

# Realisierungsbericht

<b>Status</b>	In Arbeit / <u>In Prüfung</u> / Abgeschlossen
<b>Projektname</b>	PyJump
<b>Projektleiter</b>	Dominik Schütz
<b>Auftraggeber</b>	Daniel Sterchi
<b>Autoren</b>	Dominik Schütz, Raphael Schwob
<b>Verteiler</b>	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
1.3	07.05.16	Weiterführung dieses Dokuments	Dominik Schütz
2.0	10.05.16	Fertigstellung dieses Dokuments	Dominik Schütz

## Definitionen und Abkürzungen

Begriff / Abkürzung	Bedeutung
GUI	Graphical User Interface
Malware	Software, die vorsätzlich Schaden verursacht
Platformer	Spielart
Python	Scriptsprache
PIP	Paketverwaltung für Pythonpakete
easy_install	Paketverwaltung für Pythonpakete
Terminal	Kommandozeile

**Inhaltsverzeichnis**

1	Zusammenfassung .....	3
2	Technische Detailspezifikation .....	3
2.1	Systemdesign .....	3
2.1.1	Struktur .....	3
2.1.2	Beschreibung der Elemente .....	3
2.2	Schnittstellendefinitionen .....	4
2.3	Sicherheit (ISDS) .....	4
2.4	Anforderungszuordnung .....	5
3	Systemdokumentation .....	6
3.1	Konfigurations-Dokumentation .....	6
3.2	Benutzerhandbuch .....	6
3.2.1	Systemübersicht .....	6
3.2.2	Anwenderfunktionalität .....	7
3.3	Supporthandbuch .....	8
3.3.1	Massnahmen bei Benutzerproblemen .....	8
3.3.2	Massnahmen bei technischen Problemen .....	8
3.3.3	Anhang zum Supporthandbuch .....	8
4	Systemtest .....	9
4.1	Testspezifikation .....	9
4.1.1	Kritikalität der Funktionseinheit .....	9
4.1.2	Testanforderungen .....	9
4.1.3	Testverfahren .....	9
4.1.4	Testkriterien .....	9
4.1.5	Testfälle .....	9
4.2	Testprozedur .....	11
4.2.1	Vorbereitung .....	11
4.2.2	Durchführung .....	11
4.2.3	Nachbearbeitung .....	11
4.3	Testprotokoll .....	12
4.3.1	Testobjekt .....	12
4.3.2	Testresultate .....	12
4.3.3	Testauswertung .....	12
5	Weiterführung der Projektplanung .....	13
5.1	Abgleich von Planung und tatsächlichem Verlauf der Phase Konzept .....	13
5.2	Aktualisierung der Risikosituation .....	13
5.3	Planung der nächsten Phase .....	14

**Abbildungsverzeichnis**

Abbildung 1	Systemarchitektur .....	3
Abbildung 2	Zeitplanung nach Konzeptphase .....	13
Abbildung 3	effektive Zeitplanung .....	13
Abbildung 4	Zeitplanung Einführungsphase .....	14

## 1 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.

## 2 Technische Detailspezifikation

### 2.1 Systemdesign

#### 2.1.1 Struktur

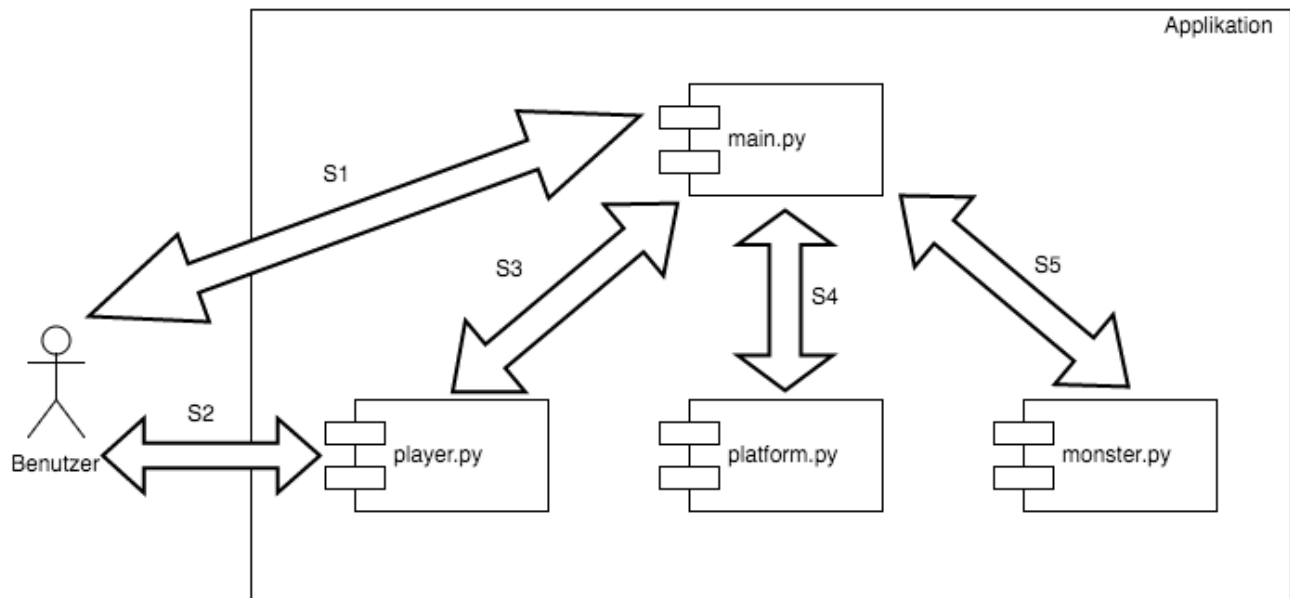


Abbildung 1 Systemarchitektur

#### 2.1.2 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.

## 2.2 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 angepasst werden können. 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.

## 2.3 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.

## 2.4 Anforderungszuordnung

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

### 3 Systemdokumentation

#### 3.1 Konfigurations-Dokumentation

Bei diesem Projekt wird ein Platformer-Spiel umgesetzt, welches Offline genutzt werden kann. Somit ist eine besondere Netzwerkkonfiguration nicht nötig. Bei diesem Spiel gibt es auch keine anderen Konfigurationsdateien. Das Verhalten des Spiels wird durch ein paar wichtige Variablen gesteuert. Diese wichtigen Variablen werden nachführend aufgelistet und erklärt.

##### Globale Variablen

Variable	Bemerkung	Standardwert
WIDTH	Breite des Fensters	500
HEIGHT	Höhe des Fensters	750
DELAY	Dauer des Timers	50
ROOT_DIR	Wurzelverzeichnis	os.path.abspath(os.path.join(__file__, "..", ".."))
VERSION	Aktuelle Version des Spiels	1.0.0

##### Lokale Variablen von initGame

Variable	Bemerkung	Standardwert
gameOver	Boolean bestimmt ob das Spiel noch läuft	False
gamePaused	Boolean bestimmt ob das Spiel pausiert ist	False
score	Punktestand des Spielers	0

##### Lokale Variablen von initObj

Variable	Bemerkung	Standardwert
bg_img	Definiert das Hintergrundbild	ImageTk.PhotoImage(file=ROOT_DIR + "/pyjump/gfx/bg.png")
player_img	Definiert die Spielfigur	ImageTk.PhotoImage(file=ROOT_DIR + "/pyjump/gfx/rocket.png")
monster_img	Definiert das Monster	ImageTk.PhotoImage(file=ROOT_DIR + "/pyjump/gfx/monster.png")
topBarFont	Definiert Schriftgrösse	tkFont.Font(size="20")

#### 3.2 Benutzerhandbuch

##### 3.2.1 Systemübersicht

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.

### 3.2.2 Anwenderfunktionalität

#### 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.

#### Installation

Da das Spiel in der Scriptsprache Python umgesetzt wurde ist eine Installation nicht nötig. Es reicht, wenn einfach alle benötigten Dateien an den gewünschten Ort kopiert werden. Gegebenenfalls muss jedoch ein Python-Modul nachinstalliert werden. Dafür muss zuerst mit Administratorrechten der Python Packagemanager pip installiert werden. Dies kann zum Beispiel über den Terminal mit dem Befehl „sudo easy\_install pip“ getan werden. Danach muss mit pip das Pillow-Modul installiert werden. Dies geschieht wieder über den Terminal mit dem Befehl „sudo pip install Pillow“. Auch hierfür werden wieder Administratorrechte benötigt.

#### 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. Zudem gibt es im Startmenü noch einen Abschnitt „Info“, wo man Informationen über die Version und die Entwickler des Spiels findet.

#### Gameplay

Im Startmenü kann man das Spiel dann 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 Symbolleiste 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 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.

## 3.3 Supporthandbuch

### 3.3.1 Massnahmen bei Benutzerproblemen

Beim PyJump-Spiel gibt es eigentlich keine Benutzerprobleme. Es kann aber vorkommen, wie bei jeder Applikation, dass sich das Spiel aufhängt. Bei solchen Problemen ist das Vorgehen gleich wie bei anderen Applikationen. Das Spiel muss neugestartet werden. Dies kann mit der Tastenkombination „cmd“ + „Q“ oder „cmd“ + „alt“ + „esc“ getan werden.

Alle anderen Benutzerprobleme sind auf falsche Bedienung zurückzuführen. Um das zu verhindern müssen die Benutzer geschult werden. Im Abschnitt 3.2 Benutzerhandbuch ist die Bedienung des Spiels ausführlich beschrieben.

### 3.3.2 Massnahmen bei technischen Problemen

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.

Möglicherweise kann das Spiel nicht gestartet werden, weil das Pillow-Modul fehlt. Falls dies der Fall wäre würde eine entsprechende Fehlermeldung erscheinen. Um dies zu beheben muss zuerst mit Administratorrechten der Python Packagemanager pip installiert werden. Dies kann zum Beispiel über den Terminal mit dem Befehl „sudo easy\_install pip“ getan werden. Danach muss mit pip das Pillow-Modul installiert werden. Dies geschieht wieder über den Terminal mit dem Befehl „sudo pip install Pillow“. Auch hierfür werden wieder Administratorrechte benötigt.

Falls nun immer noch Probleme auftauchen sollte überprüft werden ob Python richtig funktioniert. Falls dies nicht der Fall ist kann eventuell eine Neuinstallation von Python das Problem lösen. Zudem sollte beachtet werden, dass dieses Spiel für die Python Version 2.7 entwickelt wurde. Unter anderen Python Versionen kann sich das Spiel möglicherweise anders verhalten.

### 3.3.3 Anhang zum Supporthandbuch

#### Glossar

Stichwort	Beschreibung
PIP	Paketverwaltungsprogramm für Python-Pakete
Pillow	Python Library für Bilder
sudo	Befehl mit Administratorrechten ausführen
easy_install	Pythonmodul um Pythonpakete zu installieren
Terminal	Kommandozeile
Python	Scriptsprache



## 4 Systemtest

### 4.1 Testspezifikation

#### 4.1.1 Kritikalität der Funktionseinheit

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

#### 4.1.2 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.

#### 4.1.3 Testverfahren

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.

#### 4.1.4 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.

#### 4.1.5 Testfälle

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.	Erfüllt
T2	A2	Spiel ist lauffähig ohne	Der Tester drückt im		Das Spiel startet und	Erfüllt

		Internetverbindung	Startmenü die Leertaste und das Spiel beginnt.		läuft bis der Tester einen Fehler macht oder das Spiel abbricht.	
T3	A2	Spiel ist lauffähig ohne Internetverbindung	Das Spiel wurde gestartet.	Der Tester versucht die Spielfigur durch drücken der Taste „A“ nach links zu bewegen.	Die Spielfigur bewegt sich nach links.	Erfüllt
T4	A2	Spiel ist lauffähig ohne Internetverbindung	Das Spiel wurde gestartet	Der Tester versucht die Spielfigur durch drücken der Taste „D“ nach rechts zu bewegen.	Die Spielfigur bewegt sich nach rechts.	Erfüllt
T5	A2	Spiel ist lauffähig ohne Internetverbindung	Das Spiel wurde gestartet	Der Tester drückt die Taste „P“ um das Pausenmenu aufzurufen.	Das Spiel pausiert.	Erfüllt
T6	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.	Erfüllt
T7	A5.1 A5.2	Der Punktestand wird aktualisiert und angezeigt.	Das Spiel wurde gestartet.		Der Punktestand wird oben rechts angezeigt und laufend aktualisiert.	Erfüllt
T8	A6	Das Spiel ist werbefrei.	Das Spiel wurde gestartet.		Während des Spielens und im Startmenü wird keine Werbung angezeigt.	Erfüllt

## 4.2 Testprozedur

### 4.2.1 Vorbereitung

Nr.	Vorbereitung
T1	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li></ul>
T2	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li></ul>
T3	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>
T4	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>
T5	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>
T6	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>
T7	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>
T8	<ul style="list-style-type: none"><li>• Spiel ist auf dem Testgerät installiert</li><li>• Netzwerkverbindung ist getrennt</li><li>• Der Tester beginnt das Spiel durch drücken der „Leertaste“</li></ul>

Voraussetzungen:

- Python 2.7 muss installiert sein
- Pillow Modul muss installiert sein
- Je nach Installationsart muss eine Internetverbindung vorhanden sein

Konfiguration:

Es wird keine besondere Konfiguration benötigt. Es muss lediglich das Spiel auf das Testgerät kopiert werden. Falls Pillow noch nicht installiert ist kann dies wie folgt über den Terminal erledigt werden:

```
sudo easy_install pip
sudo pip install pillow
```

### 4.2.2 Durchführung

Die Testdurchführung und der Ablauf werden in den jeweiligen Testfällen ausführlich beschrieben.

### 4.2.3 Nachbearbeitung

Die Resultate sollen unter dem Abschnitt 4.3.2 Testresultate dokumentiert werden. Im Abschnitt 4.3.3 Testauswertung soll dann ein Fazit aus den Testresultaten gezogen werden.

### 4.3 Testprotokoll

#### 4.3.1 Testobjekt

Nr.	Tester	Ort	Datum	Zeit	Bemerkungen
T1	Dominik Schütz	Grosshöchstetten	10.05.16	20:45	Testfall erfüllt
T2	Dominik Schütz	Grosshöchstetten	10.05.16	20:48	Testfall erfüllt
T3	Dominik Schütz	Grosshöchstetten	10.05.16	20:50	Testfall erfüllt
T4	Dominik Schütz	Grosshöchstetten	10.05.16	20:52	Testfall erfüllt
T5	Dominik Schütz	Grosshöchstetten	10.05.16	20:55	Testfall erfüllt
T6	Dominik Schütz	Grosshöchstetten	10.05.16	20:58	Testfall erfüllt
T7	Dominik Schütz	Grosshöchstetten	10.05.16	21:01	Testfall erfüllt
T8	Dominik Schütz	Grosshöchstetten	10.05.16	20:03	Testfall erfüllt

#### 4.3.2 Testresultate

Nr.	Testergebnis	Erwartetes Ergebnis	Bemerkungen
T1	Das Spiel ist gestartet und das Startmenü wird angezeigt	Das Spiel startet und zeigt das Startmenü an.	Testfall erfüllt
T2	Das Spiel läuft bis ich einen Fehler mache.	Das Spiel startet und läuft bis der Tester einen Fehler macht oder das Spiel abbricht.	Testfall erfüllt
T3	Die Spielfigur bewegte sich nach links.	Die Spielfigur bewegt sich nach links.	Testfall erfüllt
T4	Die Spielfigur bewegte sich nach rechts.	Die Spielfigur bewegt sich nach rechts.	Testfall erfüllt
T5	Das Spiel hält an und es wird ein Label mit der Aufschrift „Pause“ angezeigt.	Das Spiel pausiert.	Testfall erfüllt
T6	Das Spiel stoppt sobald ich einen Fehler mache.	Das Spiel läuft weiter bis der Tester einen Fehler macht.	Testfall erfüllt
T7	Der Punktestand wird laufend aktualisiert. Sobald ich einen Fehler mache bleibt der Punktestand am entsprechenden Punkt stehen	Der Punktestand wird oben rechts angezeigt und laufend aktualisiert.	Testfall erfüllt
T8	Im ganzen Spiel war keine Werbung sichtbar	Während des Spielens und im Startmenü wird keine Werbung angezeigt.	Testfall erfüllt

#### 4.3.3 Testauswertung

Die Testprozedur verlief sehr gut wie erwartet. Alle acht Testfälle konnten erfolgreich abgeschlossen werden. Die Resultate der Testfälle deckten sich jeweils mit den erwarteten Ergebnissen. Somit müssen keine Nacharbeiten mehr erledigt und wir können unser Produkt mit gutem Gewissen dem Kunden übergeben.

## 5 Weiterführung der Projektplanung

### 5.1 Abgleich von Planung und tatsächlichem Verlauf der Phase Konzept

Wie man unten sieht deckt sich die Zeitplanung aus der Konzeptphase nicht mit den Effektiven Zeiten. Das liegt unter anderem daran, dass die Abgabefrist für die Realisierungsphase um zwei Wochen verlängert wurde. Dies kam uns sehr gelegen, weil wir diese Zeit gut brauchen konnten um das Spiel zu realisieren. Wir haben uns zudem entschieden eine Funktion um den Schwierigkeitsgrad anzupassen nicht einzubauen, damit wir diese Zeit investieren können um das Spiel ausführlicher zu testen. Die Funktion für den Schwierigkeitsgrad war auch nicht als Anforderung definiert, weshalb es nicht so tragisch ist, dass die Funktion nicht umgesetzt wurde.

Schwerwiegende Risiken trafen während der Realisierungsphase nicht ein, weshalb wir auch mit kaum Konsequenzen kämpfen mussten. Da allerdings die Zeit etwas knapp wurde mussten wir wie oben gesagt einen kleinen Abstrich machen und die verschiedenen Schwierigkeitsgrade weglassen.

Zeitplanung nach Konzeptphase:

Nr.	Projektphase	April					
		22.03.16	29.03.16	05.04.16	12.04.16	19.04.16	26.04.16
1	main.py						
2	platform.py						
3	player.py						
4	monster.py						
5	Abschlusstests						

Abbildung 2 Zeitplanung nach Konzeptphase

Effektive Zeitplanung:

Nr.	Projektphase	März		April				Mai	
		22.03.16	29.03.16	05.04.16	12.04.16	19.04.16	26.04.16	03.05.16	10.05.16
1	main.py								
2	platform.py								
3	player.py								
4	monster.py								
5	Abschlusstests								

Abbildung 3 effektive Zeitplanung

### 5.2 Aktualisierung der Risikosituation

Die Risikosituation hat sich über die gesamte Sicht nicht geändert. Man könnte höchstens von einer leichten Entschärfung sprechen, weil nun schon weite Teile des Projekts abgeschlossen sind.

## 5.3 Planung der nächsten Phase

		Mai		
Nr.	Projektphase	17.05.16		24.05.16
1	Spiel installieren			
2	Kunde instruieren			
3	Abnahme durchführen			
4	Abschluss vorbereiten			

Abbildung 4 Zeitplanung Einführungsphase