

DQI06

pyKer

Pflichtenheft und Programmplanung

von Niklas Pech, Jasper Schulz, Maik Bagola und Dirk Stalhut

16.11.2009

Das Pflichtenheft enthält eine Funktionsspezifikation, die voraussichtlichen Anforderungen des Programms, einen Meilensteinplans sowie die Aufgabenverteilung innerhalb der Gruppe. Im Anhang befinden sich zudem UML-Diagramme zur Programmplanung.

INHALTSVERZEICHNIS

Produktübersicht	1
Programmspezifikation	1
Server	1
Client	5
Begriffserklärungen	6
Allgemeine Begriffe	6
Pokerbegriffe	6
Anforderungen	7
Softwareanforderungen	7
Hardwareanforderungen:	7
Meilensteinplan	7
Aufgabenverteilung	7
UML-Diagramm – Netzwerkfunktionalität	8
UML-Diagramm – Spielengine	9
Komponentendiagramm	10

PRODUKTÜBERSICHT

PyKer ist eine Pokersimulation, die über das Netzwerk mit zwei bis zehn Personen gespielt wird. Der erste Spieler macht dabei den Server (in Form einer *Lobby*) auf, fungiert als Administrator und kann Spieleinstellungen vornehmen. Im Spiel soll unter anderem ein Chat integriert sein. Spieler, die teilnehmen, können sich mit Nicknamen und Passwort registrieren oder als *Anonymus* nur mit einem Nicknamen mitspielen. Registrierte Benutzer sollen in einer Datenbank gespeichert werden und haben mehr Funktionen zur Verfügung als normale Mitspieler.

Es wird die verbreitetste Pokervariante *Texas Hol'dem* unterstützt. Da bei dieser Variante meist der Modus *No Limit* gespielt wird, ist dieser ebenfalls implementiert. Die *Blinds* (und ggf. *Ante*) sowie der Faktor und der Zeitraum, in dem sie sich erhöhen, können vom Administrator vor Beginn des Spiels eingestellt werden.

PROGRAMMSPEZIFIKATION

SERVER

NETZWERK

Die Netzwerkklassse übernimmt sämtliche Netzwerkübertragungen, Authentifizierung und die Organisation der *Lobby*.

➤ REGISTRIERUNG

Organisiert die Registrierung. Die Anmeldung kann entweder Anonym (temporär) oder mit Nickname und Passwort erfolgen. Bei dieser sind dann zusätzliche Funktionalitäten verfügbar. Die registrierten Benutzer werden in einer Datenbank gespeichert.

❖ ANONYM

Bei der anonymen Anmeldung wird kein Passwort verlangt. Der Nickname (sofern noch nicht vorhanden) kann beliebig gewählt werden und wird nach Ausloggen wieder gelöscht.

❖ MIT NICKNAME U. PASSWORT

Für erweiterte Funktionalität (z. B. wiedereinsteigen ins laufende Spiel nach einem einem disconnect) muss ein Nickname mit Passwort eingegeben werden, um sich zu authentifizieren.

○ DATENBANK

Die registrierten Nutzer werden in einer Datenbank gespeichert. Diese enthält den Nicknamen, das verschlüsselte Passwort und eine Spieler-ID.

➤ AUTHENTIFIZIERUNG

Sobald sich ein Client auf den Server connected, muss sich dieser entweder anonym oder registriert authentifizieren.

➤ LOBBY STARTEN

Der erste Spieler, startet den Server und eröffnet darauf eine *Lobby*. In dieser können sich dann andere Spieler einloggen. Der Ersteller fungiert als Administrator (Admin) und kann entscheiden, wann das Spiel startet.

❖ AUF SPIELER WARTEN

Während die *Lobby* geöffnet ist, können sich bis zu 9 weitere Spieler verbinden. Es wird solange gewartet, bis der Ersteller das Spiel startet.

❖ SPIELEINSTELLUNGEN VORNEHMEN

Der Admin kann verschiedene Spieleinstellungen vornehmen, wie den Anfangs*stack*, den Start*blind* und die Steigerungsrate der *Blinds*.

- ❖ **SPIEL STARTEN**

Das Spiel wird mit allen bisher anwesenden Spielern und den vorgenommenen Einstellungen gestartet.

- **SPIELSTATUS ÜBERTRAGEN**

Während des laufenden Spiels muss der Server ständig die aktuellen Informationen über die Spieler, sowie die Spielinformationen (z. B. *Blinds*, *Pot*, etc.) an alle Spieler übertragen.

ENGINE

Die Spielengine enthält die Logik des Spiels. Hierbei werden die Spielregeln umgesetzt sowie der Spielablauf geregelt.

- **SPIEL STARTEN**

Beim Starten des Spiels werden an alle Spieler ihre *Stacks* verteilt. Die Spieler werden zufällig am Tisch verteilt und zuletzt wird der *Dealer* bestimmt.

- ❖ **STACKS VERTEILEN**

Jeder Spieler bekommt den festgelegten *Stack*.

- ❖ **SPIELER AM TISCH VERTEILEN**

Die Spieler werden in zufälliger Reihenfolge am Tisch verteilt.

- ❖ **DEALER BESTIMMEN**

Der *Dealer* wird zufällig an einen der Spieler ausgegeben.

- **HAND STARTEN**

Zu Beginn jeder *Hand* wird das *Deck* gemischt und anschließend die Handkarten an die Spieler verteilt.

- ❖ **KARTEN MISCHEN**

Das *Deck* wird gemischt. (Nur Anzeige)

- ❖ **HANDKARTEN VERTEILEN**

Jeder Spieler bekommt zwei Handkarten vom *Deck*.

- ❖ **TISCHKARTEN AUSLEGEN**

In der entsprechenden Spielphase wird zu erst *Flop* und später entsprechend den Regeln *Turn* und *River* für alle sichtbar ausgelegt.

- **FLOP**

Der *Flop* wird ausgelegt.

- **TURN**

Der *Turn* wird ausgelegt.

- **RIVER**

Der *River* wird ausgelegt.

- **HAND AUSWERTEN**

Am Ende einer Hand werden die Handkarten aller noch teilnehmenden Spieler ausgewertet und der/die Gewinner bestimmt.

- ❖ **GEWINNE VERTEILEN**

Der/die ermittelten Gewinner bekommen entsprechend ihrem Einsatz die entsprechende Anzahl Chips aus dem *Pot* gutgeschrieben. Diese werden ihrem *Stack* hinzugefügt.

- **POT VERWALTEN**

Diese Klasse soll sämtliche Funktionalitäten, die den *Pot* betreffen, übernehmen. Hierzu gehört unter anderem das Starten einer *Runde*, die aus mehreren Setzrunden besteht. Am Anfang werden die *Blinds* festgelegt und müssen regelmäßig erhöht werden. Zudem wird hier entschieden, wer zu welchem Zeitpunkt wie viel setzen kann. Der *Stack* aller Spieler sowie der aktuelle *Pot* und ggf. der/die *Sidepot(s)* werden hier gespeichert und verwaltet. Zuletzt muss/müssen noch der/die Sieger den jeweiligen Gewinn gutgeschrieben bekommen.

- ❖ **NEUE SETZRUNDE**

Es wird eine neue Setzrunde begonnen, sie endet spätestens nachdem alle Spieler geschoben, mitgegangen oder gepasst haben. Dann wird eine neue *Tischkarte* angefordert und eine neue Setzrunde begonnen.

- ❖ **SETZMÖGLICHKEITEN ERMITTELN**
Jedem Spieler wird angezeigt, welche Möglichkeiten er zurzeit hat. Dies wird anhand seines *Stacks*, des *Pots* und des Einsatzes der anderen Spieler ermittelt.
 - **FOLD**
Passen kann jeder Spieler, sobald er an der Reihe ist mit Setzen. Während dieser Hand ist er vom aktiven Spielgeschehen ausgeschieden und sieht nur noch zu. Bei der nächsten Hand kann er wieder mitspielen.
 - **CHECK**
Schieben kann ein Spieler, falls sein Einsatz bereits gleich dem aktuellen Einsatz ist.
 - **CALL**
Mitgehen kann ein Spieler, wenn der aktuelle Einsatz höher ist als sein eigener. In diesem Fall setzt er die entsprechende Differenz ein.
 - **BET**
Setzen kann ein Spieler, falls in dieser Setzrunde noch kein Einsatz gemacht wurde. Dem entsprechend kann aufgrund der *Blinds* in der ersten Setzrunde noch nicht geboten werden. Der Mindesteinsatz ist gleich dem aktuellen *Big Blind*.
 - **RAISE**
Erhöhen kann ein Spieler, sobald er an der Reihe ist und schon gesetzt wurde. Es darf in einer *Runde* maximal dreimal erhöht werden.
- ❖ **EINSÄTZE JEDES SPIELERS VERWALTEN**
Zu jedem Spieler wird gespeichert, welche Einsätze er in dieser *Runde* getätigt hat.
- ❖ **SIDEPOT ERSTELLEN**
Sobald ein Spieler alles eingesetzt hat, wird ein *Sidepot* für ihn angelegt.
- ❖ **ENDE DER SETZRUNDE**
Am Ende der Setzrunde werden die Einsätze der Spieler dem *Pot* und ggf. dem/den *Sidepot(s)* hinzugefügt. Anschließend wird der *Dealerbutton* weitergegeben.
 - **EINSÄTZE DEM POT ZUFÜHREN**
Die getätigten Einsätze werden dem *Pot* bzw. dem entsprechenden *Sidepot* hinzugefügt.
 - **DEALER WEITERGEBEN**
Der *Dealerbutton* wird an den nächsten Spieler weitergegeben.
- ❖ **BLINDS U. ANTE VERWALTEN**
Der *Small-*, der *Big Blind* und der *Ante* werden verwaltet. Das heißt die aktuellen Werte der *Blinds* werden gespeichert und regelmäßig im Rahmen der festgelegten Einstellungen erhöht. Der *Ante* wird ab einem bestimmten Zeitpunkt eingesetzt und ab dann ebenfalls regelmäßig erhöht.
 - **BLINDS U. ANTE REGELMÄßIG ERHÖHEN**
Anhand des vorher festgelegten Zeitraums und Faktors werden die Werte der *Blinds* und *Ante* erhöht.
- ❖ **GEWINNE AUSSCHÜTTEN**
Anhand der vorher bestimmten Verteilung des *Pots* und der *Sidepots* werden die entsprechenden Summen der *Stacks* der entsprechenden Spieler gutgeschrieben.
- **SPIELERVERWALTUNG**
In der Spielerverwaltung werden alle Informationen über alle Spieler verwaltet. Dazu gehören der jeweils aktuelle Status, *Stack* und Einsatz sowie die aktuelle *Hand* aller Spieler. Zudem ist die Spielerverwaltung dafür zuständig, zu regeln, welcher Spieler wann welche dieser Informationen abfragen kann/darf.
- ❖ **SPIELERSTATUS**
Der Spielerstatus stellt den aktuellen Zustand der Spieler dar. Der Status könnte sein, dass er die aktuelle *Hand* gepasst hat oder eventuell sogar langfristiger abwesend ist. Andernfalls ist der Status normal.

- **FOLDED**
Der Spieler hat die aktuelle *Hand* bereits gepasst und kann daher bei folgenden Setzrunden übersprungen werden.
- **ALLIN**
Der Spieler hat seinen gesamten *Stack* eingesetzt und braucht daher bei den nächsten Setzrunden nicht mehr beachtet zu werden.
- **ABWESEND (IMMER FOLDEN)**
Der Spieler ist dauerhaft abwesend. Das heißt bis der Status wieder geändert wird, wird jede *Hand* sofort automatisch sofort gepasst.
- **BUSTED**
Der Spieler hat keine *Chips* mehr und kann daher nicht mehr mitspielen. Er bekommt also auch keine Handkarten mehr. er kann erst wieder mitspielen, wenn sich der Status wieder ändert, also ein neues Spiel begonnen wird.
- **NORMAL**
Der Spielerstatus Normal ist der Standardwert. Nur Spieler mit diesem Status können in der aktuellen Setzrunde noch setzen.
- ❖ **SPIELERHAND**
Hier werden die aktuellen Handkarten der Spieler gespeichert und den entsprechenden Spielern zur Verfügung gestellt.
- ❖ **SPIELERSTACK**
Die Guthaben aller Spieler (also die Anzahl der *Chips*) werden auf dem Server gespeichert. Diese können alle Spieler zu jeder Zeit abfragen.
- ❖ **SPIELEREINSATZ**
Hier wird der aktuelle Einsatz des jeweiligen Spielers gespeichert.
- **CLIENTS ANWEISEN**
Der Server weist regelmäßig die Clients an und teilt unter anderem mit, was die Spieler für Möglichkeiten haben und wie groß der *Pot* grade ist. Zudem wird allen Spielern alle Informationen über die anderen Spieler mitgeteilt. Dies sorgt dafür, dass alle Spieler zu jeder Zeit alle wichtigen Informationen über die Mitspieler besitzen.
- ❖ **SETZMÖGLICHKEITEN ÜBERTRAGEN**
Der Server überträgt an den aktuell aktiven Spieler, welche Möglichkeiten er beim Setzen hat. Unter Anderem auch die entsprechenden Mindesteinsätze beim Erhöhen bzw. Bieten.
- ❖ **GRÖÖE VOM POT ÜBERTRAGEN**
Jedem Spieler wird bei jeder Änderung des *Pots* dessen Größe und die Größe jeglicher *Sitepots* mitgeteilt.
- ❖ **STATUSWERTE ALLER SPIELER ÜBERTRAGEN**
Jedem Spieler wird bei einer Änderung eines der Statuswerte eines Spielers die jeweilige Änderung mitgeteilt.
 - **SPIELERSTATUS**
Der aktuelle Status eines Spielers.
 - **SPIELERHAND**
Die aktuellen Handkarten werden nur dem Client auf dem der entsprechende Spieler spielt zur Verfügung gestellt.
 - **SPIELERSTACK**
Der aktuelle *Stack* des Spielers.
 - **SPIELEREINSATZ**
Der Einsatz, den der Spieler diese *Hand* bisher getätigt hat.

CLIENT

NETZWERK

Die Netzwerkklassse koordiniert und organisiert die gesamte Kommunikation zwischen dem Server und dem Client. Hierzu gehören unter anderem auch die Registrierung bzw. Authentifizierung.

➤ CLIENT REGISTRIEREN

Der Client muss sich bei einem Server registrieren, um an einem Spiel teilnehmen zu können.

➤ USER ANMELDEN

Sobald sich der Client mit einem Server verbunden hat, muss er "seinen" User beim Server registrieren. Dies kann entweder anonym oder mit Nicknamen und Passwort geschehen.

❖ ANONYM

Bei der anonymen Anmeldung wird lediglich der gewünschte Nickname übertragen und nach der Bestätigung seitens des Servers kann der Spieler teilnehmen.

❖ MIT NICKNAME U. PASSWORT

Bei der Anmeldung von einem registrierten Benutzer muss dieser seinen Nicknamen und sein Passwort eingeben. Diese werden dann verschlüsselt an den Server übertragen.

USERINTERFACE

Die Userinterfaceklasse ist für das Anzeigen aller Informationen auf dem Client zuständig.

➤ INFORMATIONEN EMPFANGEN

Der Client empfängt alle benötigten Informationen vom Server. Hierzu gehören die eigenen Handkarten, die *Tischkarten*, die Statuswerte aller Spieler sowie der *Pot*. Außerdem bekommt der Client vom Server die Informationen, wann der Spieler welche Möglichkeiten beim Setzen hat mitgeteilt. Ebenso werden die Aktionen aller anderen Spieler übertragen.

❖ EIGENE HAND

Die Handkarten des Spielers, der auf dem Client angemeldet ist.

❖ TISCHKARTEN

Die *Tischkarten* sind die Karten (*Flop*, *Turn* und *River*), die für alle Spieler sichtbar sind.

❖ ALLE SPIELERSTATUSWERTE

Alle öffentlichen Informationen über die anderen Spieler werden jedem Spieler zur Verfügung gestellt. Hierzu gehören unter anderem der Spielerstatus, der *Stack* und die aktuellen Einsätze aller Spieler.

○ SPIELERSTATUS

Der Spielerstatus ist ein allgemeiner Status, der zu jedem Spieler gespeichert ist.

○ SPIELERSTACK

Der *Spielerstack* enthält die Informationen, welcher Spieler wie viele *Chips* hat.

○ SPIELEREINSÄTZE

Dies sind die Einsätze die ein Spieler bei der aktuellen *Hand* bzw. *Runde* getätigt hat.

❖ POT

Der *Pot* enthält die bei der aktuellen *Hand* getätigten Einsätze aller Spieler.

○ SIDEPOT

Falls *Sidepot*(s) existieren, werden diese ebenfalls vom Server an den Client übertragen.

❖ SETZMÖGLICHKEITEN

Der Client bekommt vom Server vorgegeben, welche Möglichkeiten der Spieler beim Setzen hat.

- ❖ **SPIELERAKTION AN SERVER**
Sobald ein Spieler eine Aktion tätigt (also etwas setzt, passt, etc.) bekommen die Clients dies vom Server mitgeteilt.
- **INFORMATIONEN ANZEIGEN**
Es werden alle Informationen die dem Client zur Verfügung stehen angezeigt.

BEGRIFFSERKLÄRUNGEN

ALLGEMEINE BEGRIFFE

Anonymus - Ein Spieler, der zur Authentifizierung lediglich einen Nicknamen benutzt.

Lobby - Eingangsbereich, in dem sich vor dem Spielstart treffen und auf den Spielstart warten.

pyKer - Name des netzwerkbasiereten Poker-Programms.

POKERBEGRIFFE

All in - Wenn ein Spieler alle Chips eingesetzt hat, nennt man dies All in.

Ante - Festgelegte Anzahl Chips, die jeder Spieler zu Beginn jeder Hand setzen muss.

Blind - Festgelegte Anzahl Chips, die bestimmte Spieler zu Beginn jeder Hand setzen müssen.

- **Smallblind** - Halber Bigblind

- **Bigblind** - Der Bigblind legt zusätzlich noch den Mindesteinsatz fest

Bet - Ein Spieler kann, wenn noch kein Einsatz gemacht wurde, eine beliebige Anzahl (mind. Big Blind) Chips setzen. Dies ist der Befehl für diese Aktion

Busted - Ein Spieler hat keine Chips mehr und ist damit aus dem aktuellen Spiel ausgeschieden.

Call – Befehl, wenn ein Spieler mit dem aktuellen Einsatz mitgeht.

Check – Befehl, wenn ein Spieler bereits den aktuellen Einsatz getätigt hat. Daher kann er schieben, das heißt er wird übersprungen.

Chip - Die Einsätze der Spieler werden in Chips getauscht.

Dealer - Der Dealer ist auf der Position vor den Blinds. Beim realen Spiel gibt er die Karten.

Dealerbutton - Spielstein, der die Position des Dealers anzeigt.

Deck - Der Kartenstapel, von dem die Spielkarten gezogen und verteilt werden.

Flop - Die ersten drei Tischkarten.

Fold - Befehl, wenn ein Spieler passt. Dieser gibt dann seine Karten ab und kann bei der aktuellen Hand nur noch zusehen.

Hand - Die Phase, während der die gleiche Hand gespielt wird. Nicht zu verwechseln mit einer *Runde*.

No Limit - Spielmodus, bei dem es beim Setzen keine Obergrenze gibt.

Pot - Anzahl aller Chips, die in der aktuellen Runde bereits gesetzt wurden.

- **Sidepot** - Ein Teil des Pots.

Raise - Ein Spieler erhöht den aktuellen Mindesteinsatz, in dem er noch mehr Chips setzt.

River - Die fünfte und damit letzte Tischkarte.

Runde - Zeitraum, in dem sich die Tischkarten nicht ändern.

Stack - Anzahl der Chips, die ein Spieler hat.

Texas Hol'dem - Meistgespielte Pokerversion (siehe Pokeranleitung in der Dokumentation).

Tischkarte - Eine Karte, die offen für alle Spieler auf dem Tisch liegt.

Turn - Die vierte Tischkarte.

ANFORDERUNGEN

SOFTWAREANFORDERUNGEN

Python Version 2.x

MODULE:

- sockets
- threads
- ...

HARDWAREANFORDERUNGEN:

- Handelsüblichen funktionstüchtigen Computer mit Netzwerk (bzw. Internet)
- Für die Internetfunktionalität min. 56k Modem

MEILENSTEINPLAN

- 04.09.2009 – gemeinsamer Entwurf des Pflichtenhefts
- 30.10.2009 – Fertigstellung Ausformuliertes Pflichtenheft
- 30.10.2009 – Gemeinsame Durchsicht des Pflichtenhefts
- 05.11.2009 – Fertigstellung des Petri-Netzes
- 05.11.2009 – Fertigstellung des UML-Diagramms
- 06.11.2009 – Zusammenführung der Teilergebnisse
- 06.11.2009 – Beginn mit der Programmierung
- 13.11.2009 – Abgleichung und gegenseitige Durchsicht des Planungsteils
- **17.11.2009 – Abgabe Pflichtenheft, Petri Netz, UML-Diagramm**
- 31.12.2009 – Fertigstellung einer Beta Version
 - Fertigstellung einer lauffähigen Netzwerkfunktionalität
 - Fertigstellung einer lauffähigen Engine
 - Fertigstellung eines lauffähigen Userinterfaces
- Januar 2010 – gemeinsame Fehlersuche im Programm
 - Ausführliches Testen (White-box- und Black-box-Tests)
- Februar 2010 – Fertigstellung eines ersten Release Candidates
 - Fertigstellung der endgültigen Versionen aller Komponenten
- Ca. 26.02.2009 – Abgabe und Vorstellung des Programms in der endgültigen Version

AUFGABENVERTEILUNG

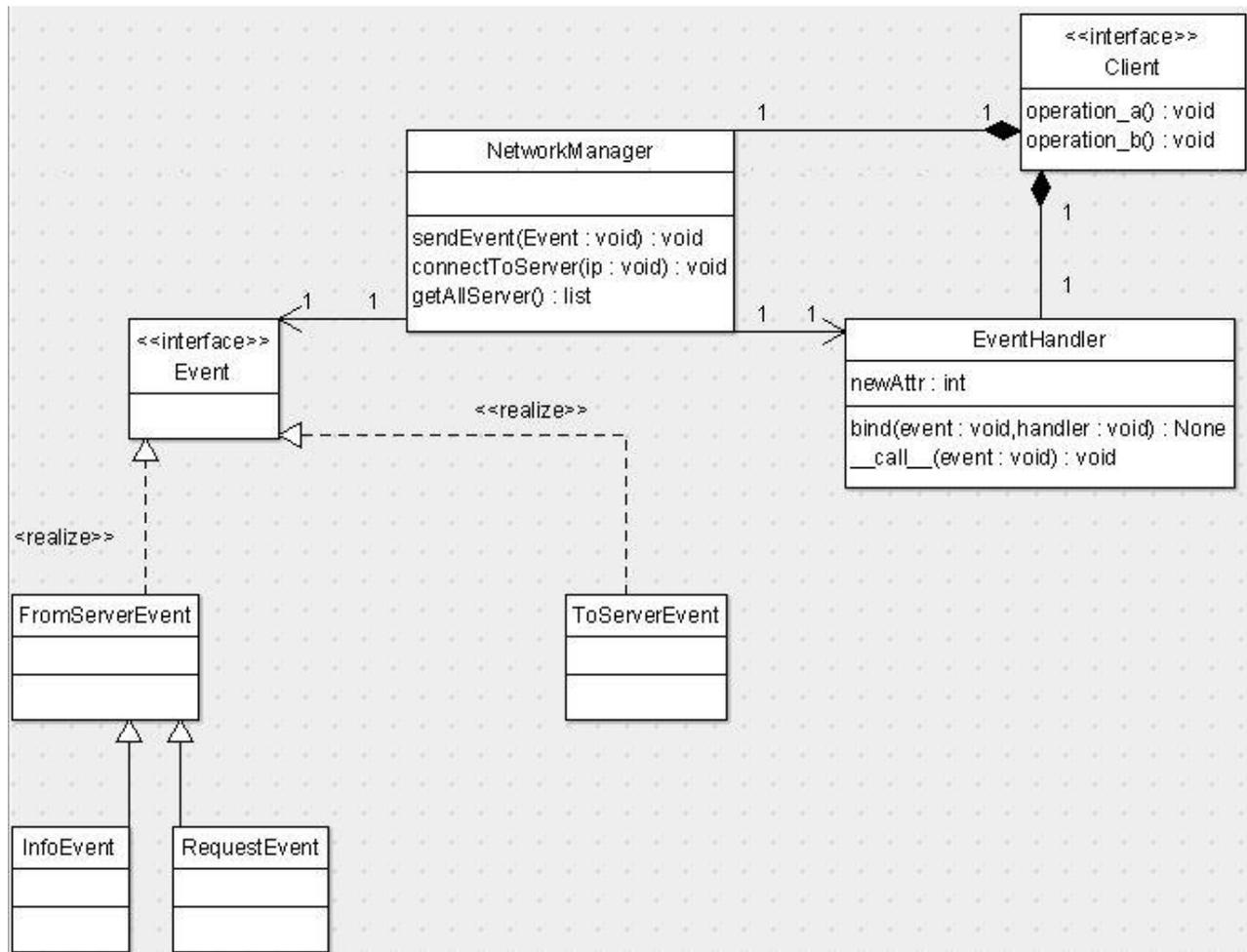
Niklas – Netzwerkfunktionalität

Maik – Spielengine

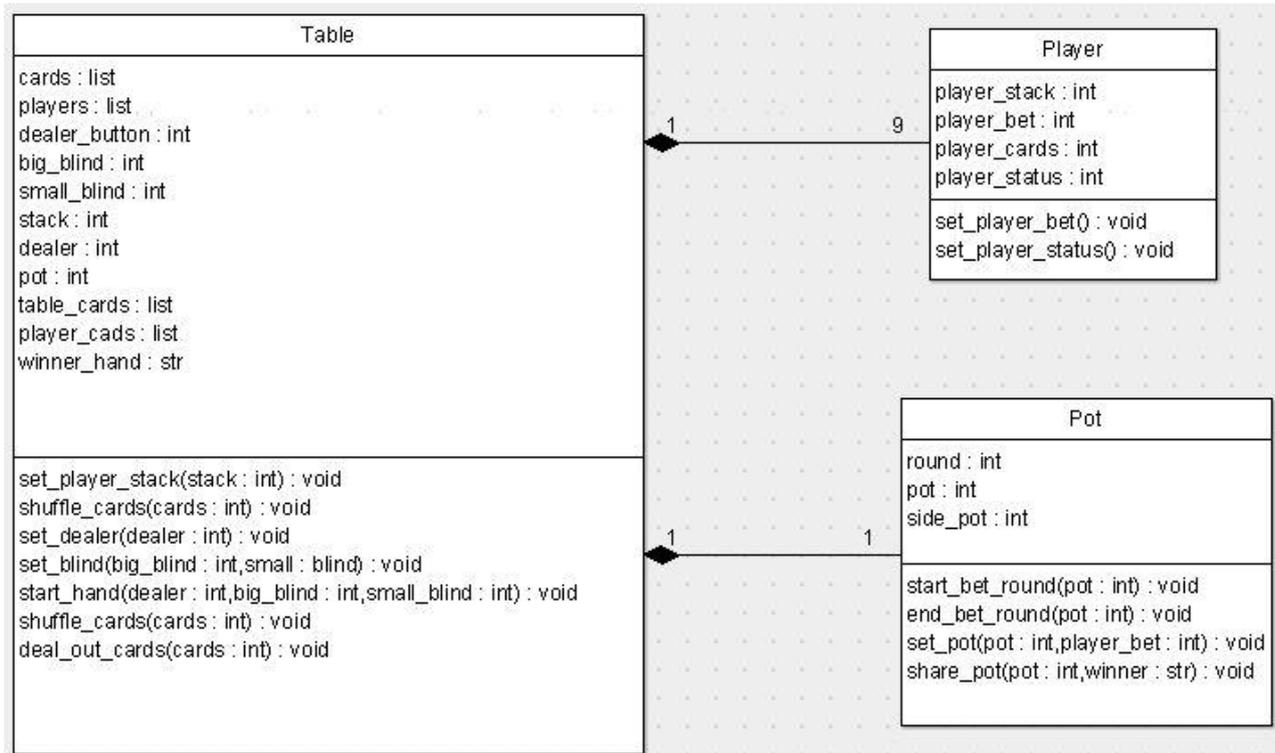
Jasper – Userinterface

Dirk – Pflichtenheft, Dokumentation

UML-DIAGRAMM - NETZWERKFUNKTIONALITÄT



UML-DIAGRAMM – SPIELENGINE



KOMPONENTENDIAGRAMM

