**2.2 Funktionale Anforderungen**

**2.2.1 Spieler**

**2.2.1.1 Movement** - Der Spieler muss sich wie folgt bewegen können:

1. Der Spieler muss sich nach rechts bewegen können [Taste „D“]

2. Der Spieler muss sich nach links bewegen können [Taste „A“]

3. Der Spieler muss springen können [Taste „W“]

4. Der Spieler muss doppelt springen können, wenn: [2x Taste „W“]

i) er sich in der Luft befindet

ii) er die Fähigkeit freigeschalten hat

iii) er keine Aktion währenddessen ausführt

iv) er genug Stamina besitzt

5. Der Spieler muss in die Bewegungsrichtung dashen können, wenn: [Taste „L-Shift“]

i) er genug Stamina besitzt

ii) er die Fähigkeit freigeschalten hat

iii) er nicht angreift

iv) der Benutzer eine bestimmte Bewegungsrichtung angibt

**2.2.1.2 Kampf** - Der Spieler muss folgende kampfbezogene Aktionen ausführen können:

1. Der Spieler muss eine leichte Attacke in Blickrichtung durchführen können, wenn:

[Taste „Linke Maustaste“]

i) er genug Stamina besitzt

2. Der Spieler muss eine schwere Attacke in Blickrichtung durchführen können, wenn:

[Taste „Rechte Maustaste“]

i) er genug Stamina besitzt

3. Der Spieler muss ein Angriff blocken können, wenn: [Taste „Leertaste“]

i) er genug Stamina besitzt

4. Der Spieler muss sich mit einem Blood Vial heilen können [Taste „E“]

**2.2.1.3 Stamina** – Der Spieler hat eine Ausdauer 0….MAX\_STAMINA, diese regeneriert sich nach 3 Sekunden, wenn keine Stamina kostende Aktion durchgeführt wurde. Folgende Aktionen reduzieren die Stamina:

1. Leichte Attacke [-10 Stamina]

2. Schwere Attacke [-25 Stamina]

3. Doppelsprung [-15 Stamina]

4. Dash [-20 Stamina]

5. Blocken stoppt Stamina Regeneration. Beim erfolgreichen Blocken [-30% vom Angriff]

**2.2.1.4 Health** – Der Spieler hat eine Trefferpunktanzahl (TP) 0….MAX\_HEALTH.

1. Der Spieler stirbt wenn Health <= 0

2. Der Spieler heilt sich wenn er ein Blood Vial trinkt [20 TP]

3. Beim Checkpoint wird der Spieler komplett geheilt [MAX\_HEALTH]

4. Beim erleiden des Schadens verliert der Spieler den Schadenswert an TP

5. Health regeneriert sich NICHT über Zeit

**2.2.1.5 Leveling & Sünden –** Der Spieler kann durch Gegner und Umgebung Sünden erlangen, mit diesen kann er folgende Attribute bei einem Checkpoint verbessern:

1. MAX\_HEALTH + 25TP [100 Sünden]

2. STAMINA\_HEALTH + 25 Stamina [100 Sünden]

3. Waffenangriff +5 DMG [100 Sünden]

4. Blood Vials: Heilkraft +5 TP [100 Sünden]

Glasscherben können verwendet werden, um ein extra Blood Vial zu erlangen

**2.2.1.6 Tod** – Der Spieler hinterlässt alle Sünden am Todesort beim Sterben. Falls der Spieler ein zweites Mal stirbt, ohne die Sünden eingesammelt zu haben, werden diese entfernt.

**2.2.1.7 Kollision** – Der Spieler muss mit folgenden Objekten kollidieren können:

1. Boden und Wände

2. Gegner

**2.2.2 HUD**

**2.2.2.1 Healthbar –** Im Display muss die Healthbar oben links angezeigt werden. Diese muss die momentane TP des Spielers abbilden.

**2.2.2.2 Staminabar –** Im Display muss die Staminabar oben links unter der Healthbar angezeigt werden. Diese muss die momentane Stamina des Spielers abbilden.

**2.2.2.3 Bloodvials –** Im Display muss die Anzahl der Blood Vials unter der Staminabar angezeigt werden. Diese muss die momentane Anzahl der Blood Vials abbilden.

**2.2.2.4 Sünden –** Im Display muss die Anzahl der Sünden oben rechts angezeigt werden. Diese muss die momentane Anzahl der Sünden abbilden.

**2.2.3 Gegner**

**2.2.3.1 Movement**:

1. Der Gegner muss den Spieler anfangen zu verfolgen, wenn dieser sich in seinem Detectionbereich befindet.

2. Der Gegner muss aufhören den Spieler zu verfolgen, wenn dieser den Pursuingbereich verlässt.

3. Der Gegner muss sich im seinem Raum patrouillieren können.

4. Der Gegner hat eine Startposition [x,y – koordinate]

5. Der Gegner kehrt nach Zeit zur Startposition zurück

**2.2.3.2 Angriff** – Der Gegner entscheidet per Zufall und Stamina welche Attacke er ausführt:

1. Der Gegner muss eine leichte Attacke durchführen können

2. Der Gegner muss eine schwere Attacke durchführen können

**2.2.3.3 Health** – Der Gegner hat eine Trefferpunktanzahl (TP) 0….MAX\_HEALTH.

1. Der Gegner stirbt, wenn Health == 0.

2. Die TP des Gegners muss über den Gegnersprite sichtbar sein

**2.2.3.4 Tod –** Wenn der Gegner stirbt, hinterlässt dieser Sünden auf dem Boden

[Anzahl Sünden: SIN\_AMOUNT]

**2.2.3.5 Kollision** – Der Gegner muss mit folgenden Objekten kollidieren können:

1. Boden und Wände

2. Spieler

**2.2.4 Environment**

**2.2.4.1 Brunnen –** Diese dienen als Checkpoint für den Spieler. Wenn dieser mit dem Brunen interagiert passiert folgendes:

1.Alle Blood Vials werden aufgefüllt

2. Der Spieler heilt sich zu 100%

3. Alle toten Gegner die Respawnable sind, respawnen an der Startposition

**2.2.4.2 Kiste –** Der Spieler kann damit interagieren. Kann Glasscherben oder Sünden enthalten.

**2.2.4.3 Level Design –** Das Spiel besteht aus vier Sektionen, mit verschiedenen Räumen, die logisch verbunden sind. Diese Räume bestehen aus Boden & Wänden. In diesen sind Gegner verteilt. Pro Sektion gibt es jeweils einen Boss. (Wird verfeinert)

**2.2.5 Spielmanagment**

**2.2.5.1 –** Der Benutzer muss das Spiel im Menu, durch ein Knopfdruck, starten und verlassen können

**2.2.5.2 Spielstand:**

1**.** Der Benutzer muss in der Lage sein, den Spielstand zu speichern [JSON-Datei]

2. Der Benutzer muss in der Lage sein, den Spielstand zu löschen

3. Der Benutzer muss in der Lage sein, den Spielstand laden zu können.

**2.2.5.3 Optionen –** Diese Optionen dienen für die Konfiguration des Spiels.

1. Der Benutzer muss die Steuerung beliebig ändern können [Tastebelegung]

2. Der Benutzer muss die Grafik ändern können [Auflösung]

3. Der Benutzer muss die Sounds ändern können, zwischen 0 – 100% [Lautstärke]

**2.3 Nicht Funktionale Anforderungen**

**2.3.1 Coding Style:**

1. PascalCase für Klassennamen: Klassennamen wie „Player“ verwenden den PascalCase-Stil.

2. Konstante Variablen werden groß geschrieben. Bsp.: SPEED, JUMP\_VELOCITY

**2.3.2 Wartbarkeit:**

1. Das Spiel muss so entwickelt werden, damit es in der Zukunft nachvollziehbar und weiterentwickelbar ist.

2. Der Code muss mit Doxygen dokumentiert werden.

**2.3.3 Leistung:**

1. Das Spiel muss flüssig laufen mit mindestens 30FPS

2. Das Spiel muss Absturzfrei laufen (0 Abstürze pro Stunde)