

RANDOOM

Plattform/Shooter-Computerspiel mit „random content generation“

Spiel: Andreas Burkard

Sound Design & Musik: Julia N. Winzer

Betreuer: Andreas Stiegler

Einleitung

Bei RanDoom handelt es sich um ein Computerspiel, das bei jedem neuen Starten des Spieles durch die zufälligen generierten und zusammengestellten Inhalte ein neues Spielerlebnis bietet.

Gameplay

Alle Modelle und die Umgebung des Spieles sind in 3D, aber spielerisch gesehen ähnelt es den alten 2D-Plattformern, da das Gameplay nur in zwei Dimensionen stattfindet. Man kann deshalb bei diesem Spiel von 2,5D sprechen (siehe Abbildung 1).

Abbildung 1: Gameplay



Zufällige Level-Generierung

Jeder Level-Abschnitt besteht aus kleinen Räumen oder Raumabschnitten, die zufällig, aber dennoch sinnvoll aneinander angefügt werden.

In so einem Raum oder Raumabschnitt befinden sich Positionen, auf denen zufällig ermittelt wird, welcher Gegner, welche Hindernisse oder welche Power-ups platziert werden.

Unter der Haube funktioniert die Level-Generierung durch sogenannte „hierarchical task networks“.

In dem Fall dieses Spieles, besteht ein Level aus einem String, z.B. „A U“.

Wie in einer Grammatik wird dieser geparkt und U (steht für „Underground“) wird dann anhand von passenden, aber zufällig ausgesuchten Regeln in andere Bausteine umgewandelt.

Als Beispiel „U“ -> „E N N R X“

- E steht für einen zufälligen Raum, der als Level-Übergang zur letzten Region dient
- N steht für einen zufälligen normalen Raum
- R steht für einen zufälligen Raum, in dem ein Charakter gefangen gehalten wird
- X steht für einen zufälligen Raum, der einen Ausgang aus dieser Region in die nächste enthält.

Diese Grammatikbausteine können dann weiter in kleinere Bestandteile geparkt werden, bis das gewünschte Level entsteht (siehe Abbildung 2 und 3).

Abbildung 2: Zufälliges Untergrund-Level

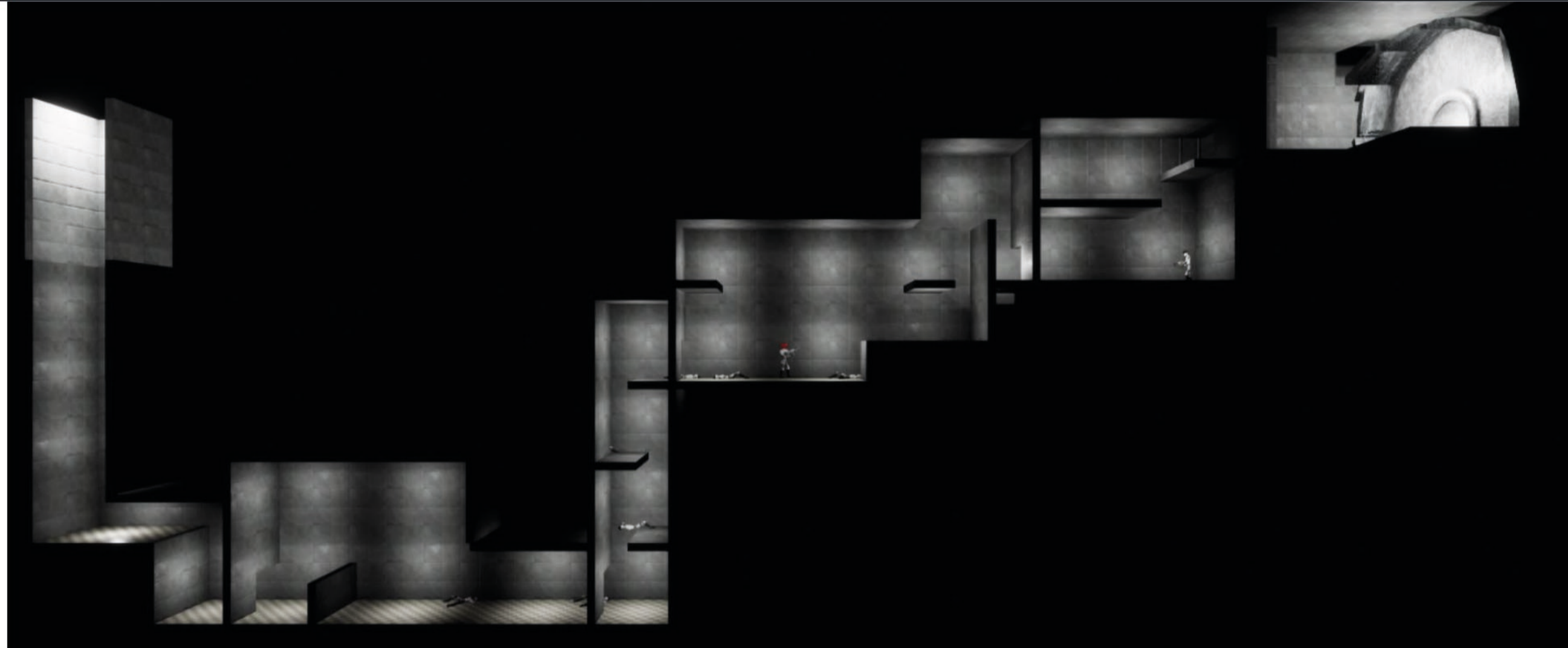


Abbildung 3: Zufälliges Oberflächen-Level



Zufällige Erstellung von Waffen

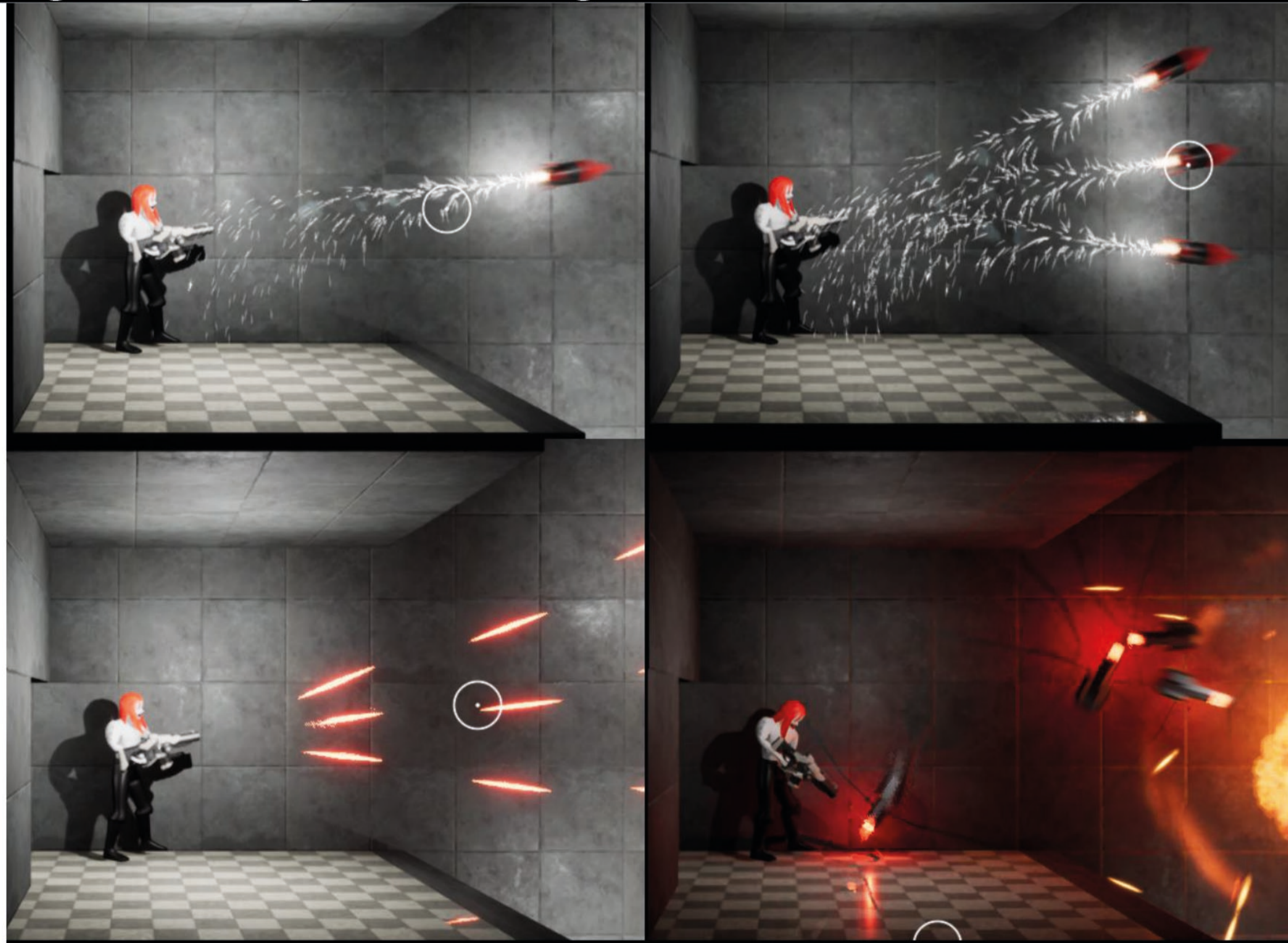
Eine Waffe besteht aus 3 verschiedenen Komponenten.

- Die **Art der Schüsse**: beispielsweise Raketen oder Laser.
- Der **Feuermodus**: einzelne Schüsse, Salven oder Schnellfeuer
- Den **Lauf** der Waffe: ob wenige starke oder streuende schwache Schüsse, ob präzise oder unpräzise Schüsse

Realisiert wird die Erstellung durch einen zufälligen **Index** pro Komponente innerhalb von Wahrscheinlichkeitstabellen (siehe Abbildung 4).

1 3 3 könnte z.B. eine Laserwaffe mit Schnellfeuer sein, die immer drei schwache Schüsse auf einmal abgibt.

Abbildung 4: Zufällig zusammengestellte Waffen



Zufällig erstellte Spielercharaktere

Ähnlich wie bei den Waffen bestehen auch die Charaktere, die der Spieler übernimmt, aus einer Aneinanderreihung von zufälligen **Indices**.

Der **Hauptindex** bestimmt, welcher Charakter-Klasse der Spieler angehört.

Dieser **Hauptindex** bestimmt auch, wie das **Basismodell** des Spielers aussieht und welche **Fähigkeit** er besitzt.

Durch einige der anderen **Indices** wird dann festgelegt, welche Frisur, welche Haarfarbe, welche Kopfbedeckung, welche Kleidung, welche Accessoires und welche Proportionen der Charakter hat (siehe Abbildung 5).

Das Anfügen von Haaren, Accessoires etc. an das Modell funktioniert über sogenannte **Sockets**.

Das sind Positions-, Rotations- und Skalierungsdaten, die sich mit dem unsichtbaren Skelett des Modells mitbewegen können. An diese **Sockets** kann man Modelle anhängen, die sich entsprechend des jeweiligen Sockets transformieren.

Abbildung 5: Zufällig zusammengestellte Charaktere



Verwendete Software

- Unreal 4
- Blender
- FMOD
- Cubase 9