

KonterMatch

Maurus Dähler, Nicolas Brunner, Ernad Hodzic, Mario Weber

16. Mai 2012

KONTER



MATCH

Inhaltsverzeichnis

1	Einleitung	3
2	Spielidee	3
3	Spielregeln	4
3.1	Allgemein	4
3.2	Trumpfen	5
3.3	Während dem Spiel	5
3.4	Spielende	6
4	GUI Erklärung	7
4.1	Login	7
4.2	Lobby	8
4.3	Spieltisch	9
5	Dokumentation	11
5.1	Verwendete Libraries	11
5.2	Funktionalitäten und Grenzen des Spiels	11
5.3	Struktur des Programms	12
5.3.1	Programmstruktur	12
5.3.2	GUI Klassenhierarchie	13
5.4	Starten und Benutzen des Client/Server-Programms	13
5.5	Software-Qualitätsmanagement	14
6	Schlusswort	14

1 Einleitung

Das Spiel KonterMatch wurde unter den Anforderungen der Vorlesung Programmier-Projekt cs108 an der Universität Basel im Frühlingsemester 2012 erstellt. Ziel der Aufgabe ist es, in einem grösseren Java-Programmierprojekt, bestehend aus Gruppen aus 4 Studierenden, ein Computerspiel in Client/Server-Architektur zu konzipieren und zu implementieren.

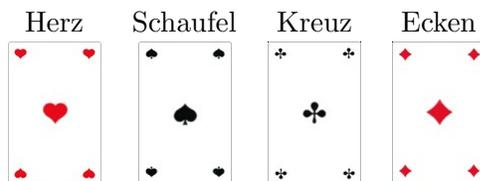
2 Spielidee

Das Spiel Kontermatch ist eine Neuimplementierung des bekannten schweizer Schieber-Jasses. Ein Kartenspiel, welches auf einem Rundenbasierten Spielprinzip aufbaut und sich somit gut in Client/Server Struktur programmieren lässt.

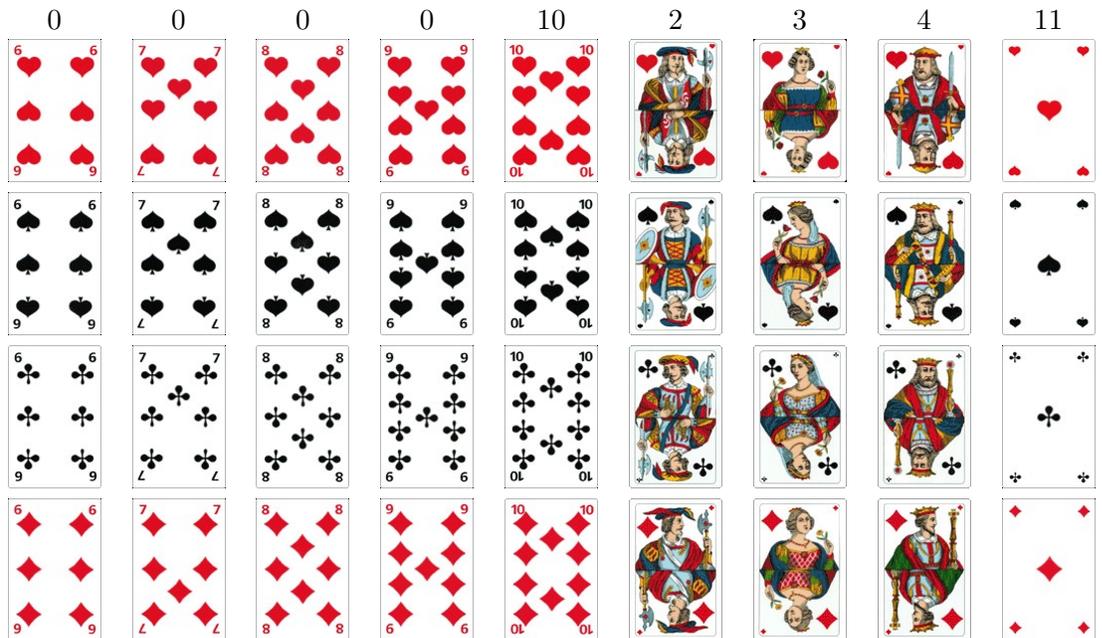
3 Spielregeln

3.1 Allgemein

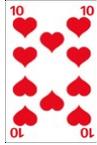
- Gespielt wird mit 4 Personen, welche sich in 2 Teams aufteilen. Also 2 gegen 2.
- Um einem Tisch setzen sich die Teams abwechslungsweise, so dass die verbündeten Spieler sich gegenüber sitzen. (Siehe GUI)
- 36 Karten mit unterschiedlichen Zählwerten.
- 4 Farben:



- Kartenwerte:



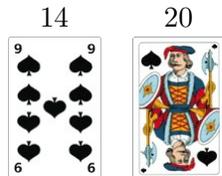
- Jeder Spieler erhält 9 gemischte Karten
- Spieler mit Herz 10 beginnt erstes Spiel.



- Die erzielten Punkte eines Teams werden zusammengezählt und gemeinsam geschrieben.

3.2 Trumpfen

- Bei jeder Spielrunde darf ein neuer Spieler, resp der nächste rechts am Tisch, die Trumpf-Farbe bestimmen. Somit wechselt die Trumpfphase jede Runde zwischen den Teams; und zwar zum nächsten Spieler im Gegenurzeigersinn.
- Wer selber den Trumpf nicht festlegen will, kann zu seinem Partner gegenüber schieben und dieser muss dann den Trumpf bestimmen. Ausspielen muss jedoch derjenige, welcher zuerst an der Reihe war und geschoben hat.
- Das Total der Punkte in einer Runde ist 157. Die Mächtigkeit der Karte sowie Punktwerte der Trumpffarbe ändern sich wie folgt:



3.3 Während dem Spiel

- Gespielt wird im Gegenurzeigersinn. Wichtig ist die als erstes ausgespielte Karte. Folgende Spieler dürfen nur Karten spielen, welche die gleiche Farbe haben, solange sie noch von dieser besitzen. Besitzt ein Spieler keine Karte der ausgespielten Farbe mehr, darf eine beliebige Karte gespielt werden.
Eine Trumpfkarte darf jederzeit gespielt werden. Ausnahme: Untertrumpfen.
- Untertrumpfen: Eine Runde beginnt mit einer Nicht-Trumpfkarte. Spieler zwei spielt einen Trumpf. Den folgenden Spielern ist es nicht erlaubt jetzt eine Trumpfkarte mit niederem Wert, als die bereits liegende, zu spielen. Ausnahme: Der Spieler hat nur noch Trumpfkarten auf der

Hand mit niederem Wert. (Beispiel: Trumpf ist Herz. Spieler1 spielt eine Schaufel 10. Spieler2 sticht mit Trumpf König. Spieler3 darf jetzt kein Herz spielen, welcher niedriger ist als der König. Spieler3 wird also, falls vorhanden, Herz spielen oder eine andere Nicht-Trumpfkarte, fall er kein Herz besitzt.

- Wer die höchste Karte bzw. stärkste Trumpfkarte gegeben hat, sticht und erhält die vier gespielten Karten.
- Der Stich hat einen Wert, der dem Punktwert aller vier Karten entspricht.

3.4 Spielende

Gewonnen hat die Partei, welche zuerst die für den Gewinn erforderliche Punktzahl (1500) erreicht hat.

4 GUI Erklärung

4.1 Login



Abbildung 1: Login Bildschirm

Server starten:

Erstellen Sie einen Server, damit sich andere Spieler mit Ihrem Rechner verbinden können.

Server IP eingeben:

Melden Sie sich an einem Server an, um mit anderen zu Spielen

Spielernamen:

Geben Sie sich einen eigenen SPielernamen

Einloggen:

Weiter zur Spiel-Lobby

Ihre IP:

Teilen Sie anderen Spielern Ihre IP mit, damit sie sich auf Ihrem Server anmelden können.

Eigenen Server erstellen:

Erstellen Sie den Server

4.2 Lobby



Abbildung 2: Lobby Bildschirm

Create:

Öffnen Sie ein neues Spiel und geben Sie dem Spiel einen Namen

Join:

Nehmen Sie an einem bereits geöffnetem Spiel teil

offene Spiele:

Liste der offenen Spiele

Spieler im Chat:

Verfügbare Spieler in der Lobby

Hall of Fame:

Betrachten Sie Highscores der momentanen Serversession

Jassregeln:

Studieren Sie die Regeln verschiedener Jass-Spiele

Chatfenster:

Nachrichtenverlauf der Spieler und des Servers

4.3 Spieltisch



Abbildung 3: Spieltisch Bildschirm

Scorefenster:

Aktueller Punktestand der zwei Teams und die zu erreichenden Siegespunkte

Trumpffenster:

Sehen Sie den die Trumpffarbe der Runde

Senden:

Chatten Sie mit anderen Spielern im Spiel. Kein Flüstern erlaubt.

offene Karten:

Ihre Karten

verdeckte Karten:

Karten der anderen Spieler

Punktestand:

Betrachten Sie den Punktestand der beiden Teams im aktuellen Spiel

Fluchknopf:

Lassen Sie Ihrem Frust freien Lauf und teilen Sie sich anderen Spielern mit

typischen Jass-Flüchen mit.

Ausloggen:

Verlassen Sie den Tisch und kehren Sie in die Lobby zurück.

Jassregeln:

Studieren Sie die Regeln.

Chatfenster:

Nachrichtenverlauf der Spieler und des Servers. Wichtige Meldungen, Spielstatus

5 Dokumentation

5.1 Verwendete Libraries

- java.io.*
- java.net.*
- java.util.*
- java.lang.*
- java.awt.*
- javax.swing.*
- sun.audio.*
- Die Klasse VerticalLabelUI.java stammt aus dem Internet:
<http://tips4java.wordpress.com/2009/02/21/vertical-label-ui/>

5.2 Funktionalitäten und Grenzen des Spiels

Funktionalitäten des Programms:

Client/server Architektur

Chat (private chat in Lobby)

Mehrere Spiele gleichzeitig möglich (nur 1 pro Client)

Soundeffekte

Swing-basierte GUI

JUnit-test überprüfte Stichmethode

Normaler Schieber mit Trumpf

Disconnect Handhabung

Grenzen (aus Zeitgründen):

Kein Obeabe (Spiel ohne Trumpf, höchste Karte gewinnt)

Kein Undenuffe (Spiel ohne Trumpf, tiefste Karte gewinnt)

Kein Weisen

Keine vervielfachung der Punktzahl diverser Spielversionen

5.3 Struktur des Programms

5.3.1 Programmstruktur

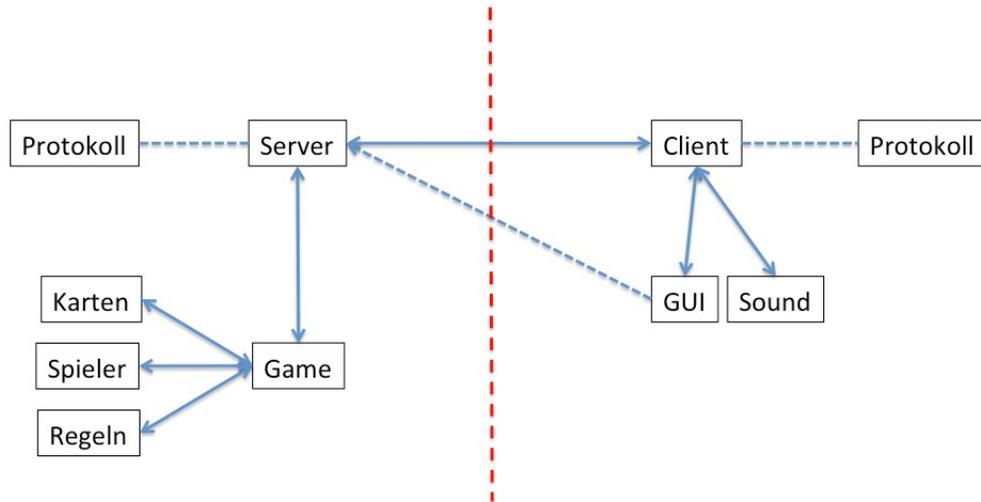


Abbildung 4: Server-Client Struktur getrennt mit Funktionalitäten und den wichtigsten Aufgaben

Der **Server** besitzt sein eigenes Protokoll und kontrolliert das gesamte Spiel. Zum Spiel gehören diverse Klassen wie Spielregeln, Spielerrechte und Kartendefinitionen.

Der **Client** besitzt ebenfalls ein Protokoll und steuert vor allem die GUI und die Soundeffekte.

5.3.2 GUI Klassenhierarchie

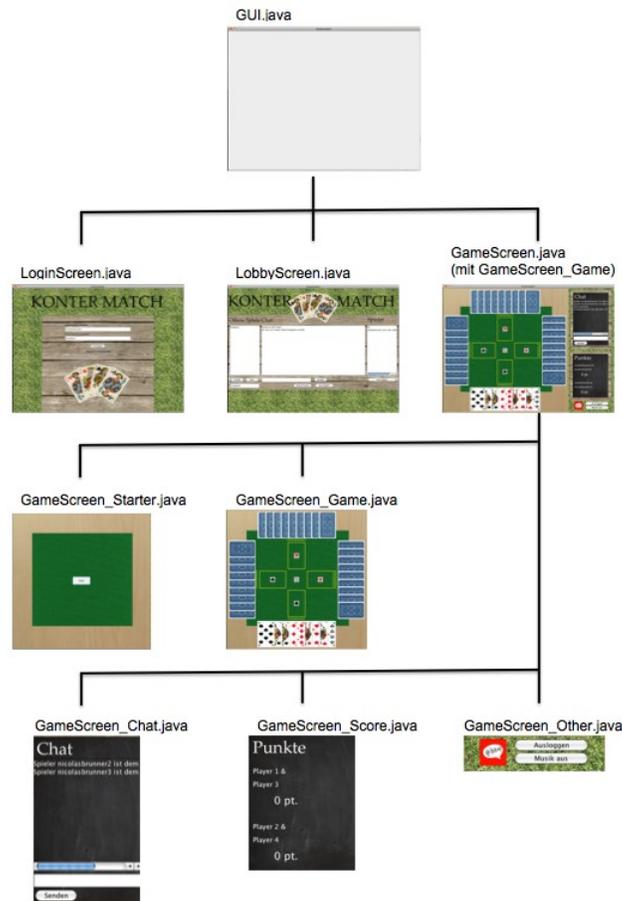


Abbildung 5: Spieltisch Bildschirm

5.4 Starten und Benutzen des Client/Server-Programms

- Server starten:
Java -jar KonterMatch.jar server 4444
- Client starten:
Java -jar KonterMatch.jar client;serverip;:4444

5.5 Software-Qualitätsmanagement

Zur Qualitätssicherung haben wir uns entschlossen, die Methode `setWinner()` in der Klasse `CardGame.java` mittels JUnit-Test zu prüfen. Dazu haben wir die Methode `setWinner()` so angepasst, um alle möglichen 4-er Kartenpaare zu simulieren und anschliessend wird die Siegerkarte wie in der Methode `setWinner()` bestimmt. Die angepasste Methode von `setWinner()` heisst `testSetWinner()` und ist im Package `test` zu finden. Beim durchführen des JUnit-Tests zeigt sich, dass die Methode `testSetWinner()` immer die korrekte Karte als Sieger bestimmt.

6 Schlusswort

Übung macht den Meister!

Weniger Geschwätz, mehr Spiel!

Auf dem Tisch müssen sie sterben!

Ein guter Jasser macht immer den letzten Stich!

Gut gewiesen ist halb gejasst, gut geschrieben ist halb gewonnen!

Hate the Game, not the Player!