Ulimate-Battlesnake

Anforderungsanalyse

|  |  |
| --- | --- |
| **Autor(en):** | Joshua Odermatt |
| **Version:** | 1.0 |
| **Erstellt am:** | 5. Dezember 2019 |
| **Letzte Änderung:** | 11. Dezember 2019 |
| **Speicherort:** | Dokument1 |

Inhalt

[1 Revisionen 3](#_Toc531702604)

[2 Einleitung 4](#_Toc531702605)

[2.1 Systemidee 4](#_Toc531702606)

[2.2 Management Summary 4](#_Toc531702607)

[2.3 Projektteam 4](#_Toc531702608)

[2.4 Glossar 4](#_Toc531702609)

[3 Ausgangslage (IST), Problembereiche 5](#_Toc531702610)

[3.1 Beschreibung der Ausgangslage 5](#_Toc531702611)

[3.2 Problembereiche und Schwachstellen 5](#_Toc531702612)

[4 Ziele (SOLL) 6](#_Toc531702613)

[4.1 Beschreibung der Ziele 6](#_Toc531702614)

[4.2 Produktperspektive, Nutzen 6](#_Toc531702615)

[4.3 Zielkonflikte 6](#_Toc531702616)

[4.4 Abgrenzung 6](#_Toc531702617)

[5 Anforderungsanalyse 7](#_Toc531702618)

[5.1 Identifizierung der Akteure 7](#_Toc531702619)

[5.2 Anforderungskatalog 8](#_Toc531702620)

[5.2.1 F.REQ: Funktionale Anforderungen 8](#_Toc531702621)

[5.2.2 NF.REQ: Nichtfunktionale Anforderungen 9](#_Toc531702622)

[6 Systemablaufmodelle (Aktivitäten) 10](#_Toc531702623)

[6.1 Aktivität "DVD ausleihen" 10](#_Toc531702624)

[6.2 Aktivität «XY» 10](#_Toc531702625)

[7 Risiko-Analyse 11](#_Toc531702626)

[7.1 Risikokatalog 11](#_Toc531702627)

[8 Anhang 12](#_Toc531702628)

[8.1 Termine 12](#_Toc531702629)

[8.2 Referenzen 12](#_Toc531702630)

# Revisionen

|  |  |  |
| --- | --- | --- |
| **Datum** | **Name** | **Kommentar** |
| 11.12.2019 | Joshua Odermatt | Dokument erstellt. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Einleitung

## Systemidee

Ich habe vor, als Projekt ein Snakespiel auf dem Web zu Programmieren. Bei meinem Snakespiel wird man in jeder Runde gegen einen anderen Spieler antreten. Wer den grösseren Score erreicht, wird mit Punkten belohnt. Anhand dieser Punkte wird es dann ein Ranking geben. Hat man einen Gewissen Rang erreicht, wird man mit einem Abzeichen belohnt.

* Funktion 1: Der Spieler kann das Snake Spielen
* Funktion 2: Der Spieler kann mit guter Leistung auf dem Ranking austeigen.
* Funktion 3: Der Spieler kann sich ein Profil erstellen.

## Management Summary

## Projektteam

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Kürzel** | **Funktion** | **Kontakt (Telefon, Mail...)** |
| Urs Nussbaumer | nuu | Projektleiter | 041 371 24 28 urs.nussbaumer@ict-bz.ch |
| Joshua Odermatt | odj | Developer | 0794136720  Joshi.odermatt@gmail.com |
|  |  |  |  |
|  |  |  |  |

## Glossar

Die im Projekt verwendeten Fachbegriffe und Abkürzungen in alphabetischer Reihenfolge.

|  |  |
| --- | --- |
| **Begriff** | **Beschreibung** |
| SRS | Software Requirement Specification (Anforderungsspezifikation nach IEEE) |
| SQL | Structured Query Language |
|  |  |
|  |  |
|  |  |
|  |  |

# Ausgangslage (IST), Problembereiche

In diesem Kapitel wird die Ausgangslage beschrieben und es werden die Problembereiche identifiziert, die sich in der heutigen Situation zeigen.

## Beschreibung der Ausgangslage

Es gibt bereits tausende verschiedene Snake’s. Jedoch hat keines bisher besonders viel Langzeitmotivation. Ich möchte das mittels eins spannenden Rankings ändern.

## Problembereiche und Schwachstellen

# Ziele (SOLL)

## Beschreibung der Ziele

Das Hauptziel ist, ein spannendes Snakespiel zu programmieren, in welchem man sich mit echtem Spieler messen kann.

## Produktperspektive, Nutzen

Das Spiel wird zur Unterhaltung oder zum Zeitvertreib eingesetzt.

# Anforderungsanalyse

Die 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

* **Kunde:** Benutzer kann das Spiel spielen.

Spieler

Spiel

Spieler

## Anforderungskatalog

Der Anforderungskatalog ist eine priorisierte Liste, die alles enthält, was im zu entwickelnden Produkt enthalten sein soll. Es wird unterschieden zwischen funktionalen und nicht-funktionalen Anforderungen.

Jede Anforderung ist in Alltagssprache, in der Form einer "User-Story", formuliert, mit einer ID zur 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" und "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.

### A: Funktionale Anforderungen

|  |  |  |
| --- | --- | --- |
| **ID** | **Anforderung** | **Prio** |
| A001 | Der Spieler soll die Schlange auf einem Feld bewegen können. | 1 |
| A002 | Der Spieler kann die Schlange auf einen Apfel zusteuern um grösser zu werden. | 1 |
| A003 | Als Spieler möchte ich ein Profil erstellen können. | 2 |
| A004 | Als Spieler möchte ich gegen einen anderen Spieler spielen können. | 2 |
| A005 | Als Spieler möchte ich auf dem Ranking richtig eingestuft werden. | 2 |
| A006 | Als Spieler möchte auf meinem Profil einige Statistiken über mich sehen können wie zum Beispiel die gesamte Spielzeit usw. | 3 |
| A007 | Als Spieler möchte ich mit einem Bestimmten Benutzer ein Duell spielen. | 3 |
|  |  |  |

#### 

#### NF: Nichtfunktionale Anforderungen

|  |  |  |
| --- | --- | --- |
| **ID** | **Anforderung** | **Prio** |
| A001 | Die Webseite ist mit Angular gestalltet. | 1 |
| A002 | Das Snake ist mit P5\* für javascript geschrieben. | 1 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Systemablaufmodelle (Aktivitäten)

Dieses Kapitel zeigt die wichtigsten und/oder komplexesten funktionalen Anforderungen in ihrem Ablauf. Dazu werden die einzelnen Aktivitätsschritte detailliert analysiert. Für die Darstellung der einzelnen Aktivitätsschritte werden Aktivitätsdiagramme nach UML verwendet.

# Anhang

## Termine

|  |  |  |
| --- | --- | --- |
| **Datum** | **Ziel, Beschreibung, Meilenstein** | **Verantwortlich** |
| 13.12.2019 | Fertigstellung Iteration 1, Prototyp dem Kunden zeigen (Sitzung) |  |
| 12.12.2019 | Meilenstein entscheid ob das Projekt umsetzbar ist. | odj |
|  |  |  |
|  |  |  |
|  |  |  |