

InstantGames

QM Handbuch



Inhaltsverzeichnis

1. FIRMENPROFIL.....	4
1.1 Firmenbeschreibung.....	4
1.2 Was sind Casual Games?	5
2. QM IN SPIELEN.....	6
2.1 Vorwort.....	6
2.2 Allgemeine Testverfahren und Methodiken	6
2.3 Alphatest	7
2.4 Betatest	8
3. QUALITÄTSMERKMALE.....	9
3.1 Detailed Design Document (DDD).....	9
Definition	9
Form	10
Inhalt.....	11
3.2 Spielspaß.....	12
Inhalt:.....	12
Atmosphäre:.....	12
Benutzerfreundlichkeit:.....	13
Technische Aspekte:.....	13
Wie wird getestet?	14
3.3 Korrektheit.....	16
Definition:.....	16
Vorabmaßnahmen:.....	16
Test-Fälle:	16
Modul- und Unit-Test:	16
Kompletttest:.....	16
Hardwarekompatibilität:	16
Mathematische Verifikation:.....	17
4. FRAGEBOGEN ZUM SPIELSPAß.....	18

1. Firmenprofil

1.1 Firmenbeschreibung

Die Firma *InstantGames* entwickelt im klassischen Sinne (Auftragsarbeit) aber auch in Eigenentwicklung, mit entsprechender Vermarktung, PC- sowie Konsolenspiele. Dabei liegt der Entwicklungsschwerpunkt bei den CasualGames welche im nächsten Abschnitt noch einmal näher definiert werden.

Zur Entwicklung der Spiele setzt *InstantGames* auf eine plattformunabhängige Render/Gameengine welche im Laufe der letzten Jahre entwickelt wurde und stetig weiterentwickelt wird. Diese Eigenentwicklung sichert zum einem einen hohen Qualitätsstandard, als auch ein hohes Maß an aktuellem technologischen Umfang, der so zur Entwicklung der Spiele zum Einsatz kommen kann.

In der Entwicklung arbeiten drei Teams parallel um immer zwei Spiele neben läufig entwickeln zu können. Das dritte Team kümmert sich speziell um die Entwicklung der Engine und ist beispielsweise im Bereich der Grafikprogrammierung mit den entsprechenden APIs und Portierungsspezifikationen zwischen verschiedenen Konsolen versiert und verfügt über ein überdurchschnittliches Wissen sowie Erfahrung im Softwareengineeringbereich. Die ersten beiden Teams bestehen aus Game- und Content programmieren, Level-Designer, 3D- und 2D Arts, Gamedesignern und kümmern sich im Gesamten betrachtet um den konzeptionellen Teil aber auch um die eigentliche Entwicklung der Spiele. Den jeweiligen Abteilung stehen Leads/Projektleiter über welche, jeweils zusammen, das Projekt koordinieren.

Obwohl sich die Entwicklung von Spielen meist deutlich von normalen Softwareprojekten unterscheidet, sind wir trotzdem und gerade deswegen bemüht einen höchstmöglichen Qualitätsstandard zu halten.

Die weiteren Abteilungen wie Marketing und Vertrieb finden in diesem Handbuch keine Erwähnung, da dort eigenständige und vollständig verschiedene Qualitätsmerkmale zur Anwendung kommen und umgesetzt werden müssen.

1.2 Was sind Casual Games?

Die Firma *InstantGames* hat sich auf die Entwicklung von CasualGames spezialisiert. Diese Spiele zeichnen sich durch eine einfache Handhabung mit einer schnellen Lernkurve aus, ohne jedoch eine gewisse Spieltiefe zu vernachlässigen. Beispiele dafür sind Mahjong, Moorhuhn aber auch viele Spiele die für die wii-Konsole entwickelt wurden. Diese Spiele sind für jede Altersklasse gedacht und beziehen Coregamer (Vielspieler) aber auch den Bereich des Family Entertainment mit ein. Das heißt im Klartext das diese Spiele sich von der üblichen Zielgruppe (männlich, 16-38) deutlich abgrenzen und somit auch zur Unterhaltung von Kinder aber auch Frauen gedacht sind.

Gespielt werden diese Spiele vermehrt zur Entspannung, Ablenkung und Stressbewältigung und nicht selten wegen ihres pädagogischen Mehrwerts.

Bedingt durch den starken Marktzuwachs im Konsolenbereich in den letzten Jahren, muss auch *InstantGames* diesen Markt bedienen um wirtschaftlich rentabel zu bleiben. Gerade auch in diesem Segment ist die Entwicklung und Einführung vieler CasualGames deutlich zu erkennen und bietet ein hohes Potenzial. Die erfolgreiche Einführung dieser Spiele auf den entsprechenden Konsolen ist nicht zuletzt auf die neuen und innovativen Eingabegeräte, wie etwa das Nunchuck, zurückzuführen. Diese Eingabegeräte erlauben ein intuitives Steuern der Spiele und ermöglichen es so, eine große Zielgruppe abzudecken.

2. QM in Spielen

2.1 Vorwort

Um schon vor Beginn eines Projektes, einen hohen Qualitätsstandart zu wahren muss das Personal regelmäßig an Schulungen teilnehmen.

Um Unklarheiten bei der Spezifikation zu vermeiden, findet ein permanenter Dialog mit dem Kunden statt, mithilfe dieser Rücksprachen können potentielle Fehlerquellen umgangen werden.

Außerdem findet die Entwicklung unserer Spiele im Prototyping statt, damit gewährleistet ist, dass es zu jedem Zeitpunkt eine Lauffähige Version des Spieles gibt.

2.2 Allgemeine Testverfahren und Methodiken

Ein großer Teil der Qualitätssicherung in der Spieleentwicklung lässt sich auf die gängigen Qualitätssicherungsmaßnahmen zurückführen, welche auch in der restlichen Softwareentwicklung angewandt werden. Konkret heißt dies, verschiedene Tests, zu verschiedenen Entwicklungsstadien. Dabei werden einige Test rein firmenintern durchgeführt und einige mit Hilfe von externen Mitarbeitern bzw. auch mit der eigentlichen Community/Zielgruppe.

Diese Tests werden ausgiebig geplant und nur von qualifizierten Testern durchgeführt. Diese Tester verfügen über langjährige Erfahrung und sind auf diese Art von Spielen, sowohl für den PC als auch auf der Konsole, spezialisiert. Weiterhin sind sie mit dem Ablauf der Produktion vertraut und haben entsprechendes Basiswissen über die Arbeitsabläufe von Programmieren, Grafikern etc. Das ermöglicht eine enge und professionelle Kommunikation bei etwaigen Fehlern.

Die Tests werden mit Hilfe der Management- und Bugtrackingsoftware Jira, normalen Excel-Tabellen aber auch ein Wiki, welches im Intranet verfügbar ist, durchgeführt.

In allen folgenden Testverfahren wird bei einem Problem eine entsprechende Priorität festgelegt, der Tester vermerkt, der das Problem entdeckt und ein zuständiger Entwickler der sich diesem Problem annimmt und dieses behebt. Die Prioritätsklassen reichen dabei von niedrig (z.B. Lokalisierungsfehler) bis hoch (z.B. ein wiederholter Fehler aus dem Netzwerk zwischen Vista und XP Systemen).

Wichtig ist weiterhin das das Problem und der Hergang durch den Tester möglichst genau beschrieben wird, damit der Entwickler den Problemfall auch replizieren kann. Wurde das Problem dann behoben, ist es eminent wichtig, dass eine erneute Prüfung durch den entsprechenden Tester stattfindet. Erst wenn diese erfolgreich war gilt das Problem als behoben.

Weiterhin sei die Qualitätssicherung der angesprochenen Engine hier ausgenommen, da hierbei, von einem Spiel vollkommen abstrakt durchgeführte Methoden, zur Anwendung kommen und sich diese Tests ausschließlich auf die technische Funktionalität beschränken. Um den Rahmen dieses Dokuments nicht zu sprengen wurde dafür ein separates Dokument angefertigt.

2.3 Alphatest

Der Alphatest wird ausschließlich firmenintern durchgeführt und wird in einem sehr frühem Stadium der Entwicklung durchgeführt. Das kann durchaus bedeuten das einige Elemente des Spiels noch nicht fertiggestellt sind. Bei diesem Test geht es folglich auch mehr um das funktionierende Zusammenspiel verschiedener Softwaremodule und die Lauffähigkeit auf allen angestrebten Konsolen um auf Basis dieses Buildes weiterarbeiten zu können. Wenn es zu größeren Problemen während der Validierung kommt erfolgen so viele Iterierungen des Alphatests, bis diese Probleme gelöst sind.

Jedoch wird aber auch in der Alphaversion des Spieles dieses auf seine eigentliche Funktionalität und die Spielmechanik hin überprüft. Ziel dieser Test ist es nun zu schauen ob das Spiel, in der aktuellen Version, unter den gegebenen Voraussetzungen, komplett funktioniert und spielbar ist. Das heißt konkret, dass zum Beispiel geprüft werden muss, ob ein Level auch tatsächlich beendet werden kann. Dabei ist es weiterhin wichtig, dass dies auch ohne Frustration des Spielers passiert und dabei die Anzahl der Versuche im Rahmen des Normalen bleiben muss. Eine zu schwer gestellte Aufgabe erfüllt nicht den Sinn und Zweck von einem Spiel bzw. Unterhaltungssoftware.

Bei einem Alphatest werden aber beispielsweise auch Fehler in der Lokalisierung gesucht und behoben. Dabei muss unter Anderem geschaut werden, das die GUI(Menü und im Spiel selbst) mit entsprechend unterschiedlich großen Satzzeichen/Satzlängen ordnungsgemäß arbeitet und es dort zu keinen unschönen Fehlern kommt(zum Beispiel das englischer Text über die Textbox hinaus läuft).

Die Alphatests werden aus den oben genannten Gründen Abteilungsübergreifend durchgeführt an dem folglich auch, neben den normalen Testern, die eigentlichen Mitarbeiter (Programmierer, Arts etc.) mitarbeiten. Sollten also auch hier schon Fehler im Gamedesign/Gameplay auffallen, können sofort alle Beteiligten zusammen in einem Diskurs ihre Meinung zur Problemlösung beitragen.

2.4 Betatest

Betatests werden durchgeführt sobald das Spiel vollkommen spielbar ist und alle spielrelevanten Elemente komplett implementiert sind.

Folglich steht hierbei nun nur noch die Prüfung der eigentlichen Spielfunktionalität wie Spielspass, Gameplay, Balancing gegenüber dem technischen Aspekt im Vordergrund. Aus diesem Grund werden bei diesem Test mehr interne sowie externe Tester, statt wie beim Alpha-Test die eigentlichen Mitarbeiter, eingesetzt.

Sind dieses internen (genannt: ClosedBeta) Betatests erfolgreich verlaufen, werden im Folgenden OpenBeta-Tests durchgeführt. Hierbei wird das Spiel an eine kontrollierte Anzahl von Personen(die Community) veröffentlicht. Wichtig sind dabei das Management und die Kommunikation zwischen der Community und der Testabteilung, die sich nun aktiv um die Betreuung dieser kümmert. Dies wird zu meist über ein extra eingerichtetes Forum gelöst, in dem die Community Probleme, mit entsprechender Beschreibung, melden kann. Diese OpenBetaTests werden aber nicht bei jedem Projekt zwingend durchgeführt. Grund dafür ist, das bedingt durch den relativ geringen Projektumfang bei der Entwicklung von CasualGames gegenüber Vollpreisspielen, dies oft nicht rentabel wäre, bzw. alle relevanten Tests und Qualitätssicherungsmaßnahmen intern erfolgen können.

3. Qualitätsmerkmale

3.1 Detailed Design Document (DDD)

Im Gegensatz zur üblichen Herangehensweise in der Softwareentwicklung erstellt *InstantGames* für ihre Spielprojekte anstelle von einem Pflichtenheft ein DDD. Dies betrifft aber lediglich die reine Spieleentwicklung und nicht die Engine- und Toolentwicklung.

Definition

Ein DDD wird häufig als die 'Bibel der Spieleentwickler' bezeichnet und bildet die Grundlage der professionellen Spieleentwicklung.

Das DDD wird von unterschiedlichen Personen gelesen was zwangsweise zu unterschiedlichen Anforderungen führt. Zu den Lesern zählen die Publisher, die Programmierer, die QM und Testing sowie Designer und eventuelle Outsourcing-Partner.

Für die Erstellung, Bearbeitung und Wartung des DDD greifen wir auf ein selbst entwickeltes Wiki, welches im Intranet läuft, zurück. Dies hat mehrere Gründe. Zum einen lassen sich damit komplexe und umfangreiche Projekte, bei gleichzeitiger, hoher Akzeptanz bei den Benutzern etc., managen und zum anderen bringt ein Wiki an sich schon viele Vorteile mit. So lassen sich beispielsweise übersichtliche Layouts leicht erstellen und verwalten, Einträge verlinken, SVN ähnliche Wiederherstellungen durchführen, Einträge abonnieren aber auch Grafiken, Musik und Galerien leicht integrieren.

Welche Form und welche inhaltlichen Aspekte ein DDD abdecken muss wird im Folgenden näher erläutert.

Form

Kriterium	Beschreibung
Kurz fassen	<ul style="list-style-type: none"> • nicht umständlich oder unverbindlich ausdrücken • Designerprosa vermeiden
Einheitliches Layout	<ul style="list-style-type: none"> • einheitliche Strukturen definieren • einheitliche Formatierung und Schreibstil • Formatvorlagen; Styleguides
Redundanzen vermeiden	<ul style="list-style-type: none"> • Redundanzen vermindern Lesbarkeit und lenken vom Wesentlichen ab • Aktualisierung wird erschwert • klare Priorisierung von Designinhalten • eindeutige Zuordnung von einzelnen Features
Designbegründungen von Featureregeln trennen	<ul style="list-style-type: none"> • Abhandlung in einem FAQ nach Designeintrag begründen • Auch Alternativen und Problemstellung darin erläutern
Bilder, Diagramme und Flowcharts	<ul style="list-style-type: none"> • 'Bilder sagen mehr als Worte' • werden schneller erfasst als Fließtext • - helfen komplexe Inhalte zu erfassen
Imperativ statt Konjunktiv	<ul style="list-style-type: none"> • Konjunktiv vermindert Aussagekraft • 'hätte, wäre, wenn, könnte' etc. (Verbindlichkeit von Regeln) • - Design wirkt so unsicher und austauschbar
Verbindliches Glossar	<ul style="list-style-type: none"> • aktuelles und Team-vertrautes Glossar • gewährleistet reibungslose Kommunikation
Aktualität	<ul style="list-style-type: none"> • unerlässlich auch im fortschreitenden Produktionsprozess da sonst nutzlos

Inhalt

Im Folgenden werden die Rahmenkriterien tabellarisch aufgeführt und erläutert.

Kriterium	Beschreibung
Grundlagen	<ul style="list-style-type: none"> • Rahmendaten und strategische sowie spieltheoretische Aspekte • Genre, Hardware-Voraussetzung, Engine, Plattform, <i>Unique Selling Propositions</i> • Mission Statement → die Vision des Spieles • Spielflussbeschreibung aus Spielersicht (vgl. <i>MS 60 seconds of Gameplay</i>) • Lernkurven, Spielertypen, Gameflow, Herausforderung ↔ Belohnung
Spielregeln	<ul style="list-style-type: none"> • Beschreibung der Regeln zu allen Features (Merkmal des Spieles) • Abgrenzung: kein Content/Assets • Spielmodi's und KI • Kamera • Physik • Tag/Nachtwechsel; Wettersystem
Spielinhalte	<ul style="list-style-type: none"> • alle Assets (Grafik, Text, Levels, Models, Charaktere, Gebäude) - Story, Missionen; Szenarien • hohe Anpassungsrate → übersichtliche undanpassungsfähige Plattform (Mindmaps, Tabellen etc.)
Interface	<ul style="list-style-type: none"> • Menüs (auch ingame); Verzahnung, Aufbau • Schrifttypen; Farbsprache; Fließtext • Optionen - Tooltips, Pop-Ups → Mock-Ups, Diagramme, Styleguides usw. • Steuerung; Maus, Tastatur, Controller
Grafik	<ul style="list-style-type: none"> • Styleguides • künstlerische und technische Ausrichtung der grafischen Elemente • Polygonanzahl, Lichtstimmung, Animationen (Emotionen) → Artbook → Figuren, Gegenstände, Terrain, Icons, Hintergrundgrafiken usw.
Sound	<ul style="list-style-type: none"> • Musikstücke und Soundeffekte • Welche Emotionen soll die Musik transportieren? • Wie viel und Wann? • Lokalisierung? • Jingles oder lange Hintergrundmusik?
Tools	<ul style="list-style-type: none"> • Klärung der Tools für Content • Toolchain • Lizenzen • Bedienung • Anforderungen (Middleware oder Eigenentwicklung)

3.2 Spielspaß

Der Spielspaß wird durch die folgenden 4 Gruppen definiert:

- Der eigentliche Inhalt,
- Die Atmosphäre,
- Die Benutzerfreundlichkeit,
- Die technischen Aspekte.

Inhalt:

Ein Casual Game sollte einen angemessenen Umfang besitzen, wobei angemessen bedeutet, dass dieser spielabhängig definiert, und im DDD festgehalten, wird. Ebenfalls angemessen sollte die Verfügbarkeit der auswählbaren Schwierigkeitsgrade sein. Typischerweise sind Casual Games sehr linear aufgebaut, was die Anzahl der verschiedenen Lösungswege stark limitiert, so dass sich gerade Gelegenheitsspieler nicht im Spiel "verlieren". Daraus resultiert eine eingeschränkte Handlungsfreiheit.

Die Komplexität der Handlung bei Casual Games ist sehr gering, jedoch steigt diese mit dem Umfang des Spiels. Gleiches gilt für die Charakterentwicklung, dennoch sollte das Level-/Missions-Design abwechslungsreich sein. Da die Komplexität der Handlung stark eingeschränkt ist, wird die Motivation über freischaltbare Extras aufrecht erhalten.

Abhängig vom Schwierigkeitsgrad muss eine angemessene KI des Gegners vorhanden sein. Die Möglichkeiten des Spielers und die Möglichkeiten der KI müssen in Abhängigkeit des Schwierigkeitsgrads in einem angemessenen Verhältnis stehen.

Atmosphäre:

Wie bei jedem Spiel sollte die Atmosphäre auch bei Casual Games stimmig sein, da die Komplexität der Handlung sehr gering ist, macht ein Großteil der Atmosphäre der Grafikstil und der Sound aus. Jedoch sollten die vorhandenen Dialoge abwechslungsreich sein, die Handlung gut transportieren und keine Logiklücken aufweisen.

Die Grafik muss vom Stil zum Spiel passen und darf nicht völlig gegen die ersten Erwartungen der Spieler sprechen.

Der Sound muss von der Stimmung zu den jeweiligen Situationen passen und darf zu keiner Zeit störend in den Spielfluss eingreifen.

Benutzerfreundlichkeit:

Damit ein Spiel als benutzerfreundlich eingestuft werden kann, muss es auf den vorgegebenen Systemen (Mindestanforderung bzw. empfohlene Anforderungen) ohne Verzögerung bedienbar sein, das bedeutet, dass das Spiel auf Benutzereingaben sofort und ohne Wartezeiten reagiert. Des Weiteren muss die Steuerung intuitiv und leicht erlernbar sein. Das bedeutet ebenfalls, dass die Menüsteuerung gut organisiert und leicht überblickbar ist.

Technische Aspekte:

Neben den atmosphärischen Aspekten der Grafik und des Sounds haben diese ebenfalls noch rein technischen Aspekte.

Die Grafik sollte so hoch aufgelöst sein, dass es dem Benutzer jederzeit möglich ist, alle Aspekte des Bildschirminhalts klar zu erkennen, das schließt Größe des Wiedergabegerätes (13" Monitor - 50" TV), genauso wie die Auflösung des Wiedergabegerätes (kleiner Röhrenfernseher - FullHD TV oder großer TFT Monitor) mit ein. Clipping-Effekte sind zu vermeiden.

Der Sound sollte so aufgelöst sein, dass zu keiner Zeit ein Rauschen oder Klirren auftritt. Die Lautstärke muss so reguliert werden, dass es zu keinen Lautstärkesprüngen kommt, d.h. dass es zu keiner Zeit für den Benutzer nötig ist, dass dieser selbst diese anpassen muss.

Einer der wichtigsten technischen Aspekte ist die Stabilität der Software. Die Lauffähigkeit dieser muss zu jedem Zeitpunkt gewährt bleiben und darf unter keinen Umständen gestört werden.

Wie wird getestet?

Getestet wird der Spielspaß durch eine qualifizierte Tester Gruppe. Diese besteht aus erfahrenen Spielern, die zu gleichen Teilen aus den Altersgruppen 14-16, 17-20, 21-28, 29-39 und 40+ zusammengesetzt wird. Jeder Tester füllt vor Testbeginn einen Fragebogen zu seiner Person aus, damit sein Testergebnis in die richtige Kategorie eingeordnet werden kann und seine Erfahrung ermittelt wird. Diese Tester spielen dann das Spiel und bewerten anhand eines Fragenkataloges einzelne Kriterien mit Punkten von 1 bis 10.

Der Fragenkatalog ist in mehrere Themenbereiche gegliedert. Je nach Spiel werden einzelne Bereiche in den Spielspaßtest miteinbezogen. So benötigt ein Spiel das keine Charakter Entwicklung beinhaltet dazu auch keine Fragen.

Jeder der Themenbereiche besteht aus mehreren Einzelfragen, welche, wie bereits erwähnt mit Punkten zwischen 0 und 10 beantwortet werden können. Die Noten der einzelnen Fragen werden mit einer definierten Gewichtung in die Gesamtnote des Bereiches verrechnet. Jeder Themenbereich erhält auch eine Gewichtung, damit sein Wert für den gesamt Spielspaß angegeben werden kann.

Die Gewichtung der Themenbereiche, sowie der einzelnen Fragen werden zu Beginn eines Projektes in Zusammenarbeit mit dem Kunden festgelegt. Auch die Auswahl der der Fragen und Themenbereiche wird dabei ermittelt.

Bewertungsskala		
Bewertung	Kurzbewertung	Punkte
1+	Herausragend	10
1	Sehr gut	9
2+	Gut bis sehr gut	8
2	Gut	7
3+	Befriedigend bis gut	6
3	Befriedigend	5
4+	Ausreichend bis befriedigend	4
4	Ausreichend	3
5+	Mangelhaft bis ausreichend	2
5	mangelhaft	1
6	ungenügend	0

Themenbereich

Motivation:

- Handlung
- Freischaltbares

Umfang

Linearität /Handlungsfreiheit

KI

Balancing

Level Design

Charakter Entwicklung

Atmosphäre

- Transport der Handlung
- Dialoge
- Abwechslungsreichtum
- Sound
- Grafik

Menüführung

HUD

- Aufbau
- Anpassbarkeit
- Übersichtlichkeit

Reaktionsgeschwindigkeit

Technik

3.3 Korrektheit

Definition:

Ein Software-System gilt als korrekt, wenn es sich genauso verhält wie im DDD festgelegt.

Vorabmaßnahmen:

Um von Beginn an die Wahrscheinlichkeit der Entstehung von Fehlern vorzubeugen, wird nach Möglichkeit auf bereits erfolgreich getestete Algorithmen und fertige Module zurückgegriffen.

Des Weiteren wird ein umfangreicher und komplexer Softwareentwurf angefertigt. Das beinhaltet unter anderem den Einsatz von Design Patterns.

Test-Fälle:

Modul- und Unit-Test:

Bei einem Modultest werden komplette Klassen oder Pakete auf ihre Funktion hin getestet. Der Modul-Test hingegen testet lediglich einzelne Methoden auf ihre Funktionale-Korrektheit.

Bei der Entwicklung werden Tests dieser Art parallel zum Quelltext der Anwendung erstellt und gepflegt. Dies ermöglicht das schnelle Auffinden von Fehlern, die durch Änderungen entstehen können. Ein Komponententest ist ein ausführbares Codefragment, das das sichtbare Verhalten einer

Komponente (z. B. einer Klasse) verifiziert und dem Programmierer eine unmittelbare Rückmeldung darüber gibt, ob die Komponente das geforderte Verhalten aufweist oder nicht. Dabei werden der Komponente festgelegte Werte (normale Arbeitsparameter und Parameter aus den Grenzbereichen bzw. außerhalb des Arbeitsbereiches) und Zufallswerte übergeben und geschaut wie die Komponente darauf reagiert.

Kompletttest:

Ein Kompletttest ist der Test der gesamten Software, der in mehreren Phasen stattfindet. Er beginnt mit den ersten Prototypen und führt sich fort bis hin zu den Alpha- und Betatests. Diese wurden bereits in dem Kapitel "QM in Spielen" beschreiben.

Hardwarekompatibilität:

Tests auf die Hardwarekompatibilität finden auf verschiedenen, vordefinierten Testumgebungen statt. Diese Testumgebungen bestehen aus 20 PC's, wobei alle sechs Monate zwei neue, gleichstarke PC's mit unterschiedlichen Komponenten gekauft werden. Und die ältesten PC's werden jeweils ausrangiert. Hierbei ist das Ziel inkompatible Hardware und die Mindestanforderungen, bei denen das Spiel flüssig läuft zu identifizieren. Flüssig bedeutet eine Mindestbildwiederholrate von 30 FPS.

Mathematische Verifikation:

Auf eine mathematische Verifikation wird auf Grund der zu hohen Kosten im Verhältnis des Nutzens verzichtet.

4. Fragebogen zum Spielspaß

Fragenbereich **Motivation**

Frage	Gewicht	Punkte	Kommentar
Handlung			
Ist die Handlung des Spieles umfangreich genug?			
Hat die Handlung genügend Tiefgang?			
Freischaltbares			
Sind die freischaltbaren Items sinnvoll?			
Gibt es genügend Freischaltbares?			

Kommentar zur Motivation:

Fragenbereich **Umfang**

Frage	Gewicht	Punkte	Kommentar
Ist der Umfang des Spieles angemessen?			
Existieren genügend Schwierigkeitsgrade?			
Sind die Schwierigkeitsgrade angemessen?			
Existieren genügend Level/Missionen?			

Kommentar zum Umfang:

Fragenbereich **Linearität /Handlungsfreiheit**

Frage	Gewicht	Punkte	Kommentar
Ist die Zahl der Lösungswege ausreichend?			
Bietet das Spiel Handlungsfreiheit?			
Reicht die Handlungsfreiheit aus?			

Kommentar zur Linearität /Handlungsfreiheit:

Fragenbereich **KI**

Frage	Gewicht	Punkte	Kommentar
Wie gut reagiert die KI?			
Ist die Stärke der KI ausreichend?			
Ist die KI ausreichend lernfähig?			

Kommentar zur KI:

Fragenbereich **Level Design**

Frage	Gewicht	Punkte	Kommentar
Sind die Level abwechslungsreich genug?			
Ist die Schwierigkeit der Level ausreichend?			

Kommentar zum Level Design:

Fragebereich **Balancing**

Frage	Gewicht	Punkte	Kommentar
Wie gut ist das Balancing des Spieles?			
Sind die Gegner angemessen stark?			
Sind die Fähigkeiten des Charakters stark genug?			

Kommentar zum Balancing:

Fragenbereich **Charakterentwicklung**

Frage	Gewicht	Punkte	Kommentar
Sind die Möglichkeiten der Entwicklung ausreichend?			
Passt die Entwicklung ins Konzept des Spieles?			

Kommentar zur Charakterentwicklung:

Fragenbereich **Reaktionsgeschwindigkeit**

Frage	Gewicht	Punkte	Kommentar
Reagiert das Spiel zuverlässig auf Eingaben?			
Reagiert das Spiel schnell auf Eingaben?			

Kommentar zur Reaktionsgeschwindigkeit:

Fragenbereich **Atmosphäre**

Frage	Gewicht	Punkte	Kommentar
Transport der Handlung			
Wirkt die Handlung schlüssig?			
Wird die Handlung ausreichend thematisiert?			
Dialoge			
Sind die Dialoge glaubwürdig?			
Sind die Dialoge passend?			
Abwechslungsreichtum			
Sind die Dialoge abwechslungsreich?			
Ist die Hintergrundmusik abwechslungsreich?			
Sound			
Passen die Sounds zum Spiel?			
Ist die Hintergrundmusik gut?			
Grafik			
Ist der Stil der Grafik passend?			
Ist der Einsatz der Grafiken angemessen?			

Kommentar zur Atmosphäre:

Fragenbereich **Menüführung**

Frage	Gewicht	Punkte	Kommentar
Sind die Wege im Menü zu lang? (Verzahnung)			
Ist das Menü übersichtlich?			
Passt der Styleguide des Menüs zum Rest des Spieles?			
Ist die Beschriftung des Menüs eindeutig?			

Kommentar zur Menüführung:

Fragenbereich **Technik**

Frage	Gewicht	Punkte	Kommentar
Läuft das Spiel stabil?			
Lief das Spiel ohne Grafikfehler?			
Lief das Spiel ohne Soundprobleme?			
Lässt sich die Lautstärke gut regeln?			
Sind die Grafiken gut zu erkennen?			

Kommentar zur Technik:

Fragenbereich **HUD**

Frage	Gewicht	Punkte	Kommentar
Aufbau			
Ist das HUD gut strukturiert?			
Haben Sie alle notwendigen Informationen erhalten?			
Anpassbarkeit			
Konnten sie das HUD nach Ihren Wünschen anpassen?			
Übersichtlichkeit			
Ist die Größe der Einzelelemente ausreichend?			
Ist die Positionierung der Elemente intuitiv verständlich?			

Kommentar zum HUD: