Entwicklung des Spiels ***vires***, und bestimmt, welche Pflichten erfüllt werden müssen, damit die Entwicklung ***vires*** als erfolgreich abgeschlossen erklärt werden kann.

# 1 Spielerklärung

Bei “*vires*” handelt es sich insgesamt um ein einfaches RTS-Spiel. Das Spiel wird in einer Draufsicht gespielt.

Auf dem Spielfeld befinden sich mehrere Cells, welche im Spiel als Kreise dargestellt werden. Cells produzieren Vires, welche die eigentlichen Einheiten des Spiels darstellen. Dieser Produktionsvorgang wird als Replication bezeichnet.

Jede Cell hat eine gewisse Cell-Force, welche durch die Größe der Cell dargestellt wird. Innerhalb eines gewissen Zeitintervalls, welcher von der Cell-Force abhängt, erhöht sich die Menge an Stationed Vires, also die Menge an Vires in einer Cell, um ein Virus. Hat die Cell eine geringe Cell-Force, so ist auch die Replication dieser Cell langsamer.

Jede Cell hat eine Capacity, also einen gewissen Maximalumfang an Vires, den sie enthalten kann. Die Capacity wird ebenfalls von der Cell-Force bestimmt: Je geringer die Cell-Force, desto kleiner ist auch die Capacity.

Jeder Player, also jeder aktiv am Match teilnehmende Nutzer, hat am Anfang des Matches eine Start-Cell. Alle Start-Cells auf dem Feld haben die gleiche Cell-Force.

Alle Cells auf dem Feld, die einem Player gehören, werden auch als Manned Cells bezeichnet. Alle Cells auf dem Feld, die keinem Player gehören, sind Neutral Cells. Neutral Cells verhalten sich wie Manned Cells, haben jedoch eine langsamere Replication.

Vires können von Players, ausgehend von Manned Cells, die ihnen gehören, zu beliebigen anderen Cells, bewegt werden. Jedes Movement, also jede Bewegung von Vires, bewegt die Hälfte der Stationed Vires in der jeweiligen Manned Cell in Richtung der Target-Cell. Es können auch mehrere Movements gleichzeitig auf die gleiche Target-Cell gestartet werden, indem Players mehrere ihrer Manned Cells auswählen.

Die Vires innerhalb eines Movements heißen Moving Vires: Je größer die Anzahl an Moving Vires innerhalb eines Movements ist, desto größer wird die Movement-Force. Je kleiner die Movement-Force eines Movements ist, desto schneller ist auch das Movement. Movements werden ebenfalls durch Kreise dargestellt, deren Größe von der Movement-Force bestimmt wird.

Treffen zwei Movements des gleichen Players auf die gleiche Target-Cell aufeinander, so vereinen sich die Movements und bilden ein einzelnes Movement mit einer Movement-Force der beiden vereinten Movements. Treffen zwei Movements unterschiedlicher Player aufeinander, so tritt ein Conflict auf: Die Vires bekämpfen sich, indem die Movement-Forces der beiden Movements voneinander abgezogen werden. Nach einem Conflict während eines Movements bleibt nur das Movement über, das die stärkere Movement-Force hat, verkleinert aber seine Movement-Force durch das Aufeinandertreffen mit dem Feind.

Cells, die das Movement auf dem Weg zur Target-Cell überquert, werden einfach ignoriert und durchlaufen.

Trifft ein Movement auf seine Target-Cell und gehört diese dem selben Player, der das Movement losgeschickt hat, so werden die Moving Vires des Movements in die Stationed Vires der Target-Cell aufgenommen. Trifft ein Movement aber auf seine Target-Cell und gehört diese einem anderen Player, so tritt ebenfalls ein Conflict auf: Die Moving Vires werden von den Stationed Vires der Target-Cell abgezogen. Ist die Menge an Moving Vires größer als die Menge an

Stationed Vires, so übernimmt der angreifende Player die Cell. Überschreitet der Übergang von Moving Vires in Stationed Vires die Capacity, so gehen die überbleibenden Moving Vires verloren.

Insgesamt gewinnt ein Player das Match, wenn er/sie als letzter noch Cells auf dem Spielfeld besitzt oder wenn er/sie nach 15 Minuten in der Summe die größte Cell-Force besitzt. Haben nach 15 Minuten mehrere Players insgesamt eine gleich starke Cell-Force, so gewinnt der Player, welcher die meisten Vires besitzt. Sind alle diese Werte gleich, so entsteht ein unentschieden.

## 2 Zielbestimmung

Im Folgenden werden eine Menge spielspezifische Begriffe verwendet, welche im Glossar (unter [12](#)) nachgeschlagen werden können.

### 2.1 Musskriterien

#### Nutzer

- Nach dem Besuchen der Hauptseite der Website können Nutzer ihren Nutzernamen eingeben, dem Matchmaking beitreten oder direkt einem Match über die Room-ID beitreten.
- In der Pre-Match-Phase können Nutzer auswählen, ob sie Player oder Spectator sein wollen.
- Nutzer können sich innerhalb eines Rooms dazu entscheiden, zur Hauptseite zurückzukehren.
- Nutzer können einem Room über die URL mit der Room-ID des Raumes beitreten.

#### Player

- Während der Pre-Match-Phase können Players dafür abstimmen, dass das Match sofort gestartet wird.
- Während der Match-Phase können Players *vires* (unter [1](#) erklärt) spielen.
- Zu jeder Zeit können sich Players dazu entscheiden, zum Spectator zu werden.

#### Spectator

- Während der Match-Phase können Spectators das Match beobachten.
- Während der Pre-Match-Phase können sich Spectators dazu entscheiden, zum Player zu werden.

#### Administrator

- Administratoren können das client- und das serverseitige Fehler-Log einsehen.
- Administratoren können das Spiel während der Laufzeit konfigurieren.

## 2.2 Wunschkriterien

### Nutzer

- Nutzer können in einem Room mit anderen Nutzern chatten.
- Nutzer können nach dem Besuchen der Website eine Theme auswählen.

### Administrator

- Administratoren können das Spiel über eine Administrationskonsole verwalten.

## 2.3 Abgrenzungskriterien

- Das Spiel soll nur auf Englisch verfügbar sein.
- Spieler sollen nicht in der Lage sein, Spielelemente sichtbar für andere Spieler anzupassen.
- Spieler sollen im Chat nicht in der Lage sein, private Nachrichten zu schreiben.
- Das spielunterstützte Bilden von Teams soll nicht möglich sein.
- Das Spiel soll keine Daten über die Nutzung der Applikation sammeln.

# 3 Produkteinsatz

## 3.1 Anwendungsbereiche

*vires* ermöglicht es Spielern aus aller Welt mithilfe eines einfachen Online-Spiels gegeneinander anzutreten.

## 3.2 Zielgruppen

*vires* wird konzipiert, um Computerspielbegeisterten die Möglichkeit zu geben, sich ohne großen Aufwand und nur mithilfe ihres Browsers in einem RTS-Spiel mit anderen Spielern zu messen.

Da das gesamte Spiel extremst simpel ist, es kein Accountsystem und auch keine wirkliche Einstiegsbarriere gibt, ist das Spiel perfekt für Spieler geeignet, die schnell in kurzen Matches gegen andere Spieler antreten wollen, ohne sich groß an das Spiel zu binden.

Das Spiel ist sehr einfach zu erlernen, ist also für viele verschiedene Gruppen geeignet. Nutzer müssen lediglich dazu in der Lage sein, eine einfache Benutzeroberfläche zu bedienen. Da das Spiel nur auf Englisch verfügbar ist, müssen Nutzer wenigstens einfaches Englisch verstehen können, um sich auf der Benutzeroberfläche zurechtzufinden.

### 3.3 Betriebsbedingungen

Die Software soll sich nicht von anderen Webdiensten unterscheiden. Dies bedeutet, dass der Dienst insbesondere ständig erreichbar ist und die Server entsprechend ausgelegt sind um eine größere Last tragen zu können, damit das System nicht zusammenbricht.

## 4 Produktumgebung

### 4.1 Software

#### Client

- Aktueller WebGL- und Websockets-fähiger Webbrowser (Internet Explorer 11, Firefox 4, o.ä.)
- JavaScript

#### Server

- Applikations-Binary (Webserver und Bibliotheken werden statisch gelinkt)

### 4.2 Hardware

#### Client

- Internetfähiger PC, die Internetverbindung muss dabei permanent bestehen um die Software nutzen zu können

#### Server

- Internetfähiger Server
- Ausreichend Rechenleistung und Arbeitsspeicher, da der Server viele Clients auf einmal verarbeiten können muss
- Da keine permanenten Daten gespeichert werden, beschränkt sich der benötigte Speicherplatz auf die Größe der verwendeten Server-Software und die der vom Server erstellten Logs
- Eine Architektur, welche von Go 1.5 unterstützt wird

### 4.3 Orgware

- Gewährleistung der Server-Hardware-Anforderungen
- Konfiguration der Applikation

## 5 Produktfunktionen

### 5.1 Benutzerfunktionen

/F0010/ *Eingabe des Benutzernamens:* Jeder Nutzer hat einen Nutzernamen, was es einfacher machen soll, Cells und Vires bestimmten Namen zuzuordnen. Der Nutzernamen kann nach dem Betreten der Hauptseite eingegeben werden.

- Nutzernamen müssen nicht eindeutig sein: Mehrere Nutzer können den gleichen Namen verwenden.
- Jeder Nutzernamen ist auf 20 Zeichen limitiert.
- Wird der Nutzernamen nicht angegeben, so gilt dieser automatisch als der Nutzernamen "Unknown".
- Nutzernamen werden über Cookies bei den Nutzern abgespeichert.
- Besuchen Nutzer die Hauptseite erneut, so befinden sich ihre Nutzernamen bereits im Eingabefeld.

/F0020/ *Matchmaking:* Auf der Hauptseite können Nutzer dem Matchmaking beitreten. Das Matchmaking sorgt dafür, dass Nutzer in einen guten Room weitergeleitet werden.

Ein guter Room ist wie folgt definiert:

- Der Room befindet sich aktuell in der Pre-Match-Phase.
- Die Maximalanzahl an Players in dem Room ist noch nicht erreicht.
- Der Room hat die geringste Anzahl an Players aus allen Rooms, die in Frage kommen.

Lässt sich kein guter Room finden, so wird ein neuer Room erstellt, in den die Nutzer dann weitergeleitet werden.

/F0030/ *Direktes Beitreten:* Mithilfe der Room-ID können Nutzer auf der Hauptseite einem bestimmten Room direkt beitreten. Ist die Maximalanzahl an Players des Rooms erreicht, so treten Nutzer dem Room automatisch als Spectators bei.

/F0040/ *Phases:* Jeder Room befindet sich entweder in der Pre-Match-Phase oder in der Match-Phase. Wird die eine Phase beendet, so wird direkt die nächste Phase gestartet.

/F0050/ *Pre-Match-Phase:* Während der Pre-Match-Phase wird auf Players für die Match-Phase gewartet.

/F0060/ *Countdown:* Sind zwei oder mehr Players im Room, so fängt ein Timer an, von 60 Sekunden aus herunterzuzählen. Erreicht der Timer null Sekunden, so beginnt das Match. Während dieser Wartezeit können die Players abstimmen, ob sie das Match sofort starten wollen. Die Abstimmung dauert 10 Sekunden und kann nur mit einer Zweidrittelmehrheit der an der Abstimmung teilnehmenden Players entschieden werden.



/F0070/ *Map-Generation:* Am Ende der Pre-Match-Phase wird eine zufällige Spielkarte generiert.

Die generierte Spielkarte muss folgende Ansprüche erfüllen:

- Die Größe des Fields passt sich der Menge an Players an.
- Cells werden gleichmäßig auf dem Field verteilt.
- Jeder Player bekommt eine Start-Cell mit der gleichen Cell-Force, bei allen anderen Cells handelt es sich um Neutral-Cells.
- Die Start-Cells werden gleichmäßig auf die Cells aufgeteilt, sodass zwischen den Start-Cells möglichst große Distanzen liegen.
- Jede Cell erhält eine zufällige Cell-Force, welche die Größe der Cell vorgibt.
- Die maximale Größe der Cell-Force ist limitiert.
- Kleinere Cell-Forces sind wahrscheinlicher als größere Cell-Forces.

/F0080/ *Match-Phase:* Während der Match-Phase wird von den Players *vires* wie es unter 1 erklärt ist, gespielt, während Spectators das Spiel beobachten.

/F0090/ *Rollen:* Tritt man einem Room bei, so ist man standardmäßig Player. Players können sich in einem Room dazu entscheiden, zu Spectators zu werden.

- Lief während der Entscheidung der Countdown, existiert aber nach der Entscheidung nur noch ein Player, so wird der Countdown abgebrochen.
- War der Player an einer Abstimmung beteiligt, so wird seine Stimme zurückgezogen.
- War der Player aktuell in der Match-Phase aktiv, so wird eine Neutralization auf alle seine Manned Cells durchgeführt und seine Moving Vires entfernt.

Während der Pre-Match-Phase können Spectators zurück zu Players wechseln, insofern die Maximalanzahl an Players des Rooms nicht erreicht ist. Das Zurückwechseln ist während der Match-Phase nicht möglich.

/F0100/ *Zurückkehren zur Hauptseite:* Innerhalb eines Rooms können Nutzer zu jedem Zeitpunkt zur Hauptseite zurückkehren, um dort ihren Namen zu ändern oder einem anderen Room beizutreten.

/F0110/ *Maximalanzahl an Players:* Jeder Room hat eine Maximalanzahl von 20 Players. Rooms können insgesamt auch mehr Nutzer enthalten, insofern die restlichen Nutzer alle Spectators sind.

/F0120/ *Room-ID:* Jeder Room hat eine einzigartige Room-ID, welche sich auch in der URL des Rooms wiederfinden lässt. Die Room-ID kann verwendet werden, um einem Room direkt beizutreten.

/F0130/ *Schnittstellenversion:* Ändert sich die Client- oder die Serverversion des Kommunikationsprotokolls so werden Nutzer über dies informiert. Der Client selbst wird so entkoppelt, dass außer dem Kommunikationsprotokoll keine Konflikte zwischen Versionen auftreten können. Treten serverseitig gravierende Veränderungen auf, welche nicht mit der Konfigurationsdatei umgesetzt werden können, so wird versucht, die Änderungen mithilfe von Hot-Patching einzufügen. Ist dies nicht möglich, so wird das System neu gestartet.

/F0140/ *Timeout:* Sind Nutzer länger als 15 Minuten abwesend, so werden ihre Verbindungen geschlossen, sie zurück auf die Hauptseite geleitet und darauf hingewiesen, dass ihre Verbindung aufgrund der Abwesenheit geschlossen wurde.

## 5.2 Administratorfunktionen

/F1010/ *Backend-Logging:* Tritt ein Fehler im Backend der Applikation auf, so wird die Fehlermeldung in einem Log abgespeichert und versucht, den Ablauf fortzusetzen. Jede Server-Session erhält ihre eigenen Backend-Logfiles.

/F1020/ *Frontend-Logging:* Tritt ein Fehler im Frontend der Applikation auf, so wird versucht, die Fehlermeldung an den Server zu schicken und den Ablauf fortzusetzen. Jede Server-Session erhält ihre eigenen Frontend-Logfiles.

/F1040/ *Konfiguration:* Der Server kann während der Laufzeit über eine Konfigurationsdatei konfiguriert werden.

Die folgenden Dinge können konfiguriert werden:

- Maximalanzahl an Zeichen in einem Nutzernamen
- Maximalanzahl an Players in einem Room
- Dauer des Countdown-Timers
- Konstanten der Map-Generation
- Timeout-Dauer

Sollten sich während der Entwicklung weitere Faktoren finden, die sich zur Konfiguration eignen, so werden diese ebenfalls konfigurierbar sein.

## 6 Produktdaten

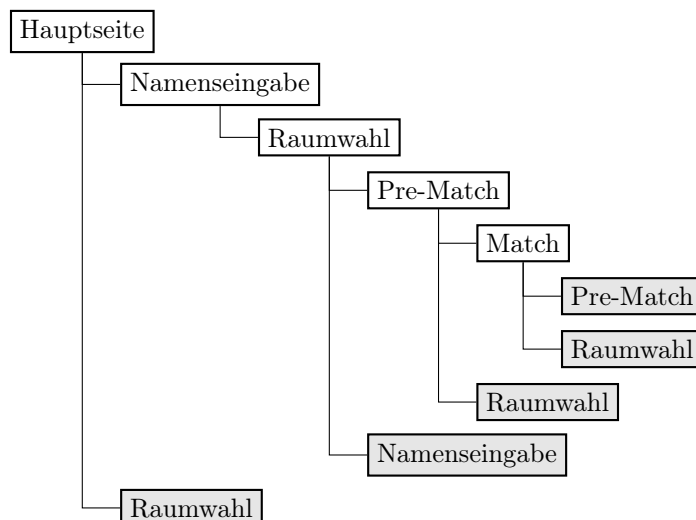
Insgesamt speichert die Applikation serverseitig keinerlei Daten über ihre Nutzer, was die Eintrittsbarriere weiter verringert, die Nutzerfreundlichkeit steigert und die benötigte Speichermenge senkt. Um die Nutzerfreundlichkeit jedoch weiter zu erhöhen, werden einige Daten clientseitig gespeichert:

/D0010/ *Nutzernamen:* Die Applikation speichert den Nutzernamen mithilfe von Cookies, damit der Nutzer diesen nicht jedes mal erneut eingeben muss, wenn er das Spiel erneut betritt.

## 7 Produktleistungen

- /L0010/ *Fehlerhafte Eingaben:* Tätigen Nutzer eine fehlerhafte Eingabe, so werden sie über die Fehler in ihrer Eingabe informiert und können ihre fehlerhafte Eingabe modifizieren.
- /L0020/ *Konfiguration:* Die Basiskonfiguration des Systems muss anpassbar sein, ohne dass die Applikation neu gestartet werden muss.
- /L0030/ *Frontend- & Backend-Abgrenzung:* Front- und Backend müssen komplett voneinander abgegrenzt sein. Die einzige Verbindung, welche Frontend und Backend teilen dürfen, ist das Kommunikationsprotokoll zwischen Client und Server.
- /L0040/ *Versionierung:* Ändert sich die Applikation, so dürfen aufgrund der Frontend- und Backend-Abgrenzung keine Versionskonflikte auftreten, während die Applikation genutzt wird. Ändert sich das Kommunikationsprotokoll zwischen Client und Server, so werden Nutzer über dies informiert und um einen Neustart gebeten.
- /L0050/ *Sicherheit:* Das Spiel muss komplett sicher sein. Players dürfen nicht mithilfe einer Modifikation ihres Clients dazu in der Lage sein, signifikante Vorteile gegenüber anderen Players zu erreichen.
- /L0060/ *Performanz:* Das Spiel selbst muss auf einem durchschnittlichen PC flüssig spielbar sein.
- /L0070/ *Fehlertoleranz:* Treten Fehler innerhalb der Applikation auf, so wird versucht, abhängig von der Situation des Fehlers, den aktuellen Ablauf insofern fortzusetzen, dass der Fehler keinen Einfluss auf den Ablauf der Applikation hat. Tritt beispielsweise ein Fehler innerhalb eines Verbindungsaufbaus auf, so wird die Verbindung zwar beendet, der Server stürzt aber nicht ab.

## 8 Benutzeroberfläche



## 9 Qualitätszielbestimmung

	sehr wichtig	wichtig	weniger wichtig	unwichtig
Robustheit	x			
Zuverlässigkeit	x			
Korrektheit	x			
Benutzerfreundlichkeit	x			
Effizienz	x			
Portierbarkeit			x	
Kompatibilität			x	

## 10 Testszenarien

Alle nicht trivialen Algorithmen sollen automatisiert getestet werden, während alle trivialen Programmabläufe manuell getestet werden sollen.

Die folgenden Funktionen werden als nicht trivial erachtet und sollen automatisiert getestet werden:

/T0010/ *Matchmaking*: Der Vorgang des Matchmakings soll mithilfe von automatisierten Tests getestet werden, um zu überprüfen, ob der Matchmaking-Algorithmus alle Bedingungen eines “guten Rooms” erfüllt.

/T0020/ *Map-Generation*: Der Vorgang der Map-Generation soll mithilfe von automatisierten Tests getestet werden, um zu überprüfen, ob die generierte Spielkarte alle Ansprüche erfüllt.

Alle anderen Funktionen werden manuell getestet.

Sollten sich bei der Entwicklung weitere Funktionen als nicht trivial und gut testbar erweisen, so werden diese ebenfalls automatisiert getestet werden.

## 11 Entwicklungsumgebung

### Betriebssysteme

- Windows 7/8.1
- Linux

### Tools

- Sublime Text 2
- GoSublime
- Better Coffeescript
- Node Package Manager
- go-Tool
- goimports
- coffee-Tool
- git

- [Github](#)
- [L<sup>A</sup>T<sub>E</sub>X](#)

### Programmiersprachen

- [Go 1.5](#) (Backend)
- [CoffeeScript 1.9.3](#) (Frontend)

### Bibliotheken

- [Gorilla Web Toolkit](#)
- [three.js](#)
- [RequireJS](#)
- [toml](#)
- [flog](#)
- [posarg](#)

## 12 Glossar

**Accountsystem** bezeichnet ein System, in welchem Nutzer mithilfe eines Nutzernamens und eines Passworts eine permanente Identität erhalten.

**Applikations-Binary** stellt eine Datei dar, welche die Applikation ausführen kann.

**Backend** stellt den serverseitigen Teil der Applikation dar.

**Capacity** bezeichnet die Maximalanzahl an Stationed Vires, welche eine Cell enthalten kann.

**Cell-Force** bezeichnet die Größe einer Cell.

**Cell** bezeichnet einen Kreis auf dem Field, welcher Stationed Vires hält und produziert.

**Client** stellt einen Nutzer dar, der mit dem Server verbunden ist.

**Conflict** stellt ein Aufeinandertreffen von feindlichen Einheiten mit freundlichen Einheiten dar.

**Cookie** bezeichnet ein kleines Speicherobjekt, welches auf dem Client-PC hinterlegt werden kann.

**Frontend** stellt den clientseitigen Teil der Applikation dar.

**Hauptseite** bezeichnet die Index-Seite der Website, welche erscheint, wenn der Nutzer nur die URL der Website angibt.

**Manned Cell** bezeichnet eine Cell, welche einem Player gehört.

**Map-Generation** stellt den Vorgang der zufälligen Generierung einer Spielkarte dar.

**Match-Phase** bezeichnet die Phase, in welcher aktuell ein Match stattfindet.

**Match-Start** stellt den Start der Match-Phase und somit den Spielbeginn dar.

**Match** bezeichnet eine Spielinstanz.

**Matchmaking** stellt den Prozess der Findung eines Rooms dar, in welchem aktuell die Pre-Match-Phase aktiv ist.

**Movement-Force** stellt die Menge an Moving Vires innerhalb eines Movements dar.

**Movement** stellt die Bewegung von Stationed Vires von einer Manned Cell über Moving Vires zu einer anderen Cell dar.

**Moving Vires** stellt Vires dar, welche sich von einer Cell zu einer anderen Cell bewegen.

**Neutral Cell** bezeichnet eine Cell, welche keinem Player gehört.

**Neutralization** bezeichnet den Vorgang des Umwandeln einer Manned Cell in eine Neutral Cell.

**Phase** bezeichnet einen abgegrenzten Zustand eines Rooms.

**Player** stellt einen Nutzer dar, welcher aktiv an einem Match teilnimmt.

**Pre-Match-Phase** bezeichnet die Phase, in welcher momentan kein Match stattfindet.

**Replication** stellt die Produktion von Vires in einer Cell dar.

**Room-ID** bezeichnet einen einzigartigen und generierten Namen eines Rooms.

**Room** stellt einen Raum innerhalb des Spiels dar, in welchem sequentiell mehrere Matches ausgetragen werden können.

**RTS** bedeutet "Real Time Strategy" und bezeichnet ein Spielgenre, in welchem Strategie im Fokus liegt und sich jede Entscheidung direkt auf das Spielgeschehen auswirkt.

**Spectator** stellt einen Nutzer dar, welcher das Match beobachtet und nicht aktiv am Match teilnimmt.

**Start-Cell** stellt die erste Cell eines Players dar, die er nach dem Match-Start besitzt.

**Stationed Vires** stellt Vires dar, welche sich in einer Cell befinden.

**Target-Cell** bezeichnet eine Cell, welche das Ziel eines Movements ist.

**Theme** bezeichnet ein Texturenpaket, welche die Oberfläche nach einer bestimmten Thematik gestaltet.

**Vires** bezeichnet mehrere Virus-Einheiten.

**Virus** bezeichnet eine Einheit auf dem Field, welche zwischen Cells bewegt werden kann.

**13    Meilensteine**

**14    Layoutdesign**