



Projektdokumentation Modul 326

Auftraggeber: René Probst, Daniel Fahrni

Autor*in: sullivan@bzz.ch

Klasse: I2a

Projekttitel: Tschau Sepp

Inhaltsverzeichnis

Projektbeschreibung	2
Anwendungsentwurf.....	2
Testfälle	12
Lösungsbeschreibung	15
Anhang	18

Abbildungsverzeichnis

Abbildung 1: Muster	Error! Bookmark not defined.
---------------------------	-------------------------------------

Tabellenverzeichnis

Tabelle 1: Muster Anwendungsfallspezifikation.....	4
Tabelle 2: Muster CRC-Karte.....	7



Projektbeschreibung

Dieses Projekt umfasst die Erstellung eines Tschau-Sepp Spiels in der Java-Umgebung.

Bei diesem Spiel handelt es sich um eine Jassvariante, welche Ähnlichkeiten mit Spielen wie UNO aufweist.

Die Spielregeln der hierbei verwendeten Variante sind hier zu finden:

<https://www.jassverzeichnis.ch/index.php/blog/115-tschau-sepp-jassen-lernen>

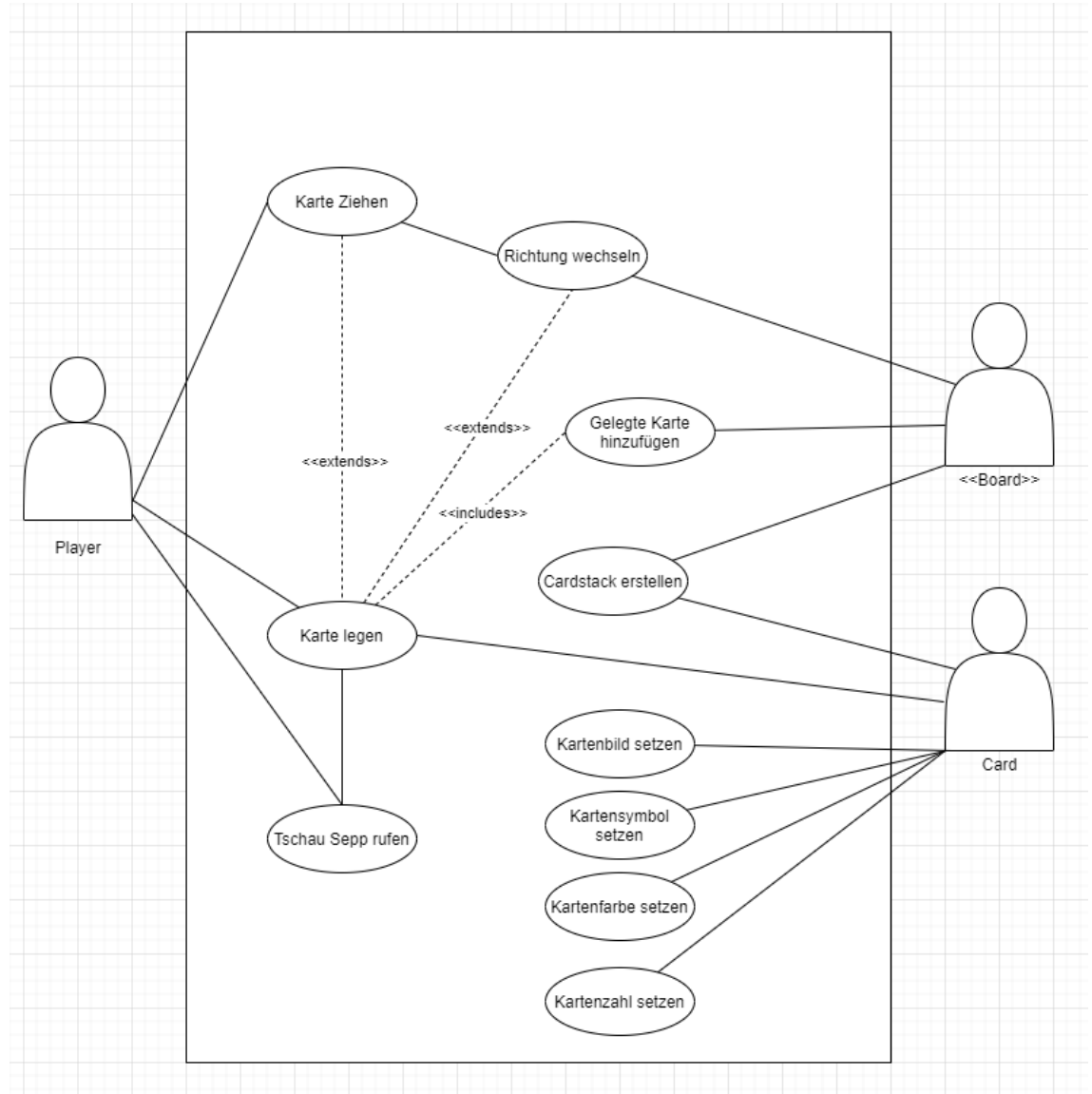
Ziele

- Das Spiel soll eine spielbare Version des Tschau-Sepp Spiels sein.
- Das Programm soll im Sinne der Spielbarkeit lauffähig und verwendbar sein.
- Das Projekt soll mit Java umgesetzt werden, um die bestmögliche Kompatibilität zwischen unterschiedlichen Systemen zu gewährleisten.
- Das Programm (und alle weiteren benötigten Dokumente) sollen spätestens bis am **4.7.** fertig und abgegeben sein (Ausnahme: Milestones, welche jeweils früher abgegeben werden).



Anwendungsentwurf

Anwendungsfalldiagramm(e)





Error! Use the Home tab to apply Titel to the text that you want to appear here.

Anwendungsfallspezifikationen

Tabelle 1: Anwendungsfallspezifikationen

ID	1.1
Name	Spielbeginn
Kurzbeschreibung	Das Spiel beginnt und die Karten werden zusammen mit dem Spielfeld erzeugt.
Akteur(e)	Card, Spielfeld
Auslöser	Spiel wird gestartet
Vorbedingung(en)	Es wurde noch kein Card-Stapel erzeugt
Ergebnis	Ein Card-Stapel wurde erzeugt
Nachbedingung(en)	keine
Ablauf	<ol style="list-style-type: none"> 1. Spiel starten 2. Card-Stapel erstellen
Alternativen	keine
Fehlerfälle	- Das Programm hat Fehler und läuft nicht.
Vorkommen	1x Pro Spiel, jeweils beim Spielbeginn

ID	1.2
Name	Spielerzug
Kurzbeschreibung	Ein Spieler ist am Zug. Er hat verschiedene Möglichkeiten für Aktionen.
Akteur(e)	Spieler, Card, System
Auslöser	<ul style="list-style-type: none"> - Das Spiel läuft - Der nächste Spieler beginnt seinen Zug
Vorbedingung(en)	Es wurde noch kein Card-Stapel erzeugt
Ergebnis	Ein Spieler hat seinen Zug beendet und hat nun weitere Aktionsmöglichkeiten
Nachbedingung(en)	keine
Ablauf	<ol style="list-style-type: none"> 1. Karte legen
Alternativen	<ul style="list-style-type: none"> - Karte ziehen - «Tschau» rufen - «Sepp» rufen
Fehlerfälle	- Ungültige Karte gelegt
Vorkommen	Regelmässig

Anforderungen

Funktional:



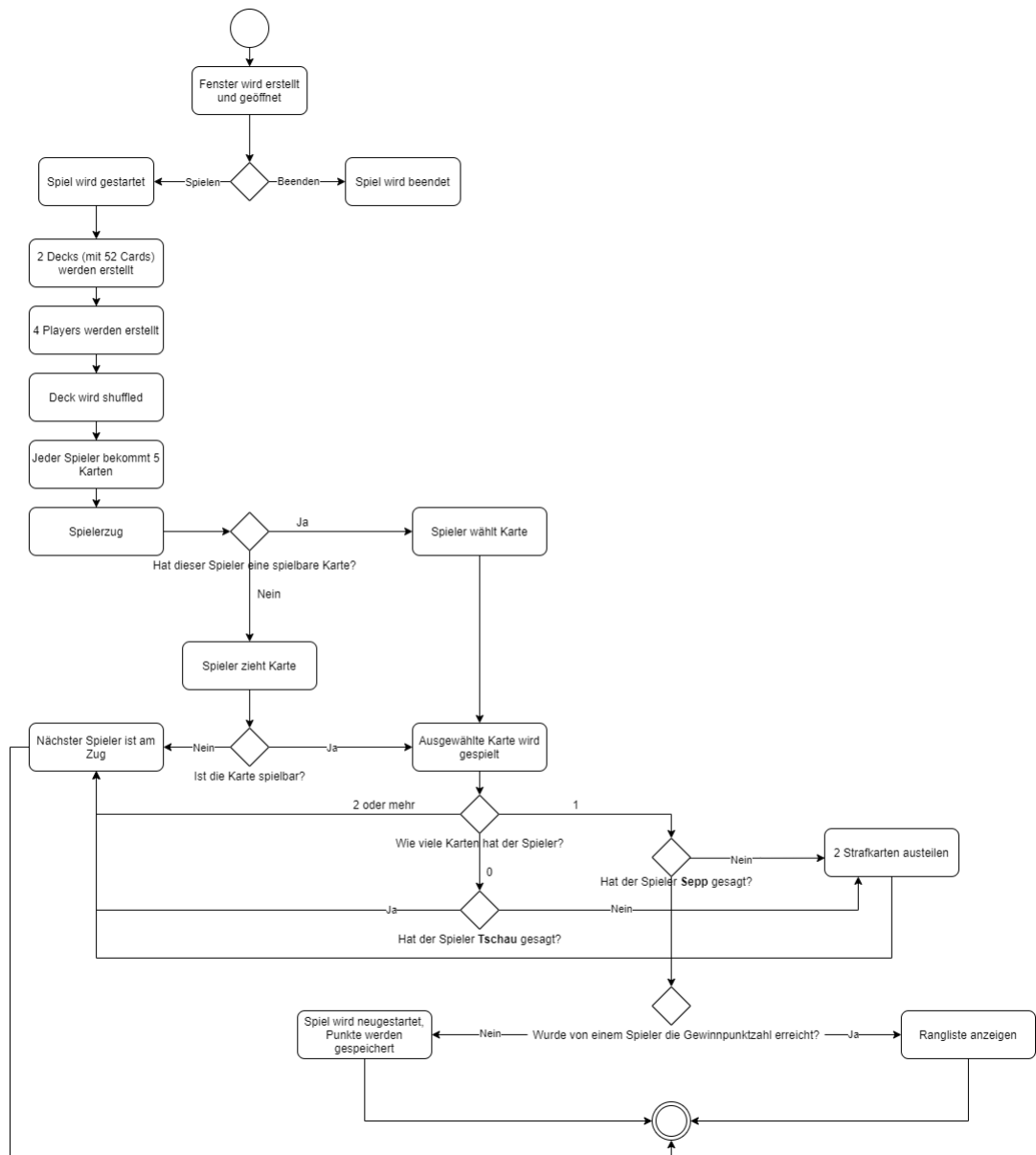
- Die offiziellen Spielregeln sind zu implementieren.
- Es werden keine Aktionskarten implementiert.
- Das Spiel wird fix für 4 Spieler ausgelegt.
- Die Punkte werden jeweils bei jedem Spieler angezeigt.

Nicht-Funktional:

- Aktionskarten sind nicht obligatorisch, können aber implementiert werden.
- Für die graphische Oberfläche wird Swing genutzt.
- Das Spiel soll mit Java umgesetzt werden.
- Das Spiel wird so ausgelegt, dass alle Spieler abwechselnd am selben Computer spielen.
- Das GUI soll maximale Benutzerfreundlichkeit durch Qualitätsmerkmale wie gut lesbare Texte und einfach bedienbare Oberflächen aufweisen.
- Das Spiel (bzw. das Programm) muss dem Benutzer erlauben, gemäss den Spielregeln zu spielen. Das heisst: Es muss erwartungsgemäss laufen und funktionieren.



Aktivitätsdiagramm





CRC-Karten

Tabelle 2: CRC-Karten

Class: Card	
Eine spielbare Karte	
Responsibilities:	Collaborators:
<ul style="list-style-type: none"> - Wert - Farbe - Bild 	<ul style="list-style-type: none"> - Player - Board

Class: Board	
Stapel mit Karten	
Responsibilities:	Collaborators:
<ul style="list-style-type: none"> - Karten enthalten - Karten kennen / setzen 	<ul style="list-style-type: none"> - Card - Player

Class: Dealer	
Verteilt und Mischt Karten	
Responsibilities:	Collaborators:
<ul style="list-style-type: none"> - Karten ausgeben - Karten mischen 	<ul style="list-style-type: none"> - Card - Board

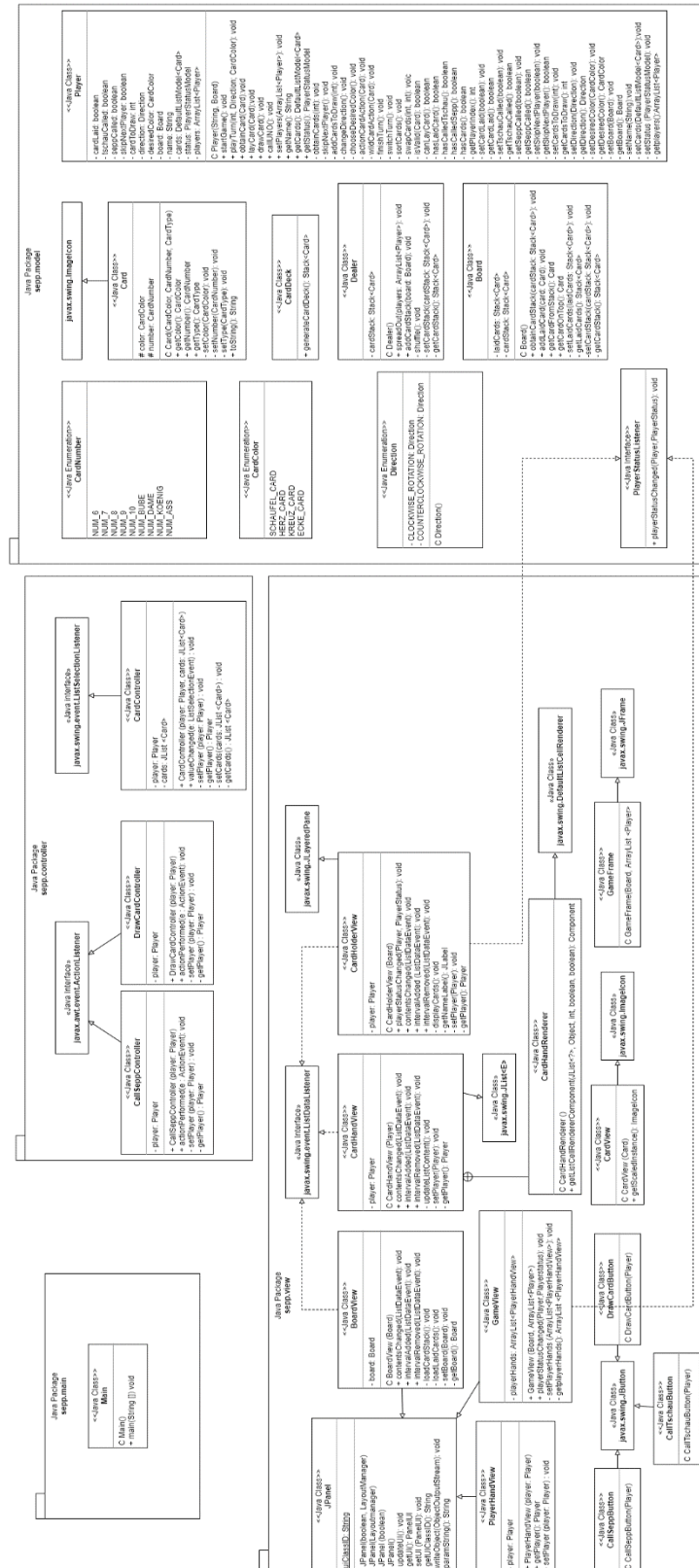
Class: Player	
Ein einzelner Spieler	
Responsibilities:	Collaborators:
<ul style="list-style-type: none"> - Tschau rufen - Sepp rufen - Punkte erhalten - Karten ziehen 	<ul style="list-style-type: none"> - CardDeck - Player

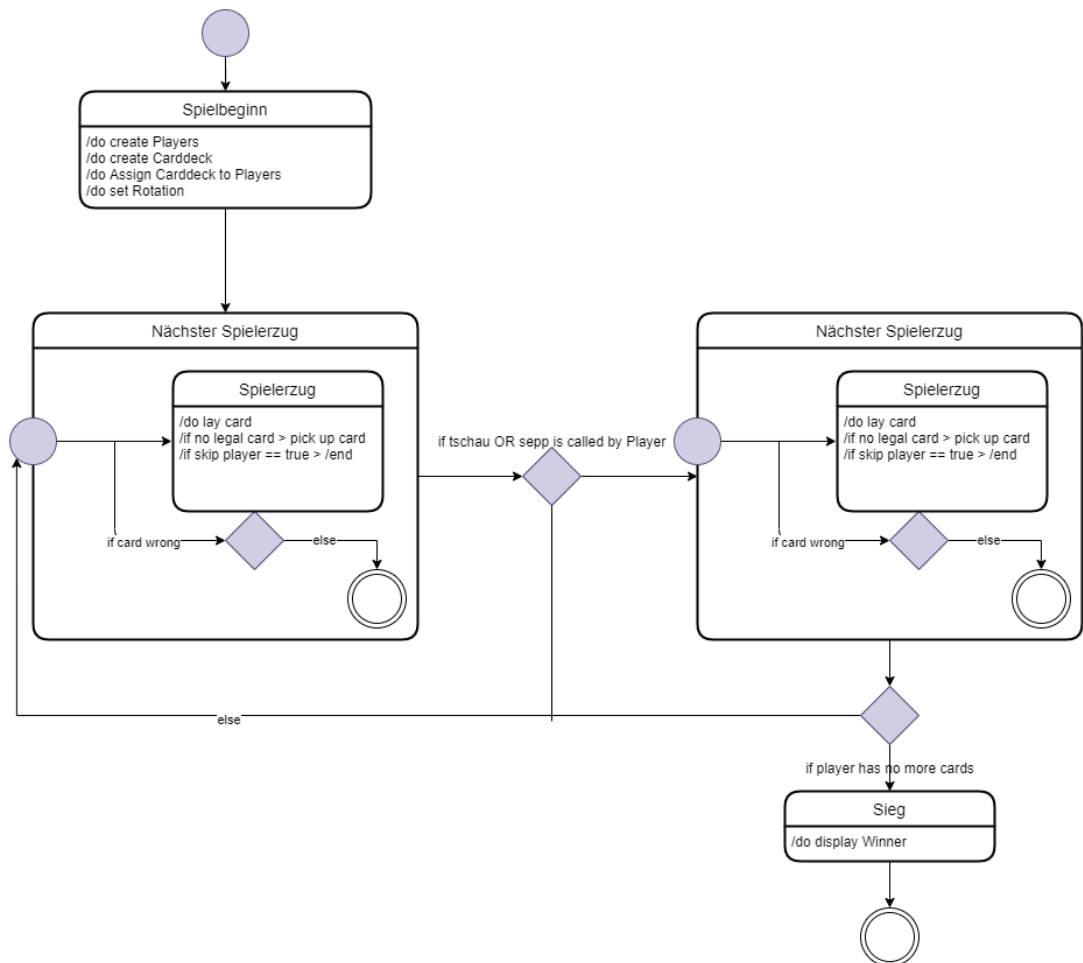


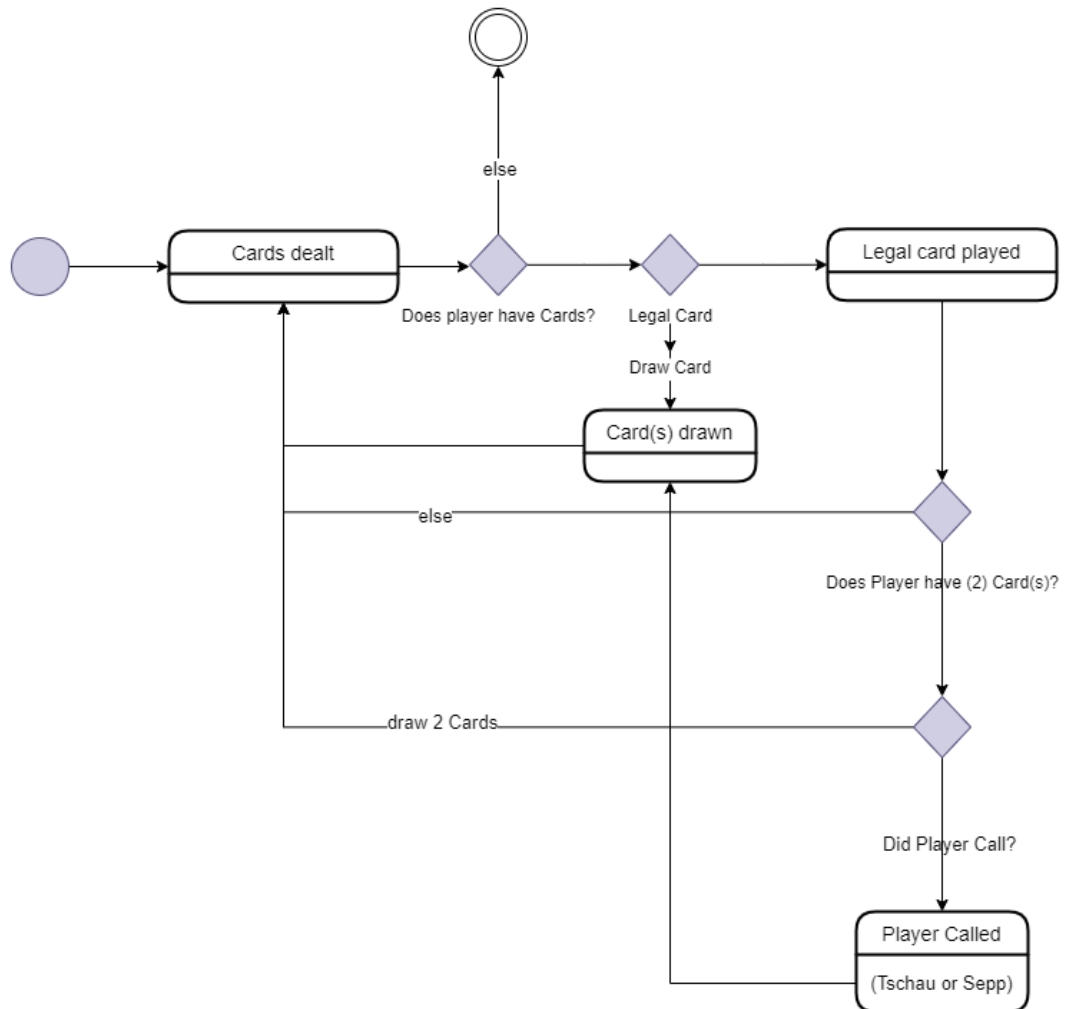
Error! Use the Home tab to apply Titel to the text that you want to appear here.

Klassendiagramm

(Auch im Anhang grösser zu finden)

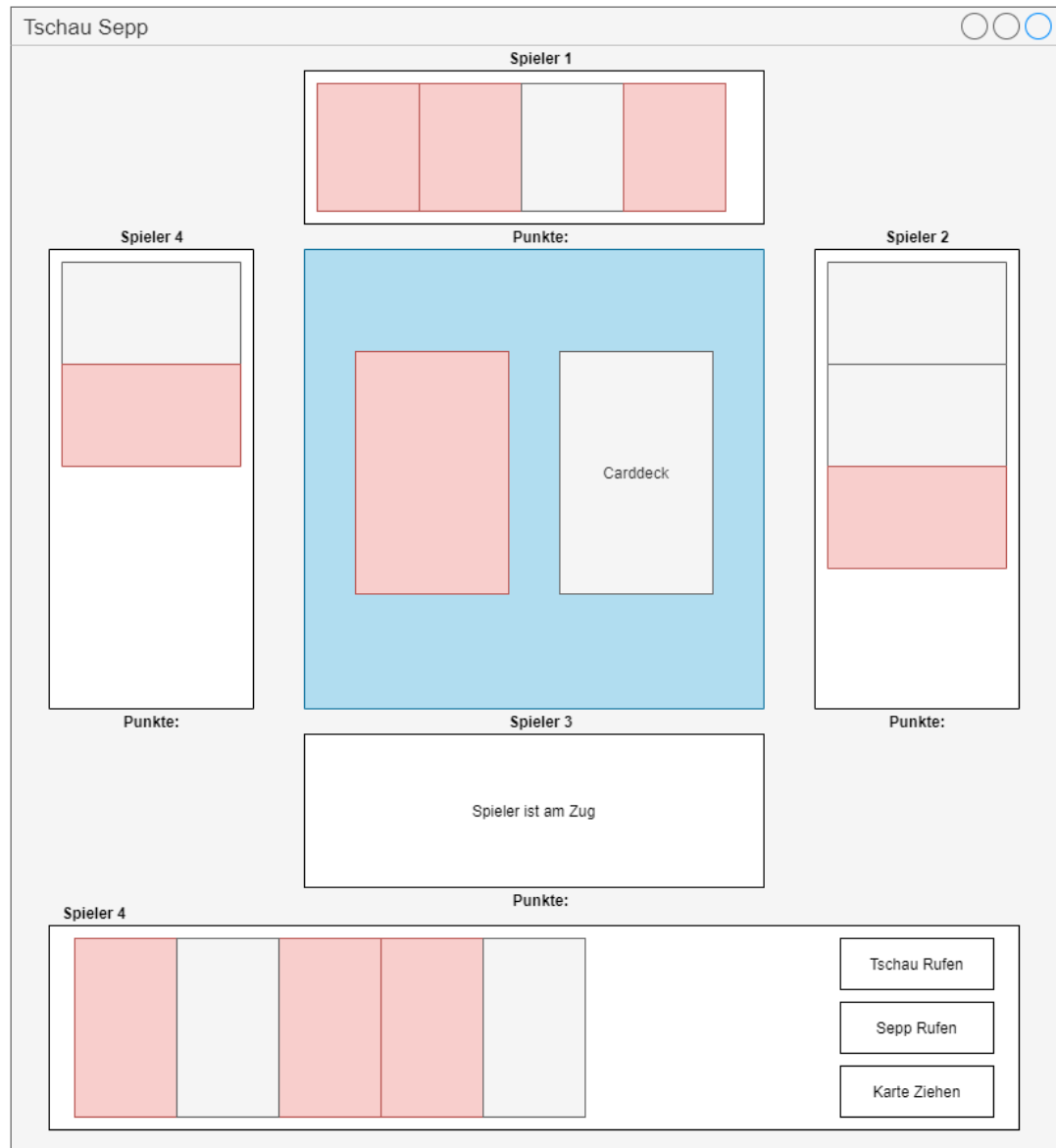


**Zustandsdiagramm**

**Sequenzdiagramm**



Mockup





Testfälle

JUnit-Testfälle

Element	Beschreibung
Testfallnummer	J-001 <i>KarteSpielen</i>
Kurzbeschreibung	Es wird getestet, ob eine legal spielbare Karte effektiv gespielt werden kann.
Testdaten	<ul style="list-style-type: none"> - Der Spieler hat eine Karte, welche laut Spielregeln in diesem Zug gespielt werden darf.
Ablauf	<ul style="list-style-type: none"> - Der Spieler legt seine Karte, indem er auf sie klickt.
Testergebnis	<ul style="list-style-type: none"> - Diejenige Karte, welcher der Spieler gerade gespielt hat, liegt nun zuoberst auf dem Stapel. - Der Spieler hat diese Karte nun nicht mehr in seiner Hand. - Der nächste Spieler ist am Zug.

Element	Beschreibung
Testfallnummer	J-002 <i>TschauVergessen</i>
Kurzbeschreibung	Es wird getestet, ob ein Spieler, welcher vergessen hat «Tschau» zu rufen, gemäss den Regeln 2 Strafkarten erteilt bekommt.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler hat noch 2 Karten übrig. - Mindestens eine davon ist spielbar.
Ablauf	<ul style="list-style-type: none"> - Der Spieler spielt eine Karte. - Er vergisst jedoch, den «Tschau» Button zu drücken.
Testergebnis	Der Spieler bekommt 2 Strafkarten erteilt.

Element	Beschreibung
Testfallnummer	J-003 <i>SeppVergessen</i>
Kurzbeschreibung	Es wird getestet, ob ein Spieler, welcher vergessen hat «Sepp» zu rufen, gemäss den Regeln 4 Strafkarten erteilt bekommt.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler hat noch 1 Karte übrig. - Diese Karte ist spielbar.
Ablauf	<ul style="list-style-type: none"> - Der Spieler spielt seine letzte Karte. - Er vergisst jedoch, den «Sepp» Button zu drücken.
Testergebnis	Der Spieler bekommt 4 Strafkarten erteilt.

Element	Beschreibung
Testfallnummer	J-004 <i>TschauErgebnis</i>



Kurzbeschreibung	Es wird getestet, ob die korrekte Benutzung des «Tschau» Buttons die richtigen Ergebnisse herbeiführt.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler hat noch 2 Karten übrig. - Mindestens eine davon ist spielbar.
Ablauf	<ul style="list-style-type: none"> - Der Spieler drückt den «Tschau» Button. - Der Spieler spielt eine Karte.
Testergebnis	Das Spiel geht weiter. Es werden keine Strafkarten erteilt.

Element	Beschreibung
Testfallnummer	J-005 <i>SeppErgebnis</i>
Kurzbeschreibung	Es wird getestet, ob die korrekte Benutzung des «Sepp» Buttons die richtigen Ergebnisse herbeiführt.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler hat noch eine Karte übrig. - Diese Karte ist spielbar.
Ablauf	<ul style="list-style-type: none"> - Der Spieler drückt den «Sepp» Button. - Der Spieler spielt seine letzte Karte.
Testergebnis	Der Gewinnerbildschirm wird angezeigt.

Element	Beschreibung
Testfallnummer	J-006 <i>SpielStart</i>
Kurzbeschreibung	Es wird getestet, ob das Spiel laut Spielregeln richtig gestartet wird.
Testdaten	<ul style="list-style-type: none"> - Das Programm ist zur Ausführung verfügbar.
Ablauf	<ul style="list-style-type: none"> - Das Programm wird ausgeführt.
Testergebnis	<ul style="list-style-type: none"> - Das GUI wird angezeigt. - Es werden 4 Spieler erstellt. - Die Karten werden geladen, verteilt und angezeigt. - Es kann nun gespielt werden.

Element	Beschreibung
Testfallnummer	J-007 <i>KartenDistribution</i>
Kurzbeschreibung	Es wird getestet, ob die Karten richtig zwischen den Spielern und dem Kartendeck distribuiert wurden.
Testdaten	Ein vollständiges Kartendeck
Ablauf	<ul style="list-style-type: none"> - Das Spiel wird gestartet - Karten werden zwischen Spielerhänden und Deck distribuiert
Testergebnis	<ul style="list-style-type: none"> - Jede Hand hat die benötigte Anzahl Karten



	- Diese Karten sind nicht mehr im Deck auffindbar.
--	--

Element	Beschreibung
Testfallnummer	J-008 <i>SpielerNichtAmZug</i>
Kurzbeschreibung	Es wird getestet, ob das Spiel erkennt, dass ein Spieler, welcher nicht am Zug ist, versucht zu Spielen.
Testdaten	Laufendes Spiel
Ablauf	<ul style="list-style-type: none"> - (Beispielsweise) Spieler 1 ist am Zug. - Spieler 2 versucht nun während diesem Zug, eine eigene Karte zu legen.
Testergebnis	Eine Fehlermeldung wird angezeigt.

Element	Beschreibung
Testfallnummer	J-009 <i>UnerlaubteKarte</i>
Kurzbeschreibung	Es wird getestet, ob das Spielen von unerlaubten Karten erkannt wird.
Testdaten	<ul style="list-style-type: none"> - Der Spieler hat mindestens eine Karte, welche momentan unspielbar ist.
Ablauf	<ul style="list-style-type: none"> - Der Spieler versucht, diese unerlaubte Karte zu spielen.
Testergebnis	Eine Fehlermeldung wird angezeigt.

Element	Beschreibung
Testfallnummer	J-010 <i>UnerlaubtesKartenziehen</i>
Kurzbeschreibung	Es wird getestet, ob das Spiel erkennt, wenn ein Spieler grundlos versucht eine Karte vom Kartendeck zu ziehen.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler ist am Zug. - Dieser Spieler hat eine spielbare Karte.
Ablauf	<ul style="list-style-type: none"> - Der Spieler versucht trotzdem, eine Karte vom Stapel aufzuheben.
Testergebnis	Es wird eine Fehlermeldung angezeigt.



Manuelle Testfälle

Element	Beschreibung
Testfallnummer	M-001 <i>IllegaleKarte</i>
Kurzbeschreibung	Es wird getestet, ob das Programm erkennt, dass eine illegale Karte gelegt wird.
Testdaten	<ul style="list-style-type: none"> - Der Spieler hat keine Karte in der Hand, welche er legen könnte.
Ablauf	<ul style="list-style-type: none"> - Der Spieler versucht, eine unerlaubte Karte zu legen, indem er auf sie klickt.
Testergebnis	Eine Fehlermeldung wird angezeigt.

Element	Beschreibung
Testfallnummer	M-002 <i>UnnoetigTschau</i>
Kurzbeschreibung	Es wird getestet, ob das Programm erkennt, dass unnötigerweise «Tschau» gerufen wurde.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler, welcher mehr als 2 Karten in der Hand hat, ist am Zug.
Ablauf	<ul style="list-style-type: none"> - Der Spieler betätigt den «Tschau» Button.
Testergebnis	Der Spieler bekommt 2 Strafkarten.

Element	Beschreibung
Testfallnummer	J-003 <i>UnnoetigSepp</i>
Kurzbeschreibung	Es wird getestet, ob das Programm erkennt, dass unnötigerweise «Sepp» gerufen wurde.
Testdaten	<ul style="list-style-type: none"> - Ein Spieler, welcher mehr als eine Karte in der Hand hat, ist am Zug.
Ablauf	<ul style="list-style-type: none"> - Der Spieler betätigt den «Sepp» Button.
Testergebnis	Der Spieler bekommt 4 Strafkarten.

Element	Beschreibung
Testfallnummer	J-004 <i>GraphicsDisplay</i>
Kurzbeschreibung	Es wird getestet, ob die Karten und dessen Grafiken richtig angezeigt werden.
Testdaten	Die Bilder für die Anzeige der Karten sind verfügbar.
Ablauf	<ul style="list-style-type: none"> - Das Fenster und das GUI wird erstellt.



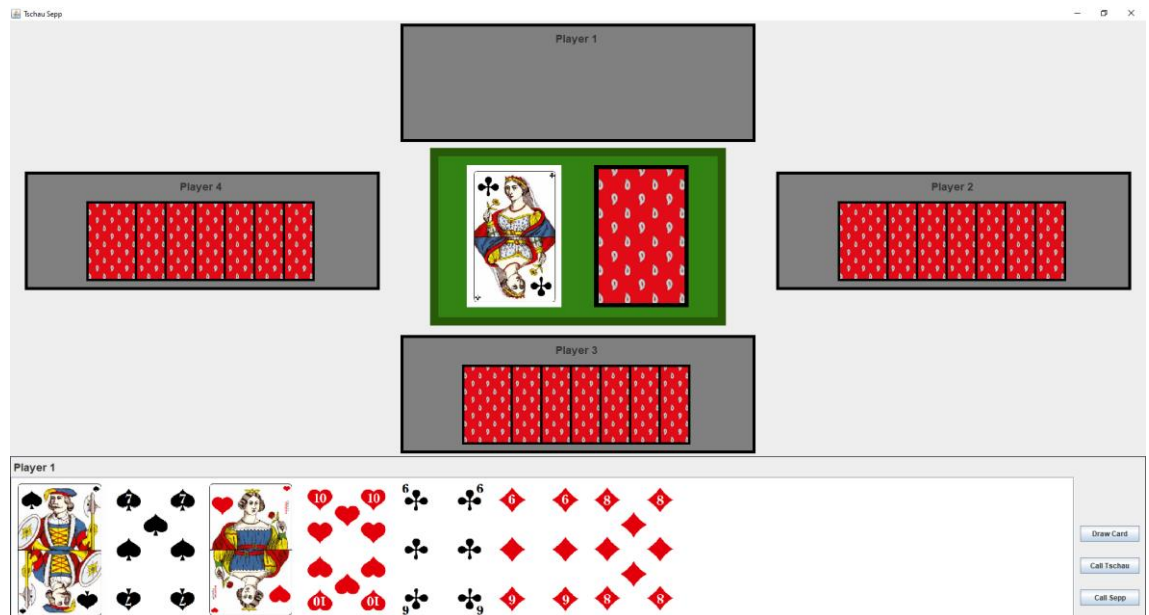
	- Die Karten und die darauf angezeigten Bilder werden richtig angezeigt. (Das heisst: Korrektheit der Ausrichtung, der Skalierung/Grösse, etc.)
Testergebnis	Die Karten werden korrekt und erwartungsgemäss angezeigt.

Element	Beschreibung
Testfallnummer	M-005 GUIDisplay
Kurzbeschreibung	Es wird getestet, ob das GUI korrekt lädt und angezeigt wird.
Testdaten	- Das Programm kann soweit ausgeführt werden, dass das GUI angezeigt wird.
Ablauf	- Der Benutzer startet das Programm
Testergebnis	Das GUI wird fehlerfrei und erwartungsgemäss angezeigt.



Error! Use the Home tab to apply Titel to the text that you want to appear here.

Lösungsbeschreibung



Die Lösung sieht momentan so aus, wie oben aufgezeigt. Sie ist lauffähig, auch wenn die Spielregeln noch nicht ganz korrekt implementiert worden sind.

Design

Die Karten wurden von der Seite Jasskarten.com entnommen.
[\(https://www.jasskarten.com/kartenspiele/schweizer-jasskarten/\)](https://www.jasskarten.com/kartenspiele/schweizer-jasskarten/)
 Die Rückseite der Karten ist das Muster eines Glarner-Tuechli.
https://de.wikipedia.org/wiki/Glarner_T%C3%BCechli

Unterschiede SOLL und IST Zustand:

Das Spiel entspricht noch nicht komplett den Spielregeln und ist daher noch nicht ganz fertig, da es während der Entwicklung noch zur Entstehung verschiedener Bugs kam, welche die Fehlerfreiheit beeinträchtigen. Zudem gibt es noch Abweichungen in dem verwendeten Kartenset, da dieses exakt einem traditionellen französischen Jasskarten-Set entspricht, statt der vorgegebenen Variante.

Das GUI wurde dem Soll-Zustand und des Mockups entsprechend umgesetzt.

Verbesserungen

Zu diesem Zeitpunkt müssten noch die Spielregeln fertig implementiert werden und kleinere Bugs und Fehler behoben werden. Zudem müsste das Kartenset den Anforderungen entsprechend angepasst werden. Aufgrund des unerwarteten Krankheitsfall, der sich lange hinauszog und einer unglücklichen Last-Minute Planung, blieb mir jedoch nicht genug Zeit.



Anhang

Im Anhang finden sie:

- Testprotokolle
- API-Dokumentation (Javadoc)
- Das Benutzerhandbuch
- Programm (runnable Jar-File)
- Source Code
- Diagramme (In voller Auflösung)

Diese Anhänge sind alle erneut im Moodle unter der finalen Abgabe zu finden.

Deklaration Eigenleistung:

Clarissa Sullivan:

- Erstellung der Diagramme
- Erstellung der Mockups
- Planung
- Dokumentation
- Umsetzung des Programms
- API