Realisierungsbericht

|  |  |
| --- | --- |
| **Status** | In Arbeit / In Prüfung / Abgeschlossen |
| **Projektname** | PyJump |
| **Projektleiter** | Dominik Schütz |
| **Auftraggeber** | Daniel Sterchi |
| **Autoren** | Dominik Schütz, Raphael Schwob |
| **Verteiler** | Daniel Sterchi, Dominik Schütz, Raphael Schwob |

**Änderungskontrolle, Prüfung, Genehmigung**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Datum | Beschreibung, Bemerkung | Name oder Rolle |
| 1.0 | 22.03.16 | Erstellung dieses Dokuments | Dominik Schütz |
| 1.1 | 29.03.16 | Weiterführung dieses Dokuments | Dominik Schütz |
| 1.2 | 26.04.16 | Weiterführung dieses Dokuments | Dominik Schütz |
|  |  |  |  |

**Definitionen und Abkürzungen**

|  |  |
| --- | --- |
| Begriff / Abkürzung | Bedeutung |
|  |  |

**Referenzen**

|  |  |
| --- | --- |
| Referenz | Titel, Quelle |
| [1] |  |
| [2] |  |
| [3] |  |

**Inhaltsverzeichnis**

[1 Zusammenfassung 3](#_Toc410826873)

[2 Technische Detailspezifikation 3](#_Toc410826874)

[2.1 Systemdesign 3](#_Toc410826875)

[2.1.1 Struktur 3](#_Toc410826876)

[2.1.2 Beschreibung der Elemente 3](#_Toc410826877)

[2.2 Schnittstellendefinitionen 3](#_Toc410826878)

[2.3 Sicherheit (ISDS) 3](#_Toc410826879)

[2.4 Anforderungszuordnung 3](#_Toc410826880)

[3 Systemdokumentation 4](#_Toc410826881)

[3.1 Konfigurations-Dokumentation 4](#_Toc410826882)

[3.2 Benutzerhandbuch 4](#_Toc410826883)

[3.2.1 Systemübersicht 4](#_Toc410826884)

[3.2.2 Anwenderfunktionalität 4](#_Toc410826885)

[3.3 Supporthandbuch 4](#_Toc410826886)

[3.3.1 Massnahmen bei Benutzerproblemen 4](#_Toc410826887)

[3.3.2 Massnahmen bei technischen Problemen 4](#_Toc410826888)

[3.3.3 Anhang zum Supporthandbuch 4](#_Toc410826889)

[4 Systemtest 5](#_Toc410826890)

[4.1 Testspezifikation 5](#_Toc410826891)

[4.1.1 Kritikalität der Funktionseinheit 5](#_Toc410826892)

[4.1.2 Testanforderungen 5](#_Toc410826893)

[4.1.3 Testverfahren 5](#_Toc410826894)

[4.1.4 Testkriterien 5](#_Toc410826895)

[4.1.5 Testfälle 5](#_Toc410826896)

[4.2 Testprozedur 6](#_Toc410826897)

[4.2.1 Vorbereitung 6](#_Toc410826898)

[4.2.2 Durchführung 6](#_Toc410826899)

[4.2.3 Nachbearbeitung 6](#_Toc410826900)

[4.3 Testprotokoll 6](#_Toc410826901)

[4.3.1 Testobjekt 6](#_Toc410826902)

[4.3.2 Testresultate 6](#_Toc410826903)

[4.3.3 Testauswertung 6](#_Toc410826904)

[5 Weiterführung der Projektplanung 6](#_Toc410826905)

[5.1 Abgleich von Planung und tatsächlichem Verlauf der Phase Konzept 6](#_Toc410826906)

[5.2 Aktualisierung der Risikosituation 6](#_Toc410826907)

[5.3 Planung der nächsten Phase 6](#_Toc410826908)

**Abbildungsverzeichnis**

Abbildung 1 Systemarchitektur 3

# Zusammenfassung

Dieses Dokument beschreibt unsere Arbeiten während der Realisierungsphase. Im Abschnitt 2 Technische Detailspezifikation wird das Systemdesign, die Schnittstellen und die Umsetzung der Sicherheitsanforderungen beschrieben. Zudem wird mit der Anforderungszuordnung der Nachweis erbracht, dass alle Anforderungen durch unser Systemdesign abgedeckt werden.

Im dritten Abschnitt wird die Systemdokumentation behandelt. Dieser Abschnitt beinhaltet ein Benutzer- und Supporthandbuch.

Der vierte Abschnitt widmet sich dem Testen. Testspezifikation, Testprozedur und das Testprotokoll wird dort festgehalten.

Im fünften Abschnitt wird die weitere Projektplanung behandelt. Diese umfasst einen Soll/Ist Vergleich mit der Zeitplanung, die aktualisierte Risikosituation und die Planung der nächsten Phase.

Geben Sie hier eine kurze Zusammenfassung des Inhalts dieses Dokumentes. Wozu dient das Dokument (Zweck) und welche Informationen enthält es?

# Technische Detailspezifikation

## Systemdesign

Im Konzeptbericht haben Sie die Systemarchitektur dargestellt. Dort haben Sie beschrieben aus welchen Elementen (Module und Schnittstellen) Ihr System bestehen soll. Während der Realisierung des Systems setzen Sie diese Architektur um und verfeinern diese, wo nötig, schrittweise bis hinab zur effektiven Konfiguration. Eventuell kommen weitere Elemente hinzu, andere müssen gegebenenfalls aufgeteilt oder anders angeordnet werden.

### Struktur



Abbildung Systemarchitektur

Dieser Abschnitt zeigt und beschreibt den Aufbau des technischen und organisatorischen Systems **bis auf Elementebene.**  
Grundlage für den Systemdesign bilden die Überlegungen im Konzeptbericht, Kapitel ´Systemarchitektur´.

### Beschreibung der Elemente

|  |  |  |
| --- | --- | --- |
| Modul | Beschreibung | Verwendung |
| main.py | Main File des Games | Steuert die Logik sowie alle Prozesse rund um den Inhalt des Spiels.  Zudem werden alle Subklassen von hier aus kontrolliert. |
| platform.py | Enthält die Klasse für die Plattformen | Die Klasse wird als Grundlage für alle Variationen von Plattformen die im Spiel vorkommen können verwendet.  Die Bewegungen sowie spezielle Attribute der Plattformen werden hier kontrolliert. |
| player.py | Enthält die Klasse für eine Spielfigur die vom Spieler gesteuert werden kann | In dieser Klasse werden die Bewegungen des Spielers aufgenommen und verarbeitet.  Dabei werden auch die verschiedenen Usereingaben verarbeitet. |
| monster.py | Hauptklasse für verschiedene Typen von Monstern | Die möglichen Variationen von Monstern die im Spiel vorkommen können verwenden alle diese Klasse als Grundlage.  Die Klasse steuert das Verhalten und vor allem die Bewegungen der Monster. |

Beschreibung der einzelnen, im Systemdesign enthaltenen Module.

## Schnittstellendefinitionen

Externe Schnittstellen

Als einzige externe Schnittstelle dient das GUI vom Game. Durch die simple Bedienung per Tastatur kann der Spieler das Spiel steuern. Zudem gibt es vor Start eines Games immer einen kleinen Dialog in dem ein paar Optionen wie z.B. der Schwierigkeitsgrad angepasst werden kann. Durch das GUI, welches jeweils vor Spielbeginn aufgerufen wird, kann der Spieler alle nötigen Interaktionen mit dem Spiel tätigen. Dieses GUI wird auch im Pausenmenu angezeigt, welches mit der Taste „P“ aufgerufen wird.

|  |  |
| --- | --- |
| Schnittstelle | Beschreibung |
| S1 | Diese Schnittstelle dient der Verbindung zwischen dem Benutzer und dem Menu. Über diese Schnittstelle kann der Benutzer die Spieloptionen einstellen. |
| S2 | Mit dieser Schnittstelle wird die Spielfigur vom Modul player.py durch die Tastatureingaben vom Benutzer gesteuert. |

Interne Schnittstellen

Als interne Schnittstelle dient, wie in Punkt 3 bereits beschrieben, die Main Klasse des Spiels. Über diese Klasse können die einzelnen Module untereinander und mit der Hauptklasse selbst kommunizieren. Die Hauptklasse dient dabei als Koordinator der ganzen Kommunikation, sie delegiert, wenn nötig gewisse Aufgaben an die einzelnen Module. Zudem basiert das Spielfeld auf der Mainklasse, wodurch sie auch verantwortlich für das ganze Geschehen auf dem Spielfeld ist.

|  |  |
| --- | --- |
| Schnittstelle | Beschreibung |
| S3 | Das Modul main.py steuert über diese Schnittstelle das Modul player.py. |
| S4 | Das Modul main.py steuert über diese Schnittstelle das Modul platform.py. |
| S5 | Das Modul main.py steuert über diese Schnittstelle das Modul monster.py. |

Die im Konzeptbericht spezifizierten externen und internen Schnittstellen werden beschrieben.

## Sicherheit (ISDS)

In diesem Abschnitt werden organisatorische und technische Massnahmen zur Sicherstellung von Verfügbarkeit und Authentizität der Daten als Anforderungen formuliert. Zudem werden die Anforderungen an den Schutz sensibler Daten vor unbefugtem Zugriff und vor missbräuchlicher Verwendung definiert.

Da vom Benutzer grundsätzlich keine Daten gesammelt werden und das Spiel über keine Verbindung zu anderen Systemen verfügt, ist die Sicherheit schon auf einem sehr tiefen Level gewährleistet. Die einzigen Daten die abgespeichert werden, sind die Scores die der Spieler bei einem Durchlauf erreicht hat. Dazu wird ein Nickname, den der Benutzer selbst wählen kann abgespeichert. Falls der Benutzer also nicht seinen richtigen Namen eingeben möchte, so kann er das auch tun.

Die Scores werden in einem Textfile abgespeichert, welches danach lokal auf dem Rechner abgelegt wird. Die Datei wird lokal gehalten und sollte von keinem anderen System einsehbar sein. Selbst wenn der Spieler also seinen echten Namen eingeben sollte, so sollten diese Daten auf dem lokalen System sicher sein.

Informationssicherheit

|  |  |  |
| --- | --- | --- |
| Anforderung | Beschreibung | Massnahme |
| A1 | Der Quellcode für das PyJump-Spiel muss gesichert werden. | Der Quellcode wird durch den Head of Development auf einem externen Medium gesichert. Dabei werden die verschiedenen Versionen unterschieden. |
| A2 | Projektdokumente müssen gesichert werden. | Die jeweils aktuellen Versionen der Projektdokumente werden durch den Projektleiter auf einem externen Medium gesichert. |
| A3 | Bei einem technischen Defekt des Kundengeräts muss das PyJump-Spiel auf dem Ersatzgerät des Kunden installiert werden können. | Das PyJump-Spiel wird auf einem externen Medium gesichert und kann von dort aus auf das Ersatzgerät des Kunden installiert werden. |
| A4 | Fahrlässiges Handeln der Mitarbeiter darf nicht zu Datenverlust führen. | Unsere Mitarbeiter arbeiten sehr sorgfältig. Zudem werden regelmässig Backups auf externen Medien erstellt. |
| A5 | Dokumentationen dürfen nicht für jedermann ersichtlich sein. | Die Dokumentationen werden auf einem Medium mit Zugangskontrolle gesichert. |

Datenschutz

|  |  |  |
| --- | --- | --- |
| Anforderung | Beschreibung | Massnahme |
| A1 | Persönliche Kundendaten müssen geschützt werden. | Unsere Mitarbeiter wurden im Umgang mit sensiblen Daten geschult. Sensible Daten werden jeweils auf einem Medium mit Zugangskontrolle gesichert. |
| A2 | Sensible Daten müssen vor Missbrauch geschützt werden. | Sensible Daten werden auf einem Medium mit Zugangskontrolle gesichert. Zudem haben unsere Mitarbeiter eine Vertraulichkeitserklärung unterschrieben. |

Stellen Sie die technische und organisatorische Umsetzung der Sicherheits- und Datenschutzanforderungen dar.

## Anforderungszuordnung

Es ist der Nachweis zu erbringen, dass durch das Systemdesign alle gestellten Anforderungen erfüllt werden. Die einzelnen Anforderungen werden den Systemelementen, in welchen sie realisiert werden, tabellarisch zugeordnet.  
Diese Übersicht ist auch eine wesentliche Grundlage für die Planung und Vorbereitung der durchzuführenden Tests.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| AFo.-Nr | Anforderung (Stichwort) | main.py | platform.py | player.py | monster.py |
| A1 | Malwarefrei | **x** | **x** | **x** | **x** |
| A2 | Offline verfügbar | **x** | **x** | **x** | **x** |
| A3 | Installationsfrei | **x** | **x** | **x** | **x** |
| A4 | Endlosspiel | **x** |  |  |  |
| A5 | Punkte aktualisieren | **x** |  |  |  |
| A6 | Werbefrei | **x** | **x** | **x** | **x** |

# Systemdokumentation

## Konfigurations-Dokumentation

Netzwerk- oder Systemdokumentation mit allen Konfigurationsdaten, geordnet nach Elementen.

## Benutzerhandbuch

### Systemübersicht

Gesamtzusammenhänge als Überblick für den Anwender:

* Ziele und Hauptfunktionen des Systems
* Struktur des Systems und externe Schnittstellen
* Allgemeines zu Sicherheit, Datenschutz, Anwenderrollen.

Diese Applikation stellt ein endlos Platformer dar. Das Ziel ist es mit der Spielfigur über mehrere Plattformen zu springen und so an höhe zu gewinnen. Der Spieler kann solange weiterspielen bis er einen Fehler macht und hinunterfällt. Je höher der Spieler springt ohne einen Fehler zu machen desto mehr Punkte erhält er.

Wie unter Abschnitt 2.1 Systemdesign ersichtlich ist wird das Spiel in mehrere Module aufgeteilt. Das hat den Vorteil, dass das Spiel einfacher erweitert werden kann. Die externen Schnittstellen werden im Abschnitt 2.2 Schnittstellendefinition beschrieben. Eine Schnittstelle wird dafür verwendet, dass der Spieler die Spielfigur steuern kann. Die andere Schnittstelle dient dazu, damit der Spieler im Menu Optionen auswählen kann.

Zur Sicherheit des Spiels muss man sich keine grossen Sorgen machen. Die Applikation wurde Malwarefrei programmiert und es werden keine sensiblen Daten gespeichert. Die einzigen Daten, die gespeichert werden sind die Scores des Spielers. Diese werden zusammen mit einem Nickname, der nicht zwingend echt sein muss gespeichert. Diese Daten werden allesamt lokal auf dem Gerät des Spielers gespeichert und bleiben auch dort, weil das Spiel über keine Verbindung zur Aussenwelt verfügt.

Als Anwenderrollen gibt es nur die des Spielers. Das Spiel kann nach der Bereitstellung ohne Technikwissen angewendet werden.

### Anwenderfunktionalität

Beschreibung der einzelnen Funktionalitäten und Verfahren sowie Anleitungen, welche der Anwender benötigt, um die für ihn relevanten Tätigkeiten durchzuführen.

Relevante Informationen können sein:

* Aufgabe
* Instruktion zu Anwendung und Betrieb
* Initialisierung
* Durchführung   
  (Ausführungsoptionen, Benutzer-Eingabe, Ausführung, erwartete Ausgabe, Beziehung zu anderen Funktionen)
* Terminierung
* Wiederanlauf («Restart»)
* Überwachungsverfahren
* Fehlerfall
* Fehlermeldungen
* Fehlerdiagnosemöglichkeiten
* Fehlerbehebungsmassnahmen
* Wiederherstellungsverfahren («Recovery»)

Allgemein

Bei diesem Spiel handelt es sich um ein Endlos-Platformer. Das heisst das Spiel läuft solange weiter bis der Spieler einen Fehler macht oder das Spiel abbricht. Solange der Spieler ohne Fehler spielt wird der Punktezähler steigen. Wenn man kurz unterbrechen möchte, kann man mit der Taste „P“ das Spiel pausieren. Durch erneutes drücken der Taste „P“ wird das Spiel am gleichen Punkt weiterlaufen.

Dieses Spiel hat keinen Lerninhalt. Es dient einzig und allein dem Unterhaltungszweck.

Startmenü

Sobald man das Spiel startet wird einem als erstes das Startmenü angezeigt. Dort gibt es einen Abschnitt „Hilfe“, bei dem man sich Informationen über die Bedienung des Spiels holen kann. Unter dem Abschnitt „Schwierigkeitsgrad“, welcher ebenfalls im Startmenü ist, kann man die Schwierigkeit des Spiels anpassen. Zudem gibt es im Startmenü noch einen Abschnitt „Info“, wo man Informationen über die Version und die Entwickler des Spiels findet.

Gameplay

Nachdem man im Startmenü einen Schwierigkeitsgrad ausgewählt hat, kann man das Spiel durch drücken der „Leertaste“ starten. Die Spielfigur fängt dann an zu springen sofern sie auf den Plattformen landet. Mit den Tasten „A“ und „D“ kann man die Spielfigur nach links oder rechts bewegen. Wenn man mit der Spielfigur am Rand das Fenster verlässt, so tritt die Spielfigur am anderen Rand wieder ins Fenster.

Während des Spielens können Monster auftreten, die die Spielfigur nicht berühren darf. Wenn die Spielfigur ein Monster berührt zählt das als Fehler und das Spiel wird beendet. Die Spielfigur kann sich aber natürlich auch gegen die Monster wehren. Wenn der Spieler die „Leertaste“ drückt, so kann die Spielfigur Kugeln gegen die Monster schiessen. Wird ein Monster getroffen so verschwindet es aus dem Fenster und die Spielfigur kann vorerst ungehindert weiter springen.

Problembehandlung

Falls beim Spielen Probleme auftauchen kann das Spiel unter Mac Computern mit der Tastenkombination „cmd“ + „Q“ oder über das Menü in der Symbolliste beendet werden. Danach kann das Spiel wieder wie gewohnt gestartet werden. Falls sich das Spiel wie vorhin beschrieben über den üblichen Weg nicht beenden lässt kann mittels der Tastenkombination „cmd“ + „alt“ + „esc“ das „Sofort Beenden“-Menü aufgerufen werden. In diesem Menü kann kann man Python auswählen und dieses dann zum stoppen erzwingen. Danach kann das Spiel ebenfalls wie gewohnt wieder gestartet werden.

Wiederherstellung

Das Spiel kann ohne grossen Aufwand wiederhergestellt werden, für den Fall das trotz neustarten des Spiels immer noch Probleme auftreten. Für die Wiederherstellung wird zuerst der ganze „pyjump“-Ordner gelöscht. Sobald alle alten Dateien vollständig gelöscht wurden kann die ursprüngliche Version wieder auf das Zielgerät kopiert werden.

Falls nach der Neuinstallation des Spiels immer noch Probleme bestehen kann es sein, dass Python ein Problem hat. Das Spiel wurde für Python Version 2.7 entwickelt. Es wird empfohlen diese Version zu installieren. Falls bereits Python 2.7 installiert ist kann eventuell eine Neuinstallation von Python das Problem lösen.

## Supporthandbuch

### Massnahmen bei Benutzerproblemen

Allgemeine Probleme bei der Bedienung des Systems durch die Anwender werden beschrieben und Lösungsmöglichkeiten aufgezeigt.

### Massnahmen bei technischen Problemen

Technische Probleme, welche den Anwender in der Benutzung des Systems hindern, werden beschrieben und Lösungsmöglichkeiten oder Umgehungsmöglichkeiten werden aufgezeigt.

Falls es während des Spielens zu einem Problem kommen sollte, so kann der Benutzer das Spiel abbrechen und Neustarten. Falls das Problem weiterhin besteht sollte zuerst sichergestellt werden, dass das Spiel auch wirklich die Fehlerquelle ist. Falls das zutrifft kann eine Neuinstallation eventuell Abhilfe leisten.

### Anhang zum Supporthandbuch

* technische Erläuterungen und Übersichten
* Fehlermeldungen (inkl. Ursachen und Lösungsmassnahmen)
* Glossar
* Index

# Systemtest

## Testspezifikation

### Kritikalität der Funktionseinheit

|  |  |
| --- | --- |
| Kritikalität | Beschreibung |
| Erfüllt | Test wurde erfolgreich abgeschlossen. Keine Nacharbeitern nötig |
| Teilweise erfüllt | Test wurde teilweise erfolgreich abgeschlossen. Kleine Nacharbeiten nötig. |
| Nicht erfüllt | Test wurde nicht erfolgreich abgeschlossen. Grössere Nacharbeiten nötig. |

### Testanforderungen

Die Tests werden unter normalen möglichst betriebsnahen Bedingungen durchgeführt, sodass wir möglichst realitätstreue Ergebnisse erhalten. Bei den Teilen wo mit Daten gearbeitet wird, werden wir zusätzlich noch mit ungültigen Werten testen.

Die Tests unter Höchstleistung oder mit einem Komponentenausfall durchzuführen ist nicht nötig, da unser Spiel nicht auf Redundanz oder Hochverfügbarkeit ausgelegt ist.

Beschreibt allgemeine Anforderungen an den Test, zum Beispiel:

* Tests sind mit Normal-, Grenz- und fehlerhaften Werten durchzuführen.
* Tests sind unter Normal- und Ausnahmebedingungen (Höchstleistungen, Komponentenausfall, usw.) durchzuführen.

### Testverfahren

Ein Test unterteilt sich in die Abschnitte «**Vorbereitung**», «**Durchführung**» und «**Auswertung**».   
Die Verfahren und Vorgehensweisen werden festgelegt und beschrieben.  
Die Vorbereitung eines Tests umfasst zum Beispiel das Erstellen von Testdaten.   
Die Verfahren der Testdurchführung werden aus der jeweiligen Kritikalität des Testobjekts und weiteren an dieses gestellten QS-Anforderungen ermittelt.

Vorbereitung: Die Software wird auf das Gerät des Kunden kopiert und dann normal gestartet.

Durchführung: Die Tests werden alle gemäss den Testfällen auf der vorbereiteten Umgebung durchgeführt.

Auswertung: Die Tests werden gemäss den Testfällen ausgewertet und im Testprotokoll festgehalten. Die Bewertung der Tests findet gemäss der Kritikalität statt, somit müssen Tests, welche nicht erfüllt wurden behoben werden.

### Testkriterien

Abdeckungsgrad: Es wird so viel getestet, dass garantiert werden kann, dass alle Anforderungen durch das Spiel abgedeckt sind und funktionieren.

Checklisten: Als Checklisten dienen die Testfälle.

Ende-Kriterien: Ein Test gilt als erfolgreich abgeschlossen, wenn er gemäss Kritikalität als „erfüllt“ bewertet wird.

### Testfälle

Unter Berücksichtigung der jeweiligen Testmethoden werden die Testfälle und deren Abdeckung der Anforderungen beschrieben.  
Mit den hier aufgeführten Testfällen sollen die oben genannten Ende-Kriterien ausreichend erfüllbar sein.  
Die Testfallbeschreibungen sollen beinhalten:

* was (Funktion, Genauigkeit, usw.) zu testen ist,
* welche Ausgangssituation hierfür erforderlich ist,
* welche Eingaben (Daten und Signale mit allen für den Test ausschlaggebenden Eigenschaften wie Zeitbedingungen) notwendig sind und
* welche Resultate (Ausgabedaten und Reaktionen/Effekte) zu erwarten sind.

Die in der Konzeptphase im Dokument „Projektführung“ begonnenen Testfalltabellen werden übernommen und weiter detailliert:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nr.** | **AFo-Nr.** | **Anwendungsfall (ggf. orientiert an Use Cases)** | **Ausgangs­situation** | **Eingabe­daten** | **erwartetes Ergebnis** | **Bemerkungen, Prüfergebnis** |
| T1 | A2 | Spiel ist lauffähig ohne Internetverbindung | Das Spiel wird gestartet. |  | Das Spiel startet und zeigt das Startmenü an. |  |
| T2 | A2 | Spiel ist lauffähig ohne Internetverbindung | Der Tester drückt im Startmenü die Leertaste und das Spiel beginnt. |  | Das Spiel startet und läuft bis der Tester einen Fehler macht oder das Spiel abbricht. |  |
| T3 | A4.1  A4.2 | Das Spiel ist an keine Höchstpunktzahl gebunden. | Das Spiel wurde gestartet. |  | Das Spiel läuft weiter bis der Tester einen Fehler macht. |  |
| T4 | A5.1  A5.2 | Der Punktestand wird aktualisiert und angezeigt. | Das Spiel wurde gestartet. |  | Der Punktestand wird oben rechts angezeigt und laufend aktualisiert. |  |
| T5 | A6 | Das Spiel ist werbefrei. | Das Spiel wurde gestartet. |  | Während des Spielens und im Startmenü wird keine Werbung angezeigt. |  |

## Testprozedur

**Bei Bedarf** wird hier je Testfall (gemäss der Testspezifikation) eine Arbeitsanleitung für die Durchführung des Tests erstellt.

### Vorbereitung

Beschreibt den erforderlichen Ausgangszustand für den jeweiligen Testfall.

|  |  |
| --- | --- |
| **Nr.** | **Vorbereitung** |
| T1 | * Spiel wird auf das Testgerät installiert * Netzwerkverbindung wird getrennt |
| T2 | * Spiel wird auf das Testgerät installiert * Netzwerkverbindung wird getrennt |
| T3 | * Spiel wird auf das Testgerät installiert * Netzwerkverbindung wird getrennt * Der Tester beginnt das Spiel durch drücken der „Leertaste“ |
| T4 | * Spiel wird auf das Testgerät installiert * Netzwerkverbindung wird getrennt * Der Tester beginnt das Spiel durch drücken der „Leertaste“ |
| T5 | * Spiel wird auf das Testgerät installiert * Netzwerkverbindung wird getrennt * Der Tester beginnt das Spiel durch drücken der „Leertaste“ |

Voraussetzungen:  
Beschreibt die Voraussetzung, d.h., wie die Testvoraussetzungen beschaffen sind und der geforderte Ausgangszustand für den Test herzustellen ist.

* Python 2.7 muss installiert sein
* Pillow package muss installiert sein
* je nach Installationsart muss eine Internetverbindung vorhanden sein

Konfiguration:  
Beschreibt, wie das Testobjekt zu konfigurieren und zu installieren ist und wie es mit der Testumgebung in Verbindung steht.

### Durchführung

Für jeden Schritt werden die Handgriffe und Interaktionen beschrieben.   
Bei automatisierten Tests ist der Testablauf zu beschreiben.

### Nachbearbeitung

Anleitung zur Auswertung und zur Sicherung der Testresultate   
(welche Resultate sind wie zu dokumentieren, o.ä.).

## Testprotokoll

### Testobjekt

Identifizierung das Testobjekts (mit Version): .......  
Tester, Ort, Datum und Zeit der Testdurchführung.

### Testresultate

Die durch den Test ermittelten Resultate werden festgehalten.   
End- und Zwischenresultate und ggf. weitere notwendige Informationen zum Test werden in chronologischer Folge aufgezeichnet.   
Die Resultate werden den erwarteten Resultaten (siehe Kapitel ´Testspezifikation´) gegenüber gestellt.

### Testauswertung

Abweichungen der Testresultate zu den Systemanforderungen werden festgehalten und deren Einfluss auf die Funktionstüchtigkeit des Systems beurteilt. Die möglichen Fehlerursachen werden genannt.  
Zeigt sich aus den Testresultaten ein bestimmter Trend im Auftreten gleichartiger Mängel, so werden diesbezügliche Vermutungen hier dokumentiert.

# Weiterführung der Projektplanung

## Abgleich von Planung und tatsächlichem Verlauf der Phase Konzept

Soll/Ist-Vergleich bezüglich der Zeitplanung für die Phase Realisierung.

Welche Risiken sind während der Phase Realisierung Realität geworden? Wie wurde diese Situation gemeistert? Welche Konsequenzen ergeben sich daraus für das gesamte Projekt?

## Aktualisierung der Risikosituation

Haben sich an der Risikosituation Änderungen (Verschärfung oder Entschärfung, neue Risiken) ergeben? Dann beschreiben Sie hier die Risikosituation neu. Führen Sie wiederum konkrete Massnahmen auf, wie Sie den Risiken in der nächsten Phase begegnen wollen.

## Planung der nächsten Phase

Aktualisieren und verfeinern Sie hier Ihren Projektplan, vor allem in Bezug auf die nächste Phase.