

Beschreibung des Interfaces IKampfsystem

Das Kampfsystem stellt die Methode „angreifen“, die von den Klassen „Spieler“ und „Monster“ verwendet wird. Innerhalb der jeweiligen Klasse wird die Methode dann definiert.

Implementationsbeispiel beim Monster:

Zuerst wird eine Zahl zwischen 0 und 1 zufällig generiert. Wenn die Zahl zwischen 0 und 0.9 liegt wird das jeweilige Lebewesen leicht verletzt. Bei einer Zahl größer als 0.9 und kleiner als 0.98 wird das Lebewesen schwer verletzt und bei allen anderen Möglichkeiten wird das Lebewesen direkt getötet.

Auf die Methode „angreifen“ des jeweiligen Lebewesens wird in der Klasse „Spiel“ zugegriffen. Mit dem Befehl „attack“ wird ein Kampf zwischen dem Spieler und dem Monster in seinem Raum simuliert.

Bei der Simulation wird erst das Monster angegriffen und als Reaktion darauf greift das Monster selbst den Spieler an. Dabei werden die Statistiken der beiden Lebewesen abwechselnd ausgegeben. Man hat jederzeit die Möglichkeit den Raum zu wechseln um sich in Sicherheit zu begeben.

Beschreibung des Zustandes „Tod“

Der Zustand „Tod“ ist Teil des Interfaces „IZustand“, in dem die jeweiligen Zustände der Lebewesen festgelegt werden. Bei dem Zustand „Tod“ handelt es sich um den letztmöglichen Zustand. Er resultiert aus einem Kampf zwischen zwei Lebewesen und beendet das Spiel für das getötete Lebewesen¹.

Den Zustand „Tod“ kann ein Lebewesen erhalten in dem es:

- Gesund ist und tödlich verletzt wird.
- Verwundet ist und schwer verletzt wird.
- Bewegungsunfähig ist und leicht verletzt wird.

Beschreibung des Gegenstands Landkarte (Map)

Die Landkarte ist ein Gegenstand, um das Spielfeld graphisch mit ASCII Zeichen anzeigen lassen kann.

Mit dem Befehl „Take“ kann das Item aufgesammelt werden.

Mit dem Befehl „use“ kann es benutzt werden. Nach dem Benutzen wird die Karte wie folgt angezeigt.

```
> use Map
+-+--+
|  @|
+  ++ +
+  ++ +
|  || |
+-+--+
```

Hierbei sieht man, dass es 4 Räume gibt und einer davon ist der Teleporter-Raum. Gekennzeichnet mit einem ‚@‘ in der Mitte des Raumes.

Ein Raum besitzt in den Ecken ein ‚+‘ und jeder nicht vorhandene Ausgang wird horizontal mit einem ‚|‘ und mit einem ‚-‘ in vertikaler Richtung gekennzeichnet.

Hierbei wird die Klasse LevelGenerator verwendet, um die Landkarte beim Start des Spiels zu generieren. Der Generator fügt Räume in die Landkarte und verbindet sie. Ebenso fügt er Monster und Gegenstände aus den Pools hinzu. Diese werden zufällig auf die Räume verteilt. Der Gegenstand ‚Landkarte‘ wird immer gesetzt, weil es ein essentieller Gegenstand ist.

Der Generator geht wie folgt vor:

1. Zufällige Richtung ermitteln
2. Schauen ob der nächste Raum kein Raum außerhalb des Arrays ist
3. Wenn ja, dann überprüfen ob der nächste Raum belegt ist
 - a. Wenn nicht, eine Tür hinzufügen
 - b. Wenn ja, neuen Raum als nächsten Raum hinzufügen
4. falls der nächste Punkt außerhalb des Arrays ist, sollte überprüft werden, ob ein anderer Nachbarraum noch unbelegt ist.
5. Die Abbruchbedingung sind
 - a. wenn die Anzahl der Durchgänge erreicht ist,
 - b. ein Raum keine leeren Ausgänge mehr hat. Damit Abwechslung bei der Anzahl der Räume entsteht

Die Klasse Landkarte spielt hierbei nur als Speicherort des Levels eine Rolle und kann einzelne Räume oder sogar einen zufälligen Raum, sowie die Anzahl der bestehenden Räume ausgeben. Die Klasse Landkarte wird dem Gegenstand Landkarte übergeben. Somit kann man über den Gegenstand Landkarte das Level ausgeben.

Veränderung an der Klasse Lebewesen

Bisher musste man einen Gegenstand erst auf den Boden werfen, um ihn benutzen zu können. Jetzt gibt es die Möglichkeit einen Gegenstand direkt einzusetzen.

Mit der Methode „getAlleGegenstaende()“ kann man auf die einzelnen Gegenstände in der Klasse Lebewesen zugreifen, um sie zu benutzen, verändern oder sogar zu löschen.