Texas Hold’em Casual

Anforderungsanalyse

|  |  |
| --- | --- |
| **Autor(en):** | Daniel Polgar |
| **Version:** | 1.0 |
| **Erstellt am:** | 5. Dezember 2018 |
| **Letzte Änderung:** | 6. Dezember 2018 |

Inhalt

[1 Revisionen 3](#_Toc531856188)

[2 Einleitung 4](#_Toc531856189)

[2.1 Systemidee 4](#_Toc531856190)

[Die wichtigste Funktionen sind: 4](#_Toc531856191)

[2.2 Management Summary 4](#_Toc531856192)

[2.3 Projektteam 4](#_Toc531856193)

[2.4 Glossar 4](#_Toc531856194)

[3 Ausgangslage (IST), Problembereiche 4](#_Toc531856195)

[3.1 Beschreibung der Ausgangslage 5](#_Toc531856196)

[3.2 Problembereiche und Schwachstellen 5](#_Toc531856197)

[4 Ziele (SOLL) 6](#_Toc531856198)

[4.1 Beschreibung der Ziele 6](#_Toc531856199)

[4.2 Produktperspektive, Nutzen 6](#_Toc531856200)

[4.3 Abgrenzung 6](#_Toc531856201)

[4.4 Spielregeln 6](#_Toc531856202)

[5 Anforderungsanalyse 7](#_Toc531856203)

[5.1 Identifizierung der Akteure 7](#_Toc531856204)

[5.2 Anforderungskatalog 8](#_Toc531856205)

[F.REQ: Funktionale Anforderungen 8](#_Toc531856206)

[NF.REQ: Nichtfunktionale Anforderungen 8](#_Toc531856207)

# Revisionen

|  |  |  |
| --- | --- | --- |
| **Datum** | **Name** | **Kommentar** |
| 05.12.2018 | Daniel | Dokument erstellt. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Einleitung

## Systemidee

In Texas Hold’em Casual kann man eine der bekannteste Kasinospiele mit Computer Spielen. Gegen 4 anderen Spielern gibt es eine Chance, um die Poker Fähigkeiten zu testen. Das Spiel garantiert kostenlose Chips jeder Zeit und man muss nicht Tageslang eine Runde Spielen.

### Die wichtigsten Funktionen sind:

* Die Mise eingeben können
* Mise erhöhen
* Mitkommen
* Die Kartenkombinationen verwenden können

## Management Summary

## Projektteam

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Kürzel** | **Funktion** | **Kontakt (Telefon, Mail...)** |
| Urs Nussbaumer | nuu | Projektleiter | 041 371 24 28 urs.nussbaumer@ict-bz.ch |
| Dániel Polgár | pod | Softwareentwickler | 076 271 40 54  danypolgar@gmail.com |

## Glossar

Erläutern Sie alle im Projekt verwendeten Fachbegriffe und Abkürzungen in alphabetischer Reihenfolge.

|  |  |
| --- | --- |
| **Begriff** | **Beschreibung** |
| SRS | Software Requirement Specification (Anforderungsspezifikation nach IEEE) |
| Mise | Das Geld, was man gewinnen kann. |
| brennen | Eine Karte auf der Seite legen und nicht zeigen. |

# 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

Das Spiel spielt man häufig mit echten Karten oder online. Es gibt mehrere Formen dieses Spiels mit anderen Regeln. Texas Hold’em ist der einfachste Poker Spiel. Am häufigsten Spielen das 2 bis 10 Personen. Dieses Spiel kann ziemlich lange dauern, manchmal spielt man mehrere Tage bis das Ganze fertig ist.

## Problembereiche und Schwachstellen

Bei Online Versionen muss man entweder zahlen oder muss man warten, bis man virtuelles Geld bekommt. Die neue Version garantiert kostenlose Chips jederzeit und das Spiel geht immer bis zum Ende rechtzeitig.

# Ziele (SOLL)

In diesem Kapitel werden die übergeordneten Ziele beschrieben, die mit dem zu entwickelnden System erreicht werden sollen.

## Beschreibung der Ziele

Das Ziel des Spiels ist, dass man das Ganze Spiel in weniger Zeit fertig spielen kann. Man muss nicht warten, dass man wieder Geld bekommt und man muss nicht mit echtem Geld zahlen.

## Produktperspektive, Nutzen

Der Auftraggeber wird Texas Hold’em spielen lernen, falls er schon kann, übt er.

## Abgrenzung

Es werden nur 5 Spieler geben und das Budget am Anfang bleibt immer gleich.

## Spielregeln

Das Spiel ist mit französischen Karten gespielt. Es gibt 52 Karten. Jede Karte hat eine Farbe und ein Wert. 2 ist der kleinste, dann kommen die anderen Zahlen aufsteigend, nachher Jumbo, Dame, König und Ass. Die 4 Farben sind:

* ♣ = Treff (Kreuz)
* ♦ = Karo
* ♥ = Herz
* ♠ = Pik

Am Anfang bekommt jeder Spieler 2 Karten. Die Spieler dürfen entscheiden, ob sie ein bestimmter Minimalwert auf die Karten legen oder die Karten wegwerfen. Es gibt 2 Ausnahmen: der Big Blind muss unbedingt das Minimalwert geben, der Small Blind muss sicher die Hälfte der Minimalwert geben. Diese Rollen werden jede in Runde gewechselt. Wenn ein Spieler die Karten wegwirft, muss er bis nächste Runde warten, dass er wieder spielen kann. Wenn ein Spieler einen höheren Wert liegt, müssen die anderen mitkommen, aber dürfen sie es auch erhöhen. Falls ein Spieler nicht mehr mitkommen kann, weil er kein Geld mehr hat, kann er weiterspielen, aber am Schluss bekommt er nicht die ganze Mise. Sobald alle Spieler ihre Werte gelegt haben, kommen die gemeinsame Karten. Eine Karte wird verbrannt bzw. auf der Seite gelegt, nachher kommen 3 Karten in der Mitte. Die Spieler dürfen wieder Werte legen. Wenn sie fertig sind, wird wieder eine Karte verbrannt und eine zu den anderen gelegt. Es geht so lange, bis in der Mitte 5 Karten liegen. Man darf noch am Schluss Werte legen, um die Mise zu erhöhen. Am Ende müssen alle Karten gezeigt werden. Der Stärkste Kombination gewinnt. Der Sieger gewinnt die ganze Mise. Die folgenden Kombinationen sind möglich:

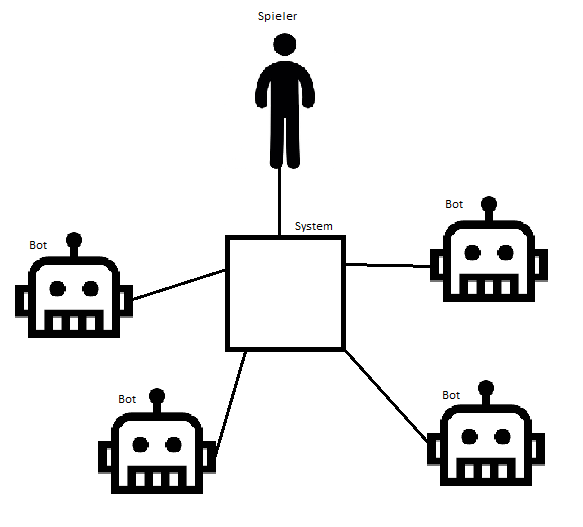
|  |  |
| --- | --- |
| **Kombination**  **(schwächste 🡪 Stärkste)** | **Beschreibung** |
| High Card | Gesamtwert der Karten |
| One Pair | Zweimal eine Kartenwert |
| Two Pairs | Zweimal beide Kartenwerte |
| Three of a kind | Dreimal eine Kartenwert |
| Straight | 5 Karten in eine Reihenfolge nach Wert |
| Flush | 5 Karten mit gleicher Farbe |
| Full house | Eine Kartenwert zweimal und der andere dreimal |
| Four of kind | Eine Kartenwert viermal |
| Straight flush | 5 Karten mit gleicher Farbe und in eine Reihenfolge nach Wert |
| Royal flush | 5 höchste Karten in eine Reihenfolge mit gleicher Farbe |

# 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

* Spieler: kann entscheiden, ob er mitkommt, die Karten wegwirft, oder Mise erhöht.
* Bot: während des Spiels hat das gleich Recht, wie der Spieler, aber darf er das Spiel nicht verlassen.



## 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.

### F.REQ: Funktionale Anforderungen

|  |  |  |
| --- | --- | --- |
| **ID** | **Anforderung** | **Prio** |
| F.REQ.001 | Als Spieler möchte ich bei einem Tisch Spielen mit Karten. | 1 |
| F.REQ.002 | Als Spieler möchte ich jede Runde 2 Karten bekommen, die nur ich sehe. | 1 |
| F.REQ.003 | Als Spieler möchte ich wählen können, welcher Option ich will. (Mise erhöhen, Mise geben, mitkommen und Karte wegwerfen) | 1 |
| F.REQ.004 | Falls nötig, wird eine Karte verbrannt. [Z1] | 1 |
| F.REQ.005 | Als Spieler möchte ich regelmässig neue Spielrollen bekommen. | 1 |
| F.REQ.006 | Als Spieler möchte ich anfangs Runde 3, Mitte Runde 4 und Ende Runde 5 gemeinsame Karten sehen. | 1 |
| F.REQ.007 | Als Spieler möchte ich Ende Runde meine Karten zeigen | 1 |
| F.REQ.008 | Als Spieler kann ich mir ein Name geben und ein Bild einstellen. | 3 |
| F.REQ.009 | Als Spieler möchte ich eine Meldung bekommen, ob ich gewonnen, oder verloren habe. | 3 |
| F.REQ.010 | Als Spieler möchte ich 3 verschiedene Schwierigkeitsgrade Spielen können. [F1] | 3 |

#### Offene Fragen

#### [F1] Nach was sie aufgeteilt sind, ist unbekannt.

#### Zusatzinformationen

[Z1] Es wird immer eine Karte von den anderen weggelegt und nicht gezeigt.

### NF.REQ: Nichtfunktionale Anforderungen

|  |  |  |
| --- | --- | --- |
| **ID** | **Anforderung** | **Prio** |
| NF.REQ.001 | Der Quellcode muss übersichtlich sein. [Z1] | 1 |
| NF.REQ.002 | Der Quellcode muss dokumentiert sein. | 1 |

[Z1] Übersichtlichkeit:

* Sinnvolle Variablen
* Methoden verwenden
* Arrays verwenden
* Gross- Kleinschreibung (Camel Case)