

26.9.2016

Iman Lünsmann

css Versicherung



Anforderungsanalyse A Simple Project

Version 1.0

Inhalt

[2 Revisionen 3](#_Toc468346846)

[3 Einleitung 4](#_Toc468346847)

[4 Management Summary 5](#_Toc468346848)

[5 Referenzen 5](#_Toc468346849)

[6 Glossar 5](#_Toc468346850)

[7 Projektteam 5](#_Toc468346851)

[8 Anfangszustand (IST), Ziele (SOLL) 5](#_Toc468346852)

[9 Anforderungsanalyse 6](#_Toc468346853)

[9.1 Identifizierung der Akteure 6](#_Toc468346854)

[9.2 Anforderungskatalog 6](#_Toc468346855)

[9.2.1 Anforderungen des Akteurs 6](#_Toc468346856)

[9.2.2 Zusatzinformationen 6](#_Toc468346857)

# Revisionen

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Name | Kommentar |
| 28.11.2016 | 1.0 | Iman Lünsmann |  |
|  |  |  |  |

# Einleitung

Liebe Leserinnen und Leser

In diesem Dokument beschreibe ich was ich entwickeln werde. Ich werde innerhalb von 10 Tagen das Spiel Pong programmieren. Das Spiel wurde 1972 von Atari veröffentlicht und gilt als Urvater der Videospiele.

Das Endprodukt ist für Gamer Freunde ausgelegt und für diejenigen bestimmt, die sich für den Prozess hinter den Kulissen interessieren.

Mit dieser Auflage von *A Simple Project* will ich das Leben verschiedener Gamer verschönern indem sie mein innovatives Spiel spielen und Freude an dem haben.

Bei Fragen und Verbesserungsvorschlägen wenden Sie sich bitte an mich.



Vielen Dank für Ihr Interesse

Iman Lünsmann, Informatiklernende der CSS Versicherung

# Systemidee

In den nächsten 10 Tagen werde ich das Spiel Pong programmieren Ich will, dass die Spieler Freude an meinem Spiel haben und es wenn möglich Ihren Freunden zeigen.

Das Spiel Pong wurde von Atari 1972 entwickelt und ist der Urvater aller Videospiele. In diesem Spiel gibt es 2 Spieler (Du, Computer oder du, Freund). Das Spielprinzip ist simpel. Ein Ball bewegt sich zwischen den Spielern hin und her. Die Spieler müssen den Ball dann mit dem «Schläger» zum Gegner schlagen. Lässt man den „Ball“ am „Schläger“ vorbei, erhält der Gegner einen Punkt.

Für den Anfang werde ich das Spiel so programmieren das man gegen eine Wand spielt. Das Spiel endet, wenn man all seine 3 Leben aufgebraucht hat.

Wenn man aber gegen einen Freund Spielt, endet das Spiel, wenn jemand 10 Punkte hat.

# Management Summary

Es soll ein Spiel namens Pong entwickelt werden. Das Spielprinzip von Pong ist simpel und ähnelt dem des [Tischtennis](https://de.wikipedia.org/wiki/Tischtennis): Ein Punkt („Ball“) bewegt sich auf dem Bildschirm hin und her. Jeder der beiden Spieler steuert einen senkrechten Strich („Schläger“), den er mit einem Drehknopf ([Paddle](https://de.wikipedia.org/wiki/Paddle_(Eingabeger%C3%A4t))) nach oben und unten verschieben kann. Lässt man den „Ball“ am „Schläger“ vorbei, erhält der Gegner einen Punkt.

Das Spiel endet, wenn man im Einzelspielermodus gegen den Computer seine 3 Leben aufgebraucht hat.

Im Mehrspielermodus gewinnt derjenige der zuerst 10 Punkte hat.



Quelle: [https://de.wikipedia.org/wiki/Pong#/media/File:Pong.png](https://de.wikipedia.org/wiki/Pong" \l "/media/File:Pong.png)

# Referenzen

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Titel/ Beschreibung | Autor | Dateiname oder URL |
| 1 | Pong | Wikipedia | https://de.wikipedia.org/wiki/Pong |
| 2 |  |  |  |

# Glossar

|  |  |
| --- | --- |
| Begriff | Beschreibung |
|  |  |

# Projektteam

|  |  |  |
| --- | --- | --- |
| Name | Funktion | Kontakt |
| Iman Lünsmann | Entwicklerin | Iman.luensmann@gmail.com |
| Urs Nussbaumer | Auftraggeber | Urs.nussbaumer@ict-bz.ch |

# Ziele (SOLL)

In diesem Kapitel wird beschrieben, was mein Ziel ist (SOLL).

Mein Ziel ist es dieses Spiel zu programmieren sodass man für den Anfang gegen eine Wand spielen muss, also quasi gegen den Computer. Ich will das Gamer Freunde gefallen an meinem Spiel finden und es gerne spielen.

# Anforderungsanalyse

Anforderungen an das zu entwickelnde System definieren alle zu erfüllenden Eigenschaften oder die zu erbringende Leistung, sowie allfällige technische Vorgaben und weitere Forderungen des Kunden im Zusammenhang mit dem (den) zu erstellenden Produkt(en).

## Identifizierung der Akteure

Ein Akteur bezeichnet eine Rolle, die mit dem zu entwickelnden System interagieren wird Es wird folgender Akteur geben:

1. **Player:** Dieser Akteur hat die Möglichkeit das Spiel zu spielen.

## Anforderungskatalog

Der Anforderungskatalog ist eine priorisierte Liste, die alles enthält, was im zu entwickelnden Produkt enthalten sein soll. Jede Anforderung wird in Alltagssprache -in der Form einer „User-Story“ -formuliert, mit einer ID zu späteren Identifizierung versehen sowie mittels 3 verschiedener Kategorien priorisiert: 1 = hohe Prorität, 2 = mittlere Priorität, 3 = keine Priorität. Diese drei Prioritäten repräsentieren die Verpflichtungen ‚must‘, ‚should‘, ‚nice to have‘.

Im Verlauf der Zeit können neue Anforderungen hinzukommen und/oder bestehende Anforderungen

können wegfallen. Anforderungen, die wegfallen, sind im Dokument zu belassen und als gestrichen zu markieren.

Hinter einer Anforderung kann ihn eckigen Klammern in der Form [Z#] eine Zusatzinformation hinterlegt werden und mit [F#] bzw. [F#, F#, ...] können eine oder mehrere offene Fragen referenziert werden.

### Anforderungen des Akteurs

|  |  |  |
| --- | --- | --- |
| ID | Anforderung | Prio |
| REQ. 001 | Als Spieler möchte ich das Spiel starten können. | 1 |
| REQ. 005 | Als Spieler möchte ich den Score sehen. | 1 |
| REQ. 002 | Als Spieler möchte ich die Schwierigkeit des Spiels ändern. **[Z01]** | 2 |
| REQ. 003 | Als Spieler möchte ich mit dem Benutzernamen anmelden können. | 2 |
| REQ. 004 | Als Spieler möchte ich, dass der Ball sich je nachdem wo ich ihn getroffen habe dem Winkel entsprechend anpasst. | 2 |
| REQ. 005 | Als Spieler möchte ich verschiedene Modi haben. **[Z02]** | 2 |

### Zusatzinformationen

Z01 3 Schwierigkeitsstufen (Noob, Expert, Torment). Die Schwierigkeit besteht darin, dass sich der Ball schneller bewegt.

Z02 Training: Du gegen eine Wand, das Spiel endet, wenn man seine 3 Leben aufgebraucht hat.

Einzelspieler 2: Du gegen den PC der auch einen Schläger hat, das Spiel endet, wenn man 10 Punkte hat.

Mehrspieler: Du spielst gegen einen Freund, das Spiel endet, wenn jemand 10 Punkte hat.