



FACHARBEIT- APPLIKATIONENTWICKLUNG

Autoren: Sharushan Atputharasa
Yanick Schlatter
Joel Eugster
Yannick Meyner
Raffael Meier

Inhaltsverzeichnis

1. Teil 1 (Obligatorische Kapitel)	5
1.1 Versionsverlauf	5
1.2 Aufgabestellung	6
1.2.1 Titel der Arbeit.....	6
1.2.2 Ausgangslage	6
1.2.3 Detaillierte Aufgabenstellung.....	6
1.3 Vorkenntnisse	7
1.4 Vorbereitungen	7
1.5 Neue Lerninhalte.....	7
1.6 Projektorganisation	8
1.7 Organisation der Arbeitsergebnisse	8
1.8 Zeitplan	9
1.9 Arbeitsjournal	10
2. Teil 2 (Projekt-Dokumentation).....	28
2.1 Management Summary.....	28
2.2 Informieren.....	29
2.2.1 Projektmanagement-Methode.....	29
2.2.2 Individuelle Bewertungskriterien	29
2.2.3 Mittel und Methoden.....	29
2.2.4 Anforderungsanalyse.....	30
2.2.4.1 Anforderungsanalyse Lernende	30
2.2.4.2 Anforderungsanalyse Schule	30
2.2.4.3 Anforderungsanalyse Programm.....	31
2.2.4.4 Anforderungsanalyse Desktop Casino-Software	32
2.2.4.5 Anforderungsanalyse Baccara	34
2.2.4.6 Anforderungsanalyse Yatzy	36
2.2.4.7 Anforderungsanalyse Blackjack.....	38
2.2.4.8 Anforderungsanalyse Roulette.....	41
2.2.4.9 Anforderungsanalyse Bingo	43
2.2.5 ISO 9241-110 Kriterien	45
2.3 Planen	48
2.3.1 Erste Idee.....	48
2.3.2 Sitemap-Navigation	48
2.3.3 Use Case	49

2.3.3.1 Use Case Casino Royale	49
2.3.3.2 Use Case Baccara.....	50
2.3.3.3 Use Case Yatzy.....	51
2.3.3.4 Use Case Blackjack.....	52
2.3.3.5 Use Case Roulette.....	53
2.3.3.6 Use Case Bingo	54
2.3.3.7 Use Case Administrator.....	55
2.3.4 Szenario	56
2.3.4.1 Szenario Casino Royale.....	56
2.3.4.2 Szenario Baccara.....	57
2.3.4.3 Szenario Yatzy.....	58
2.3.4.4 Szenario Blackjack	59
2.3.4.5 Szenario Roulette	60
2.3.4.6 Szenario Bingo	61
2.3.5 Klassendiagramm	62
2.3.5.1 Klassendiagramm Casino Royale	62
2.3.5.2 Klassendiagramm Baccara.....	63
2.3.5.3 Klassendiagramm Yatzy	64
2.3.5.4 Klassendiagramm Blackjack.....	65
2.3.5.5 Klassendiagramm Roulette.....	66
2.3.5.6 Klassendiagramm Bingo	67
2.3.6 GUI-Design.....	68
2.3.6.1 Design Login	68
2.3.6.2 Design Home	69
2.3.6.3 Design Shop	70
2.3.6.4 Design Spielauswahl	71
2.3.6.1 Design Baccara	72
2.3.6.2 Design Yatzy.....	73
2.3.6.3 Design Blackjack	74
2.3.6.4 Design Roulette	75
2.3.6.5 Design Bingo	76
2.3.7 Namenskonventionen	77
2.3.8 Datenbank	78
2.3.8.1 Vorgehen	78
2.3.8.2 Systemgrenzen / Schnittstellen zur Aussenwelt	78
2.3.8.3 Programmablaufplan mit Datenbank.....	79

2.3.8.4 Konzeptionelles Datenbankmodell	80
2.3.8.5 Logisches Datenbankmodell.....	81
2.3.8.6 Sicherheitsanalyse	82
2.3.9 Testfallspezifikation.....	83
2.3.9.1 Testfallspezifikation Lernende	83
2.3.9.2 Testfallspezifikation Schule	85
2.3.9.3 Testfallspezifikation Programm.....	87
2.3.9.4 Testfallspezifikation Desktop Casino-Software	91
2.3.9.5 Testfallspezifikation Baccara	97
2.3.9.6 Testfallspezifikation Yatzy	111
2.3.9.7 Testfallspezifikation Blackjack	126
2.3.9.8 Testfallspezifikation Roulette.....	145
2.3.9.9 Testfallspezifikation Bingo.....	157
2.4 Entscheiden	169
2.4.1 Vorgehen für GUI Entscheidung.....	169
2.4.2 Entscheidung des Designs	170
2.4.2.1 Entscheidung Login.....	170
2.4.2.2 Entscheidung Home	170
2.4.2.3 Entscheidung Shop	170
2.4.2.4 Entscheidung Spielauswahl	171
2.4.2.5 Entscheidung Baccara.....	171
2.4.2.6 Entscheidung Yatzy.....	171
2.4.2.7 Entscheidung Blackjack	171
2.4.2.8 Entscheidung Roulette	172
2.4.2.9 Entscheidung Bingo	172
2.4.3 Nutzwertanalyse.....	173
2.4.3.1 Entscheidung Branches	173
2.4.3.2 Entscheidung Pattern	174
2.5 Realisieren	175
2.5.1 Kurzfassung.....	175
2.5.1 Implementation.....	175
2.5.2.1 Implementation Login	175
2.5.2.2 Implementation Datenbank	178
2.5.2.3 Implementation Shop.....	179
2.5.2.4 Implementation Baccara	181
2.5.2.5 Implementation Yatzy	184

2.5.2.6 Implementation Blackjack	187
2.5.2.7 Implementation Roulette	194
2.5.2.9 Implementation Statistik	201
2.6 Kontrollieren	202
2.6.1 Testprotokoll	202
2.6.1.1 Testprotokoll Lernende	202
2.6.1.2 Testprotokoll Schule	202
2.6.1.3 Testprotokoll Programm	203
2.6.1.4 Testprotokoll Casino Royale	204
2.6.1.5 Testprotokoll Baccara	205
2.6.1.6 Testprotokoll Yatzy	207
2.6.1.7 Testprotokoll Blackjack	209
2.6.1.8 Testprotokoll Roulette	211
2.6.1.9 Testprotokoll Bingo	213
2.6.3 Unitest / Testszenario	215
2.7 Auswerten	216
2.7.1 Kurzbericht IPA	216
2.7.1.1 Gruppen Reflexion	216
2.7.1.2 Fazit	217
2.7.1.3 Leistungsbereitschaft	217
Abbildungsverzeichnis	218
Quellenverzeichnis	220
Glossar	220

1. Teil 1 (Obligatorische Kapitel)

1.1 Versionsverlauf

Name	Vorname	Beschreibung	Status	Datum	Version
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	04.04.2018	1.0
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	04.04.2018	1.1
Eugster	Joel	Dokumentation Casino	in Bearbeitung	05.04.2018	1.1
Meyner	Yannick	Dokumentation Casino	in Bearbeitung	23.04.2018	1.1
Schlatter	Yanick	Dokumentation Casino	in Bearbeitung	24.04.2018	1.1
Schlatter	Yanick	Dokumentation Casino	in Bearbeitung	26.04.2018	1.1
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	30.04.2018	1.2
Eugster	Joel	Dokumentation Casino	in Bearbeitung	27.04.2018	1.2
Meier	Rafael	Dokumentation Casino	in Bearbeitung	30.04.2018	1.2
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	30.04.2018	1.2
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	30.04.2018	1.2
Schlatter	Yanick	Dokumentation Casino	in Bearbeitung	01.05.2018	1.2
Atputharasa	Sharushan	Dokumentation Casino	in Bearbeitung	02.05.2018	1.2

1.2 Aufgabestellung

1.2.1 Titel der Arbeit

Realisierung einer Casino-Software mit Spiel- und Statistikfunktion

1.2.2 Ausgangslage

Sie sind Berufslernende(r) im Ausbildungsgang Informatikmittelschule und absolvieren Ihre Praxisphase im vierten Lehrjahr beim IT-Dienstleister *BBB*. Ihre Arbeit umfasst das Durchführen von unterschiedlichen Softwareentwicklungs-Projekten im Verlauf der Anstellung. Für das anliegende Projekt wünscht sich der Auftraggeber *Casino Baden* im Rahmen ihrer «Digital-Offensive» ein «Proof of Concept» für ein digitales Casino. Sie gehen die Arbeit mit Ihrem fünfköpfigen Projektteam an.

1.2.3 Detaillierte Aufgabenstellung

Die Projektteams erhalten den Auftrag, eine Desktop Casino-Software mit Spiel und Statistikfunktion inklusive grafischer Oberfläche in der Programmiersprache Java zu realisieren. Der Nutzer kann sich registrieren (Name/Passwort) und als angemeldeter Nutzer aus fünf unterschiedlichen Spielen auswählen. Der Nutzer muss nach der Anmeldung einen frei gewählten Betrag in Jetons umwandeln, die dieser für den Einsatz in den Spielen benötigt. Der Spieler kann nach der Auswahl des Spiels dieses spielen, nach persönlichem Abschluss zur Auswahl zurückkehren und sich neu entscheiden oder aufhören. Die Einsätze erfolgen in Abhängigkeit der jeweiligen Spielregeln der Spiele.

Für einen Administrator besteht die Möglichkeit, dass dieser nach erfolgreicher Anmeldung Kennzahlen einsehen kann. Pro Spieler sollen die Werte zu Einsatz, Gewinn und Verlust dokumentiert werden. Die ermittelten Werte der Spiele sollen ebenfalls dokumentiert und dargestellt werden.

Die Umsetzung eines Spiels erfolgt in Einzelarbeit durch den Lernenden, die Realisierung der übergeordneten Casino-Software inklusive Administrationsfunktion erfolgt in Gruppenarbeit durch das Projektteam. Eine Auswahl an Spielen wird durch den Auftraggeber bereitgestellt und ist die Grundlage für die Wahl des Spiels durch den Lernenden. Pro Casino-Software darf ein Spiel nur einmal vorkommen. Die Anforderungen an die einzelnen Spiele sind separat aufgeführt.

1.3 Vorkenntnisse

Datenbank mit Access:	Modul 100
IPERKA:	Modul 101
Datenbank mit SQL:	Modul 105
JavaFXML:	Modul 120
Java ohne Vererbung:	Modul 226 A
Java mit Vererbung:	Modul 226 B

1.4 Vorbereitungen

Bevor wir den Auftrag erhielten, haben wir keine speziellen Vorbereitungen für das Projekt gemacht. Aber durch unsere Vorkenntnisse waren wir schon mehr oder weniger auf das Projekt vorbereitet.

Natürlich haben wir uns nach dem Erhalten von dem Projekt vorbereitet, durch das Einteilen unserer Teilaufgaben anhand des Zeitplans. Wir haben unsere Ziele festgesetzt, damit wir immer auf unserem Pfad bleiben.

Ausserdem arbeiten wir zusätzlich im Team zusammen auf einer Cloud. Somit können wir gleichzeitig an einem Dokument arbeiten. Das macht alles einfacher, da alle Zugriff auf einem Dokument haben und alle Daten im Cloud gespeichert bleiben.

1.5 Neue Lerninhalte

Datenbankprogrammierung (JDBC), MAVEN, GIT

Zusammen an einem Programm Programmieren

Git, Repository, Branches

1.6 Projektorganisation

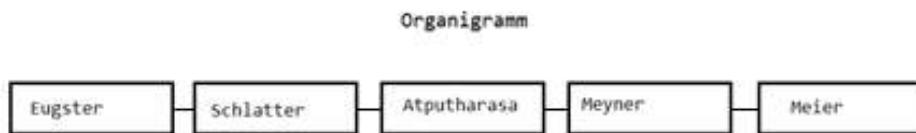


Abbildung 1: Das Team

Team:

Wir sind ein fünfköpfiges Team bestehend aus Yanick Schlatter, Yannick Meyner, Sharushan Atputharasa, Joel Eugster und Rafael Meier.

Arbeitsgeber:

Der Arbeitsgeber ist Berufsbildung Baden (BBB), welche durch mehrere Lehrer unterstützt wird. Nämlich durch Herr Fiechter, Herr Flick, Herr Weidmann, Herr Meyer, Herr Bachofner und Herr Jerg.

Spiele:

Sharushan Atputharasa: Blackjack

Yanick Schlatter: Roulette

Joel Eugster: Baccara

Yannick Meyner: Bingo

Rafael Meier: Yatzy

1.7 Organisation der Arbeitsergebnisse

Die Versionierung konnten wir mit GitLab gut umsetzen. Hier kann immer auf eine alte Version des Projekts zugreifen. Dazu muss man auf der Branch einfach eine Version zum gewünschten Zeitpunkt auswählen. So kann man auf jeden Standpunkt zu greifen. Das war gut wenn man einen unauffindbaren Fehler hatte konnte man das Programm wieder zurück setzen.

1.8 Zeitplan

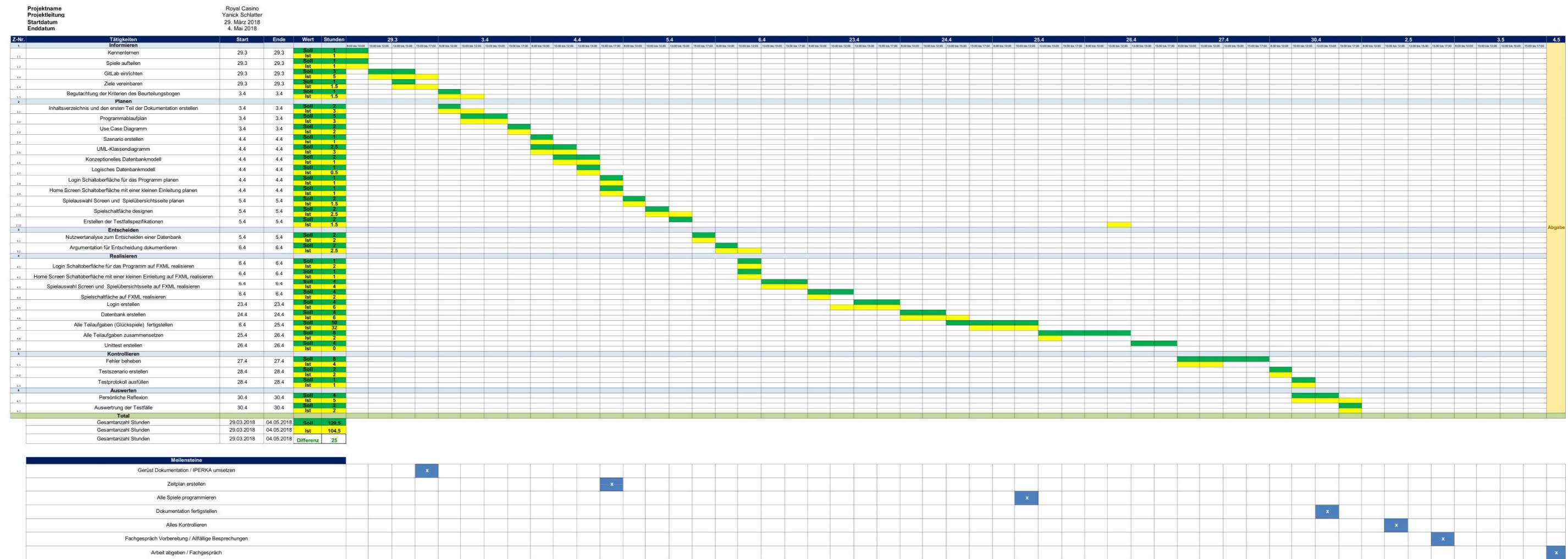


Abbildung 2: Zeitplan

Die zeitliche Planung sollte realistisch sein, es wird sicherlich ein aufwändiges Projekt. Wir sollten auch genügend Zeit haben. Von der BBB steht uns drei Wochen zur Verfügung. Dazu haben wir noch die zweiwöchigen Ferien, die wir auch benutzen können.

Die Meilensteine wurden auf dem Zeitplan definiert.

1.9 Arbeitsjournal

Datum:	Donnerstag, 29 März 2018
Checkliste Zeitplan	1.1 Kennenlernen ✓ 1.2 Spiele aufteilen ✓ 1.3 GitLab einrichten ✓ 1.4 Ziele vereinbaren ✓
Tätigkeiten:	<p>Am Donnerstag wurden wir in eine Gruppe eingeteilt. Wir haben uns zuerst kennengelernt, dann haben wir die verschiedenen Glücksspiele aufgeteilt. Dann fingen wir an uns Gedanken über das ganze Programm zu machen, wie man was machen könnte und wir haben jetzt auch eine klare Vorstellung von dem Ganzen. Wir wissen auch wie wir die Datenbank einbauen wollen. Somit haben wir die Ziele definiert.</p> <p>Danach haben wir auf unserer Dokumentation die ersten drei Stichpunkte von IPERKA umgesetzt. Währenddessen war die einte Hälfte der Gruppe mit dem Einrichten von GitLab beschäftigt. Danach haben wir uns unserem Zeitplan zugewandt.</p>
Ungeplante Tätigkeiten:	Wir haben noch einige Papierentwürfe für die GUI's erstellt.
Probleme:	Wir hatten Probleme beim Einrichten von GitLab, durch eine ungewollte einstündige Sperre. Dazu konnten die meisten der Gruppe nicht auf GitLab zugreifen da wir vergessen hatten Rechte fürs hochladen zu verteilen und die Packages richtig einzurichten.
Hilfestellungen:	<p>Die Informationen bekamen wir von der Moodleseite und deren Aufträge.</p> <p>Wir haben die Lehrpersonen Herr Meyer und Herr Bachofner um Hilfe gebeten bei der Sperre. Bei den Rechten und Packages haben uns Mitschüler geholfen.</p>
Überzeit:	-
Erfolge:	In unserem Zeitplan konnten wir die Punkte Kennenlernen, Spiele aufteilen, GitLab einrichten und Ziele vereinbaren abschließen.
Misserfolge:	Wir haben alles fertiggebracht am Ende des Tages was wir geplant haben.

Reflexion:	Uns lief alles gut bis auf die kleinen Probleme bei GitLab. Eventuell könnten wir uns ein wenig mehr Informieren für weitere Themen, wenn wir uns nicht mehr ganz sicher sind.
------------	---

Datum:	Dienstag, 3 April 2018
Checkliste Zeitplan	1.5 Begutachtung der Kriterien des Beurteilungsbogens ✓ 2.1 Inhaltsverzeichnis ✓ Erster Teil der Dokumentation erstellen ✓ 2.2 Programmablaufplan ✓ 2.3 Use Case Diagramm ✓
Tätigkeiten:	Die Gruppe hat sich intensiv mit den Kriterien des Prüfungsbogens beschäftigt. Wir haben alles gründlich durchgelesen. Danach haben wir mit der Dokumentation begonnen. Wir haben das Layout bestimmt und ein Inhaltsverzeichnis angelegt. Dann hat jeder ein PAP erstellt, wenn er das Gefühl hatte eine schwierige Stelle in seinem Programm zu haben. Um 15 Uhr haben wir mit den UseCase Diagrammen angefangen. Zuerst haben wir zusammen ein Diagramm für das ganze Programm erstellt, anschliessend hat jeder eines für sich erstellt.
Ungeplante Tätigkeiten:	Wir haben das ganze Projekt auf Maven umstrukturiert, damit bei jedem, welcher das Programm ausführt die Plug-Ins automatisch installiert werden.
Probleme:	Das Durchlesen der Kriterien gestaltete sich etwas schwieriger als erwartet. Da es sehr viele sind und sie sind kompliziert geschrieben. Deshalb hat sich der ganze Zeitplan verschoben.
Hilfestellungen:	Wir haben das Dokument „Detail_Bewertungsbogen“ verwendet. Weitere Hilfestellungen haben wir von Moodle bezogen. Für Maven brauchten wir eine Lehrperson, die uns half.
Überzeit:	Sharushan hat am Abend noch die GUI's eingescannt um sie in digitaler Form zu haben. Das dauerte ca. 5 min.
Erfolge:	Heute konnte unser Team, die Phase Informieren abschliessen. Die Grundlage für das Dokument für die Dokumentation erstellt. Zudem haben wir alle benötigten UseCase Diagramme erstellt.
Misserfolge:	Die Einbindung von Maven war am Anfang schwierig, deshalb ist es ein Misserfolg. Trotzdem konnten wir den Fehler am gleichen Tag zum Glück noch beheben.
Reflexion:	Wir hätten das ganze Projekt von Anfang Maven einbinden sollen, dies hätte Zeitgespart.

Datum:	Mittwoch, 4 April 2018
Checkliste Zeitplan	2.4 Szenario erstellen ✓ 2.5 UML-Klassendiagramm ✓ 2.6 Konzeptionelles Datenbankmodell ✓ 2.7 Logisches Datenbankmodell ✓ 2.8 Login Schaltfläche für das Programm planen ✓ 2.9 Home Screen Schaltfläche mit einer kleinen Einleitung planen ✓
Tätigkeiten:	Ganz am Anfang des Tages haben wir Szenarios für unsere Spiele erstellt. Danach hat jeder zu seinem Spiel ein UML gezeichnet und anschliessend haben wir sie in der Gruppe besprochen. Nach dem Besprechen haben wir die Datenbank geplant. Dafür haben wir ein Konzeptionelles und Logisches Datenbankmodell gezeichnet. Um 13:15 Uhr hatten wir das erste Expertengespräch mit Herren Fiechter. Wir haben mit ihm zum ersten Mal unsere Dokumentation besprochen. Er hat uns erste Tipps gegeben, auf was wir achten sollten. Dann haben wir begonnen unsere Login Oberfläche zu Planen. Wir haben uns über folgendes Gedanken gemacht. Im Login sollte der Benutzer sich anmelden können und einen Account erstellen können. Zusätzlich sollte er ein paar Infos zum Casino oder zu den möglichen Spielen erhalten. Am Ende des Tages haben wir noch an einer Willkommens-Maske, mit einer kleinen Einleitung, geplant.
Ungeplante Tätigkeiten:	Das Planen der Datenbank hat etwas länger gedauert, da wir uns zu dem Thema, nochmals gründlich informieren mussten.
Probleme:	Wir hatten Probleme mit unserer Arbeits-Cloud. Wir hatten keine Rechte mehr um Dokumente zu löschen oder neue Dokumente zu erstellen. Yanick Schlatter konnte dies aber nach einer halben Stunde wieder beheben.
Hilfestellungen:	S. Fiechter hat uns geholfen und geschaut, wo wir stehen, was wir verbessern könnten und ging mit uns über die Verschiedenen Leitfragen.
Überzeit:	Yanick Schlatter hat das Home Screen GUI zuhause nochmals überarbeitet, da es noch fehlerhaft wurde.
Erfolge:	Wir hatten unser erstes Expertengespräch.

Wir haben erste GUI's geplant.	
Misserfolge:	Der Misserfolg heute war, dass wir zu lange für die Modelle gebraucht haben. Dafür müssen wir beim nächsten Mal sicherlich mehr Zeit einplanen.
Reflexion:	Die Gruppe hat heute gut gearbeitet, wir konnten Fehler schnell beheben, da wir zusammengearbeitet haben.

Datum:	Donnerstag, 5 April 2018
Checkliste Zeitplan	2.10 Spielauswahl Screen und Spielübersichtsseite planen ✓ 2.11 Spielschaltfläche designen ✓ 2.12 Erstellen der Testfallspezifikationen ✓ 3.1 Nutzwertanalyse zum Entscheiden einer Datenbank ✓
Tätigkeiten:	<p>Wir haben die Kriterien des Zeitplanes abgearbeitet. Zu Beginn des Tages haben wir an einer Spielauswahlmaske und einer Spielübersichtsmaske geplant. Auf der Spielauswahlmaske kann man die verschiedenen Spiele auswählen. Auf der Spielauswahlmaske wird der Benutzer Informationen und eine Spielanleitung für die Spiele sehen können. Ebenfalls am Morgen designte jeder eine Maske für sein Spiel. Dies nahm den ganzen Morgen in Anspruch, am Nachmittag machten wir zuerst mal eine Anforderungsanalyse. Wir wollten hier bereits die Testfälle schreiben mussten dies aber verschieben. Danach teilten wir uns in 2 Gruppen auf, die eine der Beiden machte eine Nutzwertanalyse für die Datenbank, die andere argumentierte währenddessen direkt die Entscheide für die Analyse.</p>
Ungeplante Tätigkeiten:	Uns wurde von Herr Fiechter mitgeteilt, die Anforderungen sind am besten mit einer Anforderungsanalyse zu testen. Darum mussten wir zuerst noch so eine erstellen
Probleme:	<p>Wir hatten ein Problem, dass wir noch gar keine Anforderungen hatten. Und somit mussten wir diese zuerst machen. Danach hatten wir keine Zeit und wir mussten die Testfälle verschieben. Ein paar Leute haben sich bei designen etwas länger Zeit genommen.</p>
Hilfestellungen:	Herr Fiechter hat uns geholfen. Er kam mit der Idee eine Anforderungsanalyse zu machen, da es nachher einfacher ist Testfälle zu schreiben.
Überzeit:	Zwei Leute der Gruppe brauchten für ihr Spiel Design etwas länger.
Erfolge:	<p>Wir konnten die Punkte Spielauswahl Screen und Spielübersichtsseite planen, Spielschaltfläche designen, Erstellen der Testfallspezifikationen, Nutzwertanalyse zum Entscheiden einer Datenbank und Argumentation für Entscheidungen dokumentieren nach Plan erledigen und konnten sogar noch Programmieren.</p>

Freitag, 4. Mai 2018

Misserfolge:	Zwei Leute hatten etwas länger bei den Tätigkeiten, die wir am Morgen gemacht haben. Sie konnten aber diesen kleinen Misserfolg bis am Abend wieder aufholen.
Reflexion:	Wir hätten die Anforderungen in den Zeitplan aufnehmen sollen und diese auch machen. Vor den Testfällen. Jetzt müssen wir es weit hinausschieben.

Datum:	Freitag, 6 April 2018
Checkliste Zeitplan	<p>3.2 Argumentation für Entscheidung dokumentieren ✓</p> <p>4.1 Login Schaltfläche für das Programm auf FXML realisieren ✓</p> <p>4.2 Home Screen Schaltfläche mit einer kleinen Einleitung auf FXML realisieren ✓</p> <p>4.3 Spielauswahl Screen und Spielübersichtsseite auf FXML realisieren ✓</p>
Tätigkeiten:	<p>Heute haben wir unsere Notizen, der gestrigen Diskussion in die Dokumentation übertragen. Leider haben wir dafür etwas länger gebraucht als geplant. Denn es kamen noch ein paar neue Gedanken hinzu. Danach haben wir unsere Design Ideen für das Login in FXML umgesetzt. Yannick und Joel haben das während zwei Stunden realisiert. Der Rest der Gruppe hat an einer HomeScreen Maske gearbeitet.</p> <p>Am Nachmittag hat die Gruppe an einen GameSelectScreen gearbeitet. Der ganze Tag nahm etwas mehr Zeit in Anspruch, denn wir nahmen uns die Zeit die GUI's mit einer hochwertigen Grafik Qualität auszustatten.</p>
Ungeplante Tätigkeiten:	Viele Grafiken mussten wir mit Photoshop bearbeiten.
Probleme:	Joel hatte etwa 1,5 Stunden Probleme mit seiner FXML-Datei. Nämlich gab es einen kleinen Fehler, er musste ein FXML laden und vergaß beim Pfad ein „/“.
Hilfestellungen:	<p>Riccardo Landolfo konnte Joel beim obenstehenden Problem helfen.</p> <p>Wir haben heute viel mit Photoshop Bilder bearbeitet, damit sie in unser Programm passen.</p>
Überzeit:	Wir haben bei verschiedenen Punkten etwas länger gebraucht und somit Überzeit geschaffen. Zum Glück konnten wir sie bis am Abend schon wieder abarbeiten.
Erfolge:	Wir haben die Phase Entscheiden abgeschlossen. Wir konnten ein Grafisch hochwertiges GUI umsetzen.

Misserfolge:	Ein kleinerer Misserfolg hatte, wie schon gesagt, Joel. Denn es nahm schon ein bisschen viel Zeit in Anspruch für einen so kleinen Fehler
Reflexion:	Das Team hat heute gut zusammengearbeitet und einander geholfen. Das Teamwork funktioniert entwischen sehr gut. Deswegen konnten wir die Probleme, die wir hatten gut und schnell lösen.

Datum:	Montag, 23 April 2018
Checkliste Zeitplan	4.4 Spielschaltfläche auf FXML realisieren ✓ 4.5 Login erstellen ✓
Tätigkeiten:	Heute Montag hat jeder sein eigenes Design auf FXML für sein gewähltes Spiel umgesetzt. Am Nachmittag erstellten wir das Login. Wir haben für heute etwas viel Zeit eingeplant, aber es hat sich gelohnt, denn so konnten wir das Design hochwertig umsetzen.
Ungeplante Tätigkeiten:	Yanick Schlatter kam auf die Idee ein Menu zu gestalten. Es sollte in jedem Spiel verfügbar sein.
Probleme:	Programm Probleme gab es keine. Aber das bearbeiten der Grafiken war zum Teil für unsere Fähigkeiten etwas schwierig. Wir mussten unsere Kenntnisse auffrischen und das hat auch wieder Zeit gekostet.
Hilfestellungen:	Yanick hat für sein Menu ein YouTube Video von Genuine Coder verwendet. Um unsere Photoshop Fähigkeiten aufzufrischen benutzten wir auch YouTube.
Überzeit:	Heute haben wir gut geplant zum Teil sogar zu viel Zeit eingeplant. Fast alle haben dafür schon vorher zu Hause begonnen. Somit hatten sie nicht viel zu tun in der Schule.
Erfolge:	Wir konnten ein Login realisieren, das alle Aspekte abdeckt. Wir waren alle zufrieden mit unserer Leistung und erfreuten uns über das Resultat.
Misserfolge:	Heute gab es zum Glück keine Misserfolge. Denn es waren zum Glück eher einfach Prozesse, die umgesetzt wurden.
Reflexion:	Das Team hat heute eine gute Leistung gezeigt, wir haben gut allein gearbeitet, aber auch am Nachmittag sehr gut zusammen. Das ist nicht selbstverständlich.

Datum:	Dienstag, 24 April 2018
Checkliste Zeitplan	4.6 Datenbank erstellen ✓ 4.7 Alle Teilaufgaben (Glückspiele) fertigstellen ✓
Tätigkeiten:	<p>Yannick Meyner und S. Atputharasa haben heute die Datenbank erstellt. Das stellte sich schwieriger heraus als gedacht, denn es nahm doch einige Zeit in Anspruch. Wir hatten sehr lange einen Fehler bei der Datenbank. Wir wussten nicht wo her er kam. Nach einigem Überdenken fanden wir heraus, dass wenn jemand neues das Programm startet, die Datenbank bei ihm gar nicht angelegt ist. Somit brauchten wir einen Webserver, der uns zur Verfügung gestellt wurde.</p> <p>Zwei Stunden vor Schluss konnten wir aber dann doch noch mit der Logik und dem Feinschliff der Spiele beginnen.</p>
Ungeplante Tätigkeiten:	Wir mussten die Datenbank auf einen XAMP Webserver laden, da sonst jeder eine lokale SQL Datenbank erstellen müsste. Und wenn diese lokale Datenbank nicht vorhanden wäre, könnte man sich bei unserem Programm nicht anmelden.
Probleme:	Es hat Zeit gekostet bis wir einen Fehler behoben konnten. Denn wir wussten nicht, dass die Datenbank auf einen Server muss, damit das Login richtig funktioniert.
Hilfestellungen:	Yannick Hugeler hat sich bereit erklärt uns einen Webserver zur Verfügung zu stellen. Zudem haben wir sehr viel mit Photoshop gearbeitet, für die GUI Oberfläche. Für die Verknüpfung des Programms mit der Datenbank haben uns auch YouTube Tutorials angeschaut.
Überzeit:	Yannick musste zuhause noch ein Admin für die Datenbank erstellen.
Erfolge:	Wir konnten die Datenbank erfolgreich umsetzen, dies war sicher der grösste Erfolg heute.
Misserfolge:	Wir konnten eine Datenbank aufsetzen, welche funktioniert und die auch bei jeder neu Installation des Programmes funktioniert.
Reflexion:	Der einzige, aber grösste Misserfolg heute war die Datenbank, da sie nicht funktioniert hat bei einem anderen Computer. Und wir dort etwas Zeit verbraten haben, denn wir versuchten einen anderen Lösungsweg zu erst. Dieser hat aber nicht geklappt.

Datum:	Mittwoch, 25 April 2018
Checkliste Zeitplan	4.7 Alle Teilaufgaben (Glückspiele) fertigstellen ✓ 4.8 Alle Teilaufgaben zusammensetzen ✓
Tätigkeiten:	<p>Heute Morgen konnten wir bereits mit unseren Spielen abschliessen. Wir haben dafür viel zu viel Zeit eingeplant. Hat aber damit zu tun, dass die ganze Gruppe schon einen grossen Teil in den Ferien erledigt hat. Trotzdem hätten nach unserem Ermessen ca. 30 Stunden gereicht.</p> <p>Am Nachmittag wollten wir nach Zeitplan das ganze Projekt zusammenfügen. Diese Arbeit viel aber aus, da wir alle auf dem gleichen Branch gearbeitet haben. Somit hatten wir mehr Zeit an der Dokumentation zu arbeiten.</p>
Ungeplante Tätigkeiten:	Wir hatten hier Zeit an der Dokumentation zu arbeiten, was sehr hilfreich war. Denn wir hatten es nötig die Dokumentation zu überarbeiten und zu verbessern.
Probleme:	Es gab keine Probleme heute. Es kam einfach vor das beim Pullen, kleinere Probleme auftauchten, weil heute zum Teil mehrere Personen am gleichen Dokument gearbeitet haben. Dann musste die Person das ganze Projekt neu installieren. Es war umständlich aber nicht wirklich ein grosses Problem.
Hilfestellungen:	Wir konnten heute alles selber machen, da wir keine unbekannten Probleme hatten.
Überzeit:	Heute haben wir keine Überzeit gemacht.
Erfolge:	<p>Wir haben heute zu einem grossen Teil das Programm fertig programmiert. Alle Spiele funktionieren so wie sie sollen.</p> <p>Wir konnten das Projekt erfolgreich zusammenführen.</p>
Misserfolge:	Es gab heute keine Misserfolge, ausser das Problem beim Mergen, wenn jemand gepullt hat.
Reflexion:	Wir kamen gut voran hatten mehr als genug Zeit um an der Dokumentation arbeiten. Wir waren alle froh, dass wir etwas Zuhause gearbeitet haben. Denn wir haben nicht mehr viel Zeit.

Datum:	Donnerstag, 26 April 2018
Checkliste Zeitplan	4.8 Alle Teilaufgaben zusammensetzen ✓ 4.9 Unitest erstellen ✗
Tätigkeiten:	<p>Heute konnten wir uns gar nicht an den Zeitplan halten. Denn wir mussten sehr viel an der Dokumentation arbeiten. Wir konnten heute endlich die Testfallspezifikation erledigen. Zusätzlich haben wir viele Stellen des Programmes dokumentiert.</p> <p>Leider fanden wir keine Zeit um Unitest zu machen. Niemand wusste wie man sie genau macht. Deshalb schieben wir es heraus.</p>
Ungeplante Tätigkeiten:	Die Testfallspezifikation kamen unerwartet aber wir mussten sie heute machen.
Probleme:	Wir konnten keine Unitests machen und somit können wir unser Programm nicht testen. Wir haben das Problem noch nicht behoben.
Hilfestellungen:	Wir haben das Dokument «Wegleitung» benutzt um die Dokumentation zu verfeinern. Wir haben darauf geschaut, dass alle geforderten Kriterien in unserem Dokument erfüllt werden.
Überzeit:	Wir hatten heute keine Überzeit gemacht.
Erfolge:	Wir konnten das letzte Element der Phase Planen verspätet abschliessen. Wir kamen gut voran in der Dokumentation.
Misserfolge:	Heute gab es zum Glück keine Misserfolge.
Reflexion:	Wir haben uns heute gut aus unserer Verspätung herausgeholt, denn wir haben miteinander gearbeitet.

Datum:	Freitag, 27 April 2018
Checkliste Zeitplan	5.1 Fehler beheben ✓
Tätigkeiten:	<p>Wir haben heute noch klein Fehler im Programm behoben. Yannick hat bei der Kommunikation mit der Datenbank ein paar Fehler in der Programmlogik behoben. Der Rest der Gruppe hat kleinere Fehler in ihrem Programm. Leider mussten wir am Nachmittag mit der Präsentation weiterfahren.</p>
Ungeplante Tätigkeiten:	<p>Wir mussten heute wieder an der Dokumentation arbeiten.</p>
Probleme:	<p>Yannick hatte das Problem, dass ein «varchar» Attribut die länge 1 hat auf der Datenbank. Er wollte aber mehrere Zeichen hinzufügen. Dies konnte er heute beheben.</p>
Hilfestellungen:	<p>Wir haben für unsere Fehler auf verschiedenen Webseiten Lösungen gesucht.</p>
Überzeit:	<p>Heute haben wir keine Überzeit.</p>
Erfolge:	<p>Wir konnten heute bei unserem Programm erfolgreich die Fehler, die wir noch hatten beheben.</p>
Misserfolge:	<p>Heute gab es keine Misserfolge wir konnten die Vorgaben für heute abarbeiten.</p>
Reflexion:	<p>Wir haben heute eher selbständig gearbeitet. Aber jeder hat einen guten Job gemacht.</p>

Datum:	Montag, 30 April 2018
Checkliste Zeitplan	5.2 Testszenario erstellen X 5.3 Testprotokoll ausfüllen ✓ 6.1 Persönliche Reflexion ✓ 6.2 Auswertung der Testfälle ✓
Tätigkeiten:	Die persönliche Reflexion hatte uns schon etwas mehr Mühe gemacht, denn darüber musste wir uns zuerst einmal informieren. Darum brauchten wir dort etwas länger. Trotzdem konnten wir heute die letzte Tätigkeit auf dem Zeitplan ausführen. Denn wir haben Joel und Rafael bei der Reflexion abgesetzt und sie konnten die Auswertung machen.
Ungeplante Tätigkeiten:	Wir mussten uns über die persönliche Reflexion informieren, denn wir wussten nicht ob unser Text alles beinhaltet.
Probleme:	Wir hatten keine Probleme heute. Wir haben aber das Gefühl, dass wir hintendrein sind mit der Zeit. Denn wir müssen noch viel an der Dokumentation arbeiten.
Hilfestellungen:	Wir benutzten wieder eine Webseite um uns über die Reflexion zu informieren.
Überzeit:	Am Samstag (der 28. April) hat Sharushan die Tabellen für das Testfallprotokoll erstellt. Dann wurde das Testfallprotokolle von Yanick Schlatter, Yannick Meyner, Sharushan Atputharasa, Joel Eugster und Raffael Meier ausgefüllt.
Erfolge:	Wir wurden heute mit allen Punkten des Zeitplanes fertig. Trotzdem sind wir noch nicht fertig. Wir müssen immer noch an der Dokumentation arbeiten und daran schleifen.
Misserfolge:	Wir hatten heute keine Misserfolge.
Reflexion:	Wir haben gut gearbeitet, da wir zusammengearbeitet haben.

Datum:	Mittwoch, 2 Mai 2018
Checkliste Zeitplan	(Meilensteine) Alles Kontrollieren X
	Fachgespräch Vorbereitung X
Tätigkeiten:	Heute haben wir zuerst den ganzen Projekt Code kontrolliert. Wir haben noch kleinere Fehler gefunden, die aber schnell behoben wurden. Diese waren auch nicht grundlegend. Danach haben wir die Dokumentation überarbeitet. Hier gab es grössere Probleme mit Satzstrukturen und Grammatik. Leider langte es dann nicht mit der Vorbereitung für das Fachgespräch zu beginnen.
Ungeplante Tätigkeiten:	Heute gab es keine ungeplanten Tätigkeiten.
Probleme:	Die Gruppe hatte kleinere Probleme im Code entdeckt. Danach haben wir viele Sprachfehler in der Dokumentation gefunden. Wir haben diese aber überarbeitet und machen den Rest heute Abend.
Hilfestellungen:	Wir haben heute völlig selbstständig gearbeitet.
Überzeit:	Heute haben wir keine, dafür werden wir morgen haben.
Erfolge:	Wir konnten den Code überarbeiten. Er sollte jetzt einwandfrei funktionieren. Zudem haben wir den grössten Teil der Doku überarbeitet.
Misserfolge:	Wie haben heute gemerkt, dass einige in unserer Gruppe kein gutes Deutsch haben und wie mussten das überarbeiten.
Reflexion:	Nächstes Mal müssen wir vermehrt das Dokument vorlaufend auf Fehler überprüfen. So können wir Zeitsparen.

Datum:	Donnerstag, 3 Mai 2018
Checkliste Zeitplan	(Meilensteine) Alles Kontrollieren ✓ Fachgespräch Vorbereitung ✓
Tätigkeiten:	Leider konnten wir gestern Abend das Dokument nicht mehr ganz Überarbeiten, deshalb mussten wir die ersten 2 h des heutigen Tages für das investieren. Danach konnten wir aber für Morgen Vorbereiten. Wir haben dazu Texte erstellt, die wir vortragen. Nach dem haben wir zu dem Text Folien auf PowerPoint erstellt.
Ungeplante Tätigkeiten:	Wir mussten die Dokumentation noch auarbeiten.
Probleme:	Leider hat die eingeplante Überzeit für die Dokumentation nicht ausgereicht. Um alles zu Hause zu erledigen.
Hilfestellungen:	Keine Hilfestellung wurde für den heutigen Tag benutzt.
Überzeit:	Wie haben gestern Abend noch zwei Stunden an der Dokumentation gearbeitet.
Erfolge:	Heute konnten wir den Meilenstein von gestern noch abschliessen und somit sind wir jetzt auch mit diesen fertig.
Misserfolge:	Heute gab es zum Glück so kurz vor Schluss keine Misserfolge mehr.
Reflexion:	Heute haben wir gut zusammengearbeitet. Trotzdem sind wir alle froh, wenn der ganze Stress bald vorbei ist.

Datum:	Freitag, 4 Mai 2018
Checkliste Zeitplan	Abgabe ✓
Tätigkeiten:	Heute Morgen haben wir um 09:15 mit der Präsentation begonnen. Danach kamen die Demonstration und das Fachgespräch. Dann haben wir das Dokument mit der Dokumentation für die Abgabe vorbereitet. Wir haben das ganze Projekt anschliessend auf GitLab hochgeladen damit sind wir fertig.
Ungeplante Tätigkeiten:	Gab es heute keine.
Probleme:	Wir hatten heute keine Probleme mit dem Ganzen. Was erstaunlich ist, denn wir haben wieder einmal mit GitHub gearbeitet.
Hilfestellungen:	Heute benötigten wir keine Hilfestellung.
Überzeit:	Wir mussten heute keine Überzeit leisten.
Erfolge:	Wir sind heute endlich mit allem fertig geworden. Wir sind alle froh und wir sind sehr zufrieden. Wir führten heute das Expertengespräch mit Herrn Fiechter.
Misserfolge:	An diesem Tag gab es keine Misserfolge. Alles lief wie am Schnürchen.
Reflexion:	Super gemacht! Wir konnten alles machen und alles hat funktioniert.

2. Teil 2 (Projekt-Dokumentation)

2.1 Management Summary

Ausgangslage

Wir sind ein fünfköpfiges Programmier-Team, mit der Aufgabe innerhalb von 3 Arbeitswochen, ein Programm mit Java zu schreiben. Das Ganze wird von uns dokumentiert, damit jeder versteht was wir gemacht haben. Alle Programmierer gehen in die Informatikmittelschule, zwei aus der alten Kantonschule in Aarau, drei aus der Kantonschule Baden. Das Projekt wird aber an der BBB Baden umgesetzt. Wir fünf sollten ein vollwertiges virtuelles Casino erstellen. Dazu bekamen wir eine Variation von spielen, die es beinhalten sollte. Jeder konnte ein Spiel frei auswählen.

Vorgehen

Wir sind nach der Projektmanagement-Methode IPERKA vorgegangen. Wir haben das ganze Projekt nach diesen Schritten abgearbeitet. Zuerst haben wir alle nötigen Informationen, die wir für das Programm und die einzelnen Spiele benötigen zusammengesucht. Hier haben wir viel mit den Lehrpersonen zusammengearbeitet oder wir haben auf dem World Wide Web gesucht.

Danach haben wir, in der Phase Planen, uns Gedanken darüber gemacht, wie man das Spiel umsetzen könnte. Zum Beispiel haben wir je zwei GUI Designs erstellt, damit wir im nächsten Schritt uns für eines entscheiden können.

Erst jetzt beginnen wir mit Realisieren. Hier programmieren wir und setzen die Designs um. Wir haben uns für das Codepattern MVC entschieden, da es einfach zu bedienen ist und wir so rasch vorankommen. Das war grössten Teils Einzelarbeit für die Spiele. Trotzdem gab es einige Masken, die wir gemeinsam machen mussten.

Bei dem Schritt kontrollieren haben wir das ganze Programm auf Fehler überprüft und diese gegebenenfalls noch behoben. Der letzte Schritt war Auswerten. In dieser Phase ging es darum, eine Reflexion für die Gruppe, sowie für sich selbst zu schreiben.

Ergebnis

Das Ergebnis dieser drei Wochen ist ein funktionierendes Casino Programm. Mit den Spielen: Black Jack, Baccara, Bingo, Roulette, Yatzy. Man kann sich ein Benutzerkonto erstellen und Chips kaufen. Das Design haben wir hochwertig gehalten, wir hoffen, dass es viele Personen anspreche kann. Wir haben alles Mögliche getan um die Dokumentation so vollständig wie möglich fertig zu stellen.

Wir selber sind sehr zufrieden mit dem Endresultat.

2.2 Informieren

2.2.1 Projektmanagement-Methode

Wir haben uns bei der Projektmanagement-Methode für IPERKA entschieden, da wir IPERKA in der BBB kennengelernt haben und schon mehrmals angewendet haben. IPERKA ist für uns auch am sinnvollsten, da IPERKA sinnvoll gegliedert ist und am besten zu unserem Projekt passt. Beginnen werden wir bei unserem Projekt mit dem Informieren und über Entscheiden zum Auswerten gehen. Die Punkte von IPERKA sind wie oben erwähnt sinnvoll gegliedert, was eine grosse Unterstützung für unser Projekt bietet. Dadurch, dass wir schon Erfahrung mit IPERKA gemacht haben, wird diese Methode auch sehr umgänglich sein. Da wir eine umfangreiche Aufgabe bekommen haben, kann man durch diese Methode systematisch vorgehen, ausserdem ist mit dieser Arbeitstechnik auch leichter, gestreckte Ziele zu erreichen.



Abbildung 3: IPERKA

in Quellenverzeichnis angegeben

2.2.2 Individuelle Bewertungskriterien

- I 1) Systematik der Lösungsfindung/Lösungsvorschläge
- I 2) Benutzerfreundlichkeit über die ganze Applikation)
- I 3) Systemeinbettung, Schnittstellen (technisch)
- I 4) Codierung: Fehlerbehandlung
- I 5) Sicherheitsanalyse (Applikation)
- I 6) Codingstyle - lesbarer Code
- I 7) Gliederung des Programms

2.2.3 Mittel und Methoden

Java, JDBC, JavaFX, TestFX, JUNIT, UML, IPERKA, MAVEN, GIT

2.2.4 Anforderungsanalyse

2.2.4.1 Anforderungsanalyse Lernende

Anforderung der Lernende an das Programm

Anforderungs-Nummer	Anforderung
1.1	Das Programm sollte eine Help Funktion beinhalten.
1.2	Im Programm sollte man einen frei gewählten Betrag in Chips umwandeln.
1.3	Das Programm sollte keine Kompilierfehler beinhalten.
1.4	Das Programm sollte beim Eingeben des Passwortes, dieses nicht anzeigen
1.5	Das Programm sollte bei allen Spielen ein ausklappbares Menu haben.

2.2.4.2 Anforderungsanalyse Schule

Anforderung der Schule an das Programm

Anforderungs-Nummer	Anforderung
2.1	Das Programm sollte mit Java realisiert werden.
2.2	Im Programm sollte dir CRAP Kriterien erfüllen.
2.3	Das Programm sollte eine Anmeldefunktion für Benutzer haben.
2.4	Das Programm sollte einen frei gewählten Betrag in Jetons umwandeln können.
2.5	Das Programm sollte eine Funktion anbieten, in welcher der Administrator sich anmelden kann, um die Kennzahlen einsehen zu können.

2.2.4.3 Anforderungsanalyse Programm

Anforderungen Benutzer an das Programm:

Anforderungs-Nummer	Anforderung
3.1	Ein Benutzer kann einen Account erstellen oder sich anmelden.
3.2	Ein Benutzer kann einen frei gewählten Betrag in Chips umwandeln.
3.3	Ein Benutzer kann Bingo, Baccara, Blackjack, Roulette, Yatzy spielen.
3.4	Ein Benutzer kann sich Abmelden.
3.5	Ein Benutzer kann Chips durch das Spielen gewinnen/verlieren.

Anforderungen Administrator an das Programm:

Anforderungs-Nummer	Anforderung
4.1	Ein Administrator kann Anzahl Runden vom jeweiligen Spiel sehen.
4.2	Ein Administrator kann die E-Mail-Adresse jedes Spielers ansehen.
4.3	Ein Administrator kann den Kontostand (Chipsanzahl) jedes Spielers ansehen.
4.4	Ein Administrator kann sehen welcher Spieler am meisten Gewinn gemacht hat.
4.5	Ein Administrator kann Sieg und Niederlagen der einzelnen Spieler sehen.

2.2.4.4 Anforderungsanalyse Desktop Casino-Software

Anforderungs-Nummer	Anforderung
5.1	Das Programm muss eine Desktop Casino – Software sein.
5.2	Die fünf Spiele Blackjack, Roulette, Baccara, Bingo und Yatzy müssen in diesem Programm vorhanden sein.
5.3	Eine Statistikfunktion für den Administrator soll das Programm beinhalten.
5.4	Das Programm sollte eine grafische Spieloberfläche haben.
5.5	Das Programm sollte in der Programmiersprache Java realisiert werden.
5.6	Das Programm sollte eine Registrierfunktion haben, in welcher der Nutzer sich mit Name und Passwort registrieren kann.
5.7	Die fünf Spiele Blackjack, Roulette, Baccara, Bingo und Yatzy müssen auswählbar sein.
5.8	Der Nutzer muss nach der Anmeldung einen frei gewählten Betrag in Jetons umwandeln können.
5.9	Der Nutzer kann die Jetons in den Spielen benutzen.
5.10	Der Spieler kann nach der Auswahl des Spiels, das Spiel spielen.
5.11	Der Spieler kann nach persönlichem Abschluss des Spiels zur Spieldatenwahl zurückkehren und sich neu entscheiden oder aufhören.
5.12	Die Einsätze des Spielers erfolgen in Abhängigkeit der jeweiligen Spielregeln der Spiele.
5.15	Für einen Administrator besteht die Möglichkeit, dass dieser nach erfolgreicher Anmeldung Kennzahlen einsehen kann.
5.13	Der Administrator kann nach der Anmeldung von jedem Spieler die Werte zu Einsatz, Gewinn und Verlust anschauen können.

5.14	Die ermittelten Werte der Spiele sollen ebenfalls dokumentiert und dem Administrator dargestellt werden.
5.15	Jeder Lernender muss ein Spiel der Glückspiele selber realisiert haben.
5.16	Die Realisierung der übergeordneten Casino-Software muss in Gruppenarbeit realisiert werden.
5.17	Die Administrationsfunktion erfolgt in Gruppenarbeit durch das Projektteam.
5.18	In der Casino-Software darf ein Spiel nur einmal vorkommen.

2.2.4.5 Anforderungsanalyse Baccara

Anforderungs-Nummer	Anforderung
6.1	Das Spiel enthält eine Titelseite.
6.2	Die Titelseite enthält den Namen des Spiels.
6.3	Die Titelseite enthält den aktuellen Kontostand.
6.4	Das Spiel wird gestartet, indem ich auf einen Startbutton klicke.
6.5	Die gezogenen Karten des Spielers werden graphisch dargestellt.
6.6	Die gezogenen Karten des Groupiers werden graphisch dargestellt.
6.7	Der gesetzte Betrag wird während dem Spiel angezeigt.
6.8	Der aktuelle Kontostand wird während dem Spiel angezeigt.
6.9	Wenn es die Regeln erlauben, kann ich entscheiden, ob ich eine Karte ziehe oder nicht.
6.10	Ich kann den zu setzenden Betrag selber wählen.
6.11	Der Betrag wird meinem Konto abgezogen.
6.12	Der gesetzte Betrag muss ganzzahlig sein.
6.13	Der Betrag muss mindestens 10 Coins Betragen.
6.14	Der gesetzte Betrag muss kleiner als das Bankguthaben.
6.15	Der Spieler kann die ersten beiden Karten ziehen.
6.16	Das Spiel gibt eine Meldung an, wenn ich verloren habe.
6.17	Das Spiel gibt eine Meldung an, wenn ich gewonnen habe.
6.18	Alle Karten kommen aus einem Gemischten, endlichen Kartenstapel.
6.19	Wenn es die Regeln erlauben, kann der Spieler eine weitere

	Karte ziehen.
6.20	Wenn ich Black Jack habe, dann wird mir der gesetzte Betrag zu 3:2 ausbezahlt.
6.21	Wenn ich einfach so gewinne, dann wird mein gesetzter Betrag verdoppelt ausbezahlt.
6.22	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
6.23	Bei einem Unentschieden wird eine Meldung angezeigt.
6.24	Beim Unentschieden wird der eingesetzte Betrag wieder auf das Konto gezahlt.
6.25	Das Spiel kann neu gestartet werden, indem ich eine Meldung bestätige.
6.26	Wenn ich verliere wird der Betrag meinem Konto abgezogen.
6.27	Ich kann das Spiel beliebig abbrechen.
6.28	Wenn ich das Spiel abbreche, dann wird mein gesetzter Betrag von meinem Konto abgezogen.

2.2.4.6 Anforderungsanalyse Yatzy

Anforderungs-Nummer	Anforderung
7.1	Es ist eine Übersichtseite vorhanden.
7.2	Die Übersichtsseite enthält den Titel des Spiels.
7.3	Die Übersichtsseite enthält den aktuellen Kontostand.
7.4	Mit Klick auf einen Button wird das Spiel gestartet.
7.5	Es kann ein Betrag ausgewählt werden.
7.6	Der Betrag wird meinem Konto abgezogen.
7.7	Der Betrag muss ganzzahlig sein.
7.8	Der Betrag muss grösser als Null sein.
7.9	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.
7.10	Die möglichen Figuren werden dargestellt.
7.11	Die Punkte der bereits gewählten Figuren werden angezeigt.
7.12	Die Anzahl verbleibende Würfe werden angezeigt.
7.13	Die Subtotale, der Bonus und die Gesamtanzahl der Punkte werden angezeigt.

7.14	Die selektierten und gewürfelten Würfel werden angezeigt.
7.15	Die selektierten Würfel werden nicht neu gewürfelt.
7.16	Die neu gewürfelten Würfel werden angezeigt.
7.17	Die Punkte der selektierten Würfel werden bei der Kategorie angezeigt.
7.18	Die entsprechende Kategorie wird gesperrt.
7.19	Meine Runde wird beendet und der nächste Spieler kommt dran.
7.20	Gewinnt der Spieler, so wird ihm der eingesetzte Betrag verdoppelt und ausgezahlt.
7.21	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
7.22	Beim Unentschieden erscheint eine Meldung.
7.23	Beim Unentschieden wird der eingesetzte Betrag wieder auf das Konto gezahlt.
7.24	Beim Bestätigen der Meldung kann das Spiel neu gestartet werden.
7.25	Beim Verlieren des Spiels wird der eingesetzte Betrag gelöscht.
7.26	Das Spiel kann jederzeit abgebrochen werden.
7.27	Der gesetzte Betrag geht beim Abbruch verloren.

2.2.4.7 Anforderungsanalyse Blackjack

Anforderungs-Nummer	Anforderung
8.1	Das Spiel hat eine Übersichtseite, welche den Titel des Spiels und den aktuellen Kontostand anzeigt.
8.2	Die Übersichtsseite sollte ein Button beinhalten, mit dem das Spiel gestartet werden kann.
8.3	Die gezogenen Karten des Spielers müssen angezeigt werden.
8.4	Die gezogenen Karten des Groupiers müssen angezeigt werden.
8.5	Der eingesetzte Betrag muss angezeigt werden.
8.6	Der aktuelle Kontostand muss angezeigt werden.
8.7	Der Spieler kann wählen, ob eine weitere Karte gezogen werden soll oder nicht.
8.8	Der Spieler kann einen Betrag auswählen, welches er einsetzen möchte.
8.9	Der eingesetzte Betrag wird von dem aktuellen Kontostand abgezogen.
8.10	Der ausgewählte Betrag muss ganzzahlig sein.
8.11	Der ausgewählte Betrag muss grösser als Null
8.12	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.

8.13	Wenn das Spiel verloren wurde wird eine Meldung angezeigt.
8.14	Wenn das Spiel gewonnen wurde wird eine Meldung angezeigt.
8.15	Alle gezogenen Karten kommen aus einem gemischten, endlichen Kartenstapel.
8.16	Gewinnt der Spieler mit Black Jack, wird der von ihm eingesetzte Betrag im Verhältnis 3:2 ausgezahlt.
8.17	Gewinnt der Spieler nicht mit Black Jack, so wird ihm der eingesetzte Betrag verdoppelt und ausgezahlt.
8.18	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
8.19	Beim Unentschieden erscheint eine Meldung.
8.20	Beim Bestätigen der Meldung von Unentschieden kann das Spiel neu gestartet werden.
8.21	Beim Unentschieden wird der eingesetzte Betrag wieder auf das Konto gezahlt.
8.22	Beim Verlieren des Spiels wird der eingesetzte Betrag gelöscht.
8.23	Das Spiel kann jederzeit abgebrochen werden, mit der dann das eingesetzte Betrag verloren geht.
8.24	Falls es die Regeln zulassen, erscheint ein Button für die Versicherung.
8.25	Mit einem Klick auf dem Versicherungsbutton, sollte der Spieler ein Betrag setzen können. Dieser Betrag muss positiv, ganzzahlig sein und es darf nicht grösser als das Kontostand sein.

8.26	Hat der Groupier im nächsten Zug ein Black Jack, wenn der Versicherungsbutton geklickt wurde, wird eine Versicherungsbetrag von 2:1 ausbezahlt.
8.27	Hat der Groupier im nächsten Zug kein Black Jack, wenn der Versicherungsbutton geklickt wurde, wird das eingesetzte Betrag verloren.
8.28	Falls es die Regeln zulassen erscheint ein Button zur Verdoppelung des Einsatzes.
8.29	Beim Klick auf den Verdopplungsbutton wird der bereits eingesetzte Betrag nochmals dem Konto abgezogen und dem eingesetzten Betrag hinzugefügt.
8.30	Wenn der Spieler den Betrag durch den Verdopplungsbutton verdoppelt hat, wird nur noch eine Karte gezogen.

2.2.4.8 Anforderungsanalyse Roulette

Anforderungs-Nummer	Anforderung
9.1	Es ist eine Übersichtseite vorhanden.
9.2	Die Übersichtsseite enthält den Titel des Spiels.
9.3	Die Übersichtsseite enthält den aktuellen Kontostand.
9.4	Mit Klick auf einen Button wird das Spiel gestartet.
9.5	Es kann eine Zahl, Zahlengruppe, Farbe oder Art ausgewählt werden.
9.6	Es kann ein Betrag ausgewählt werden.
9.7	Der Betrag wird meinem Konto abgezogen.
9.8	Der Betrag muss ganzzahlig sein.
9.9	Der Betrag muss grösser als Null sein.
9.10	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.
9.11	Einsatz auf eine einzelne Zahl (Straight Up Bets): Die Auszahlung liegt hier bei 35:1.
9.12	Einsatz auf zwei benachbarte Zahlen (Split Bets): Die Gewinnauszahlung beträgt 17:1.
9.13	Einsatz auf drei Zahlen (Street Bets): Die Gewinnauszahlung beträgt 11 zu 1.
9.14	Einsatz auf vier Zahlen (Corner Bets): Die Gewinnauszahlung beträgt 8:1.
9.15	Einsatz auf fünf Zahlen (Five Number Bets): Wird das mit 6:1 ausgezahlt.
9.16	Rot oder schwarz: Wird das mit 1:1 ausgezahlt.
9.17	Ungerade oder Gerade: Wird das mit 1:1 ausgezahlt.
9.18	Niedrig oder Hoch: Im Gewinnfall wird Ihnen der gleiche Betrag ausgezahlt.

9.19	Spalten: Wird das mit 2:1 ausgezahlt.
9.20	Dutzend: Im Gewinnfall wird der doppelte Einsatz ausgezahlt.
9.21	Der gewonnene Betrag sowie der neue Kontostands werden angezeigt.

2.2.4.9 Anforderungsanalyse Bingo

Anforderungs-Nummer	Anforderung
10.1	Es ist eine Übersichtseite vorhanden.
10.2	Die Übersichtsseite enthält den Titel des Spiels.
10.3	Die Übersichtsseite enthält den aktuellen Kontostand.
10.4	Mit Klick auf einen Button wird das Spiel gestartet.
10.5	Der Betrag für die Karte darf nicht grösser als das Guthaben auf dem Konto sein.
10.6	Der Betrag wird meinem Konto abgezogen.
10.7	Die erworbene Karte wird angezeigt.
10.8	Die markierten Felder sind jederzeit ersichtlich.
10.9	Die Nummer auf der aktuell gezogenen Bingo-Kugel ist ersichtlich.
10.10	Die Anzahl der virtuellen Gegenspieler ist ersichtlich.
10.11	Der aktuelle Kontostand wird angezeigt.
10.12	Es wird alle 30 Sekunden eine neue Bingo-Kugel aus der Trommel gezogen.
10.13	Die Nummer auf der aktuell gezogenen Bingo-Kugel wird gezeigt und bleibt für den Rest der Runde ersichtlich.

10.14	Die Felder auf den Karten können ohne Prüfung oder Restriktionen markiert werden.
10.15	Das Spiel bietet die Möglichkeit, den geglaubten Gewinn anzumelden.
10.16	Meine Karte ist auf die Legitimität des Gewinnes validiert.
10.17	Es wurde von keinem meiner Gegenspieler vor mir «Bingo» angemeldet.
10.18	Einer meiner 5-10 Gegenspieler mit je einer Karte, brechen das Spiel mit «Bingo» mit einer Verzögerung von 15 Sekunden vor mir ab, wenn sie einen Gewinn auf der Karte verzeichnen.
10.19	Mein mit «Bingo» angemeldeter Gewinn bestätigt sich als falsch.
10.20	Der Gewinn wird meinem Konto gutgeschrieben.
10.21	Das Spiel kann jederzeit abgebrochen werden.
10.22	Bereits gesetzte Beträge gehen beim Abbruch verloren.

2.2.5 ISO 9241-110 Kriterien

Wir haben schon Vorkenntnisse mit der ISO 9241-110 -Kriterien gemacht, deshalb versuchen wir diese Kriterien in unserem Programm stets zu berücksichtigen.

Aufgabenangemessenheit

Der Benutzer wird die gewünschten Spiele und Funktionen schnell erreichen, da die Navigation des Programms klar und intuitiv ist.

Selbstbeschreibungsfähigkeit

Der Benutzer weiss stets wo er sich gerade befindet, da jede Seite des Programms einen Titel hat.

Dazu kann der Benutzer bei den Eingaben keine grossen Fehler machen durch falsch Eingaben, da der Benutzer dann mit einer Meldung informiert wird. Die Meldung zeigt dem Benutzer, was er falsch gemacht hat.

Lernförderlichkeit

Der Benutzer kann immer auf die Regeln der Spiele zugreifen, falls er nicht weiss wie das Spiel genau funktioniert. Diese Funktion wird den Benutzer zu einer Webseite verlinken.

Steuerbarkeit

Der Benutzer kann selbstständig entscheiden welches Spiel er spielen möchte. Darüber hinaus kann er jedes Spiel verlassen und ein anderes Spiel weiterspielen.

Erwartungskonformität

Der Benutzer wird beim Einloggen und beim Kaufen von Chips immer benachrichtigt, damit der Benutzer versteht was bei den Arbeitsschritten passiert.

Individualisierbarkeit

Der Benutzer kann seine Benutzernamen eingeben wie er es gerne haben möchte.

Fehlertoleranz

Wenn ein Fehler passiert, soll dem Benutzer klar aufgezeigt werden, was falsch ist. Auch bei einer falschen Eingabe.

Anwendung der ISO 9241-110 Kriterien

Durch dieses Bild, unsere Login Seite des Programms, versuchen wir die sieben Kriterien der ISO 9241-110 zu beschreiben.

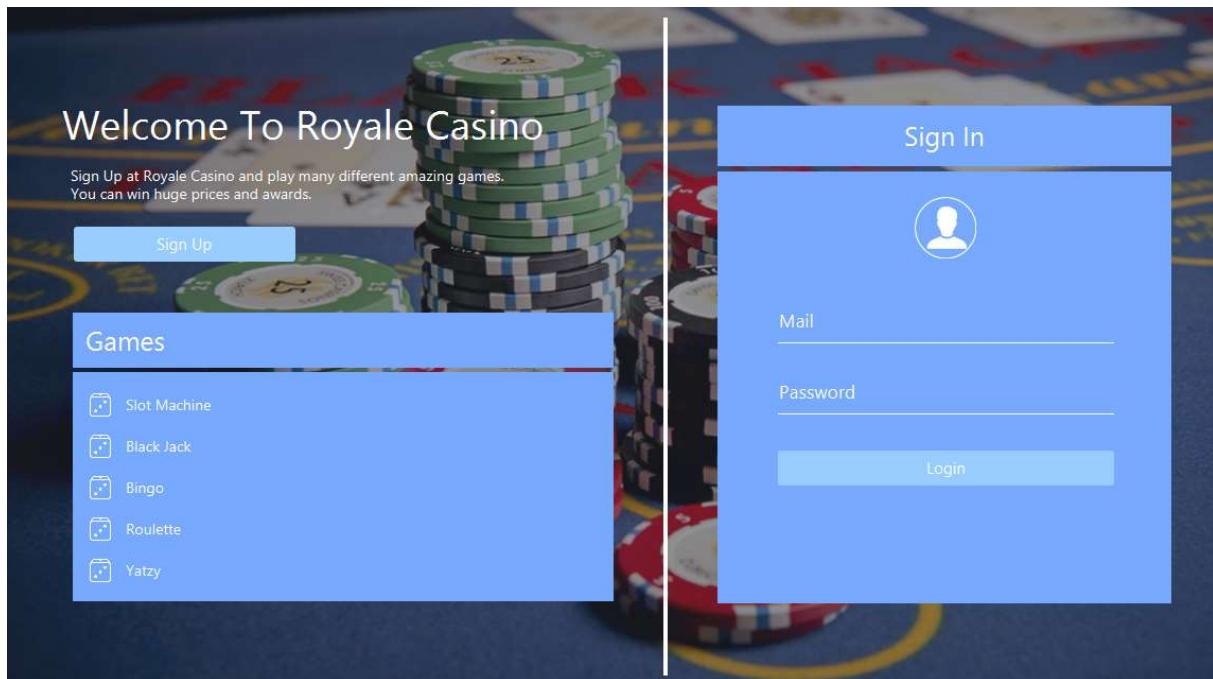


Abbildung 4: Login Screen Casino Royale

Aufgabenangemessenheit

Der Benutzer kann die Funktionen wie «Sign Up» und «Sign In» schnell erreichen, da die Navigation des Programms auf einen Blick sichtbar ist.

Selbstbeschreibungsfähigkeit

Der Benutzer weiss, dass er sich auf dem Login Seite befindet. Somit wird es dem Benutzer auf einem Blick sichtbar, wo er sich befindet.

Dazu wird der Benutzer bei einem falschen Anmeldeversuch mit einer Meldung informiert, dass die Anmeldedaten ungültig sind.

Lernförderlichkeit

Oben rechts haben wir eine Hilfefunktion, welche dem Benutzer genau erklärt was er machen muss, wenn er nicht mehr weiter weiss.

Steuerbarkeit

Der Benutzer kann selbstständig entscheiden, ob er sich durch «Sign Up» ein neues Konto erstellen möchte oder ob er sich durch «Sign In» mit dem bestehenden Konto anmelden will.

Erwartungskonformität

Wenn der Benutzer beim Einloggen (Sign In) seine Anmeldedaten richtig eingegeben hat, wird es durch eine Meldung bestätigt.

Individualisierbarkeit

Der Benutzer kann seinen Namen und Vorname eingeben wie er es gerne haben möchte.

Fehlertoleranz

Wenn der Benutzer beim Einloggen (Sign In) ein Fehler macht, wird ein Fehlermeldung auftauchen, worin «ungültig» stehen wird. Somit werden falsche Anmeldeversuche als ungültig angezeigt.

2.3 Planen

2.3.1 Erste Idee

Erste Idee für Programmablauf:

Unser Plan ist es eine Desktop-Software zu machen, in welcher sich der Benutzer zuerst anmelden muss. Falls der Benutzer keine Anmeldedaten hat, muss der Benutzer sich registrieren. Somit kann sich jeder Benutzer bei dem Programm anmelden.

Nach einer erfolgreichen Anmeldung sollte der Benutzer in einem Shop gelangen. Auf dieser Seite kann man Echtgeld in Chips umwandeln, die von uns gewählte Währung ist US Dollar.

Nach dem Kaufen von Chips kommt man in die Spielauswahlseite. Auf dieser Seite hat man die Möglichkeit sich zwischen fünf verschiedene Spiele zu entscheiden. Natürlich kann der Benutzer das Spiel immer beenden und zurück in die Spielauswahlseite gelangen, um eine anderes Spiel zu spielen.

2.3.2 Sitemap-Navigation

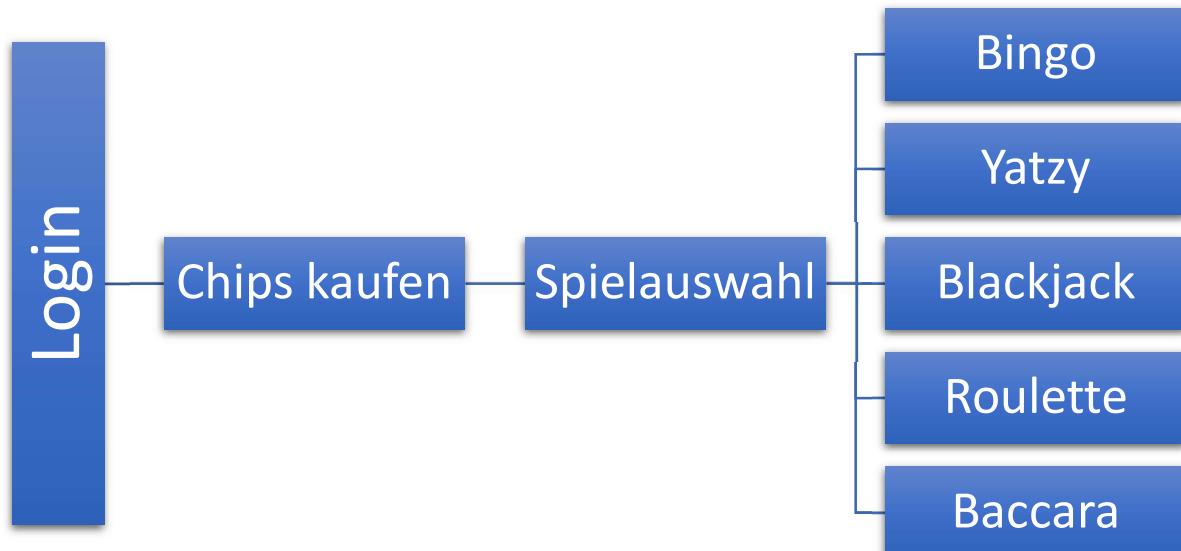


Abbildung 5: Sitemap-Navigation, alle Screens

2.3.3 Use Case

2.3.3.1 Use Case Casino Royale

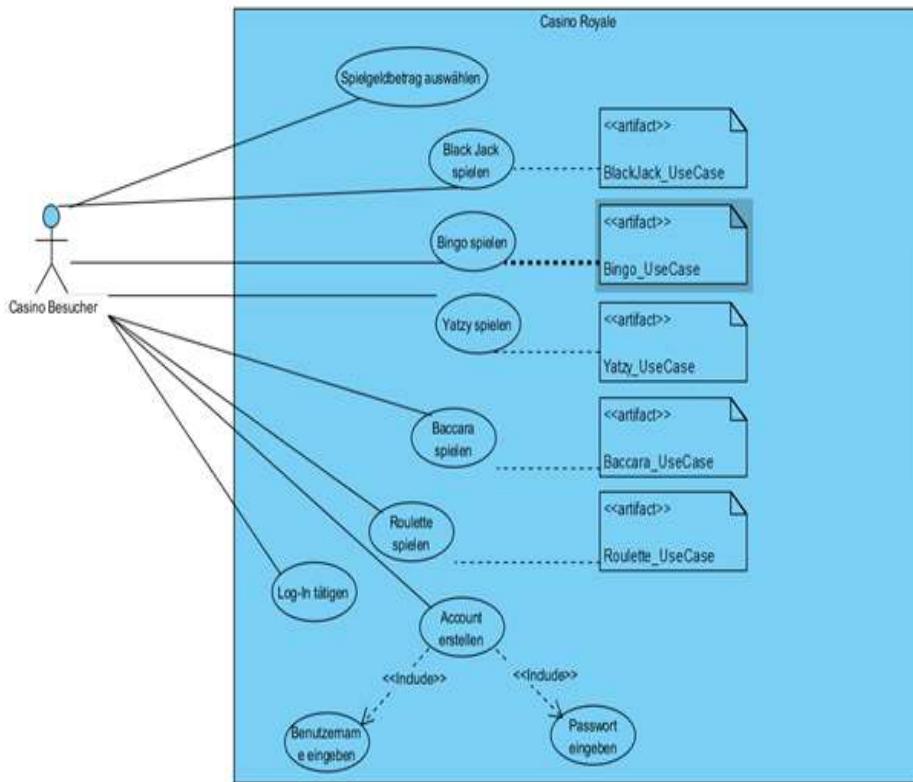


Abbildung 6: Use-Case Casino Royale

In diesem Bild ist der Casino Besucher zu sehen. Der Besucher kann Spiele spielen, Log-In oder Sign-In tätigen und er kann Chips kaufen.

2.3.3.2 Use Case Baccara

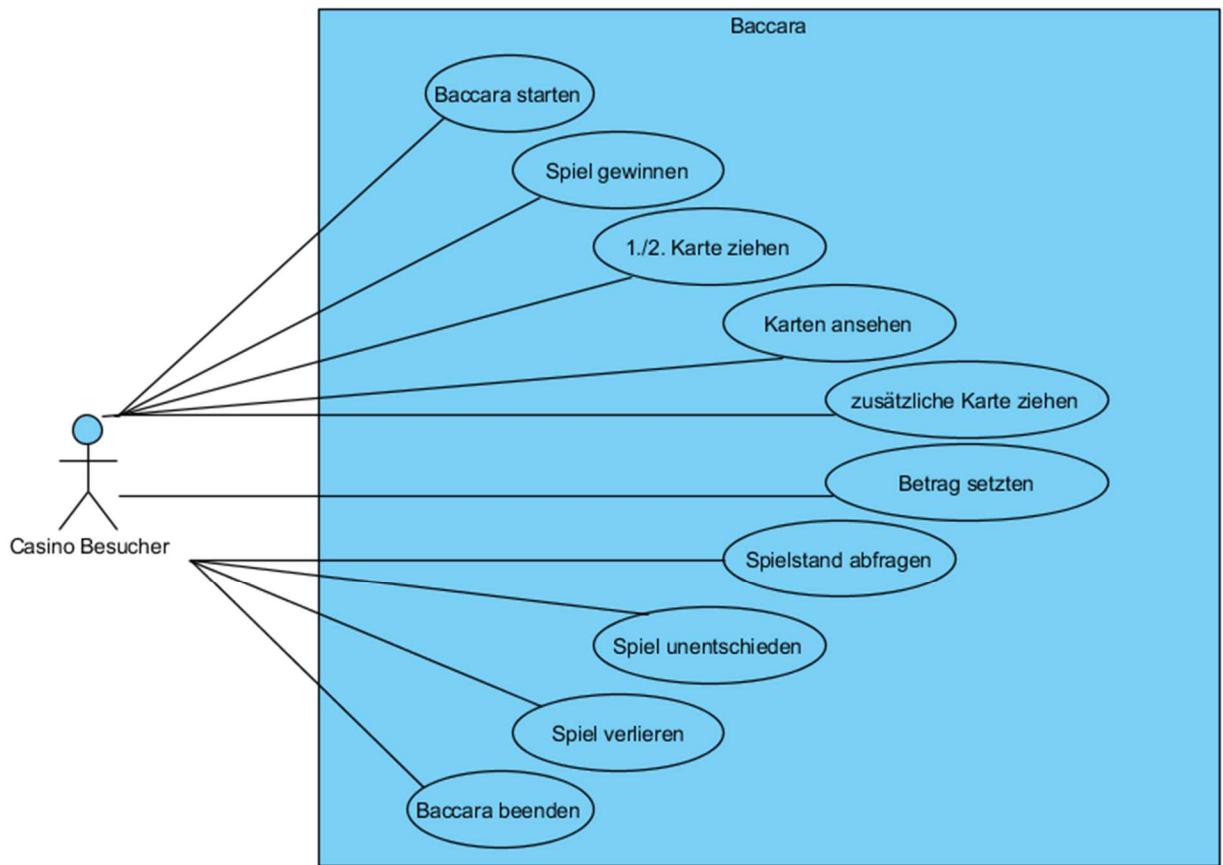


Abbildung 7: Use-Case Baccara

Hier ist ein Use-Case Diagramm zu sehen, bei dem der Casino Besucher verschiedene Aktionen im Spiel Baccara.

2.3.3.3 Use Case Yatzy

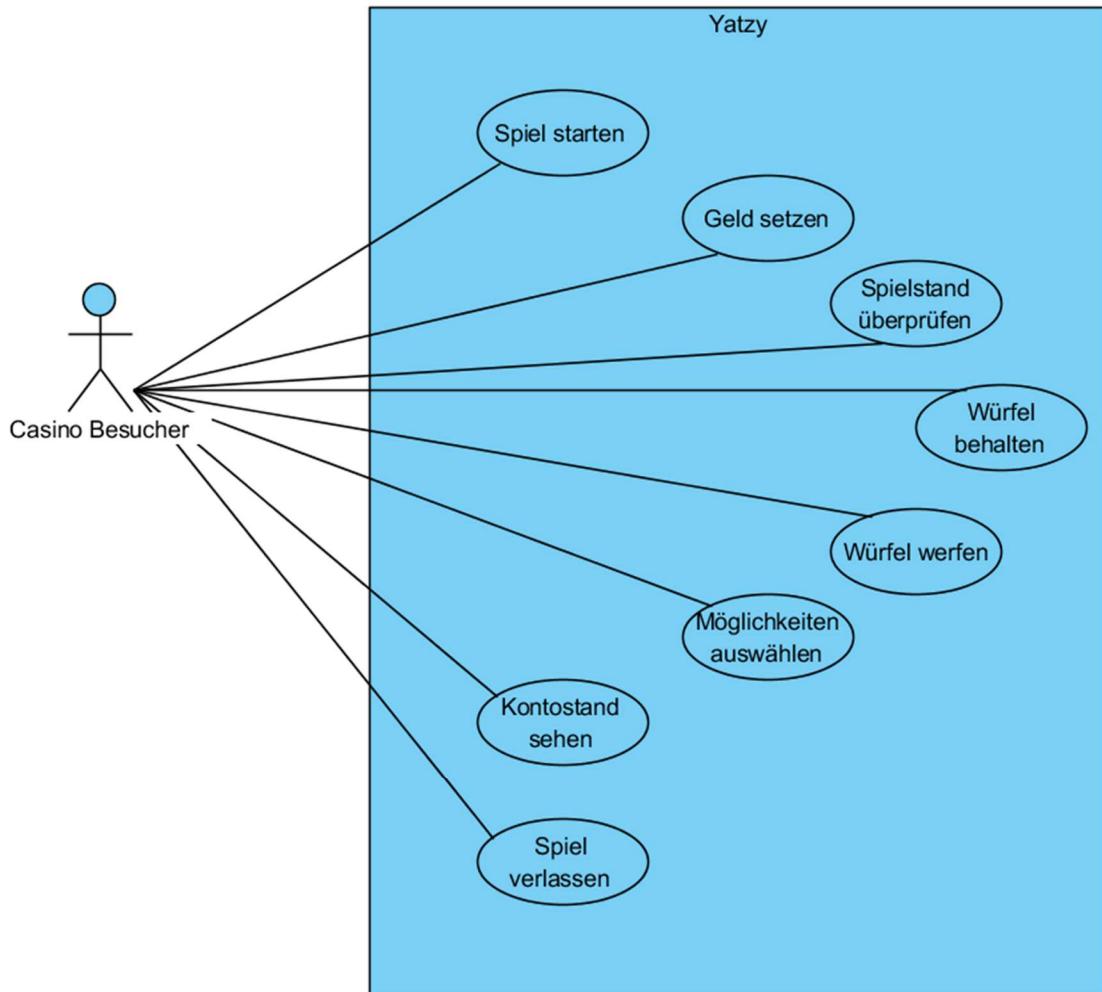


Abbildung 8: Use-Case Yatzy

Hier ist ein Use-Case Diagramm zu sehen, bei dem der Casino Besucher verschiedene Aktionen im Spiel Yatzy.

2.3.3.4 Use Case Blackjack

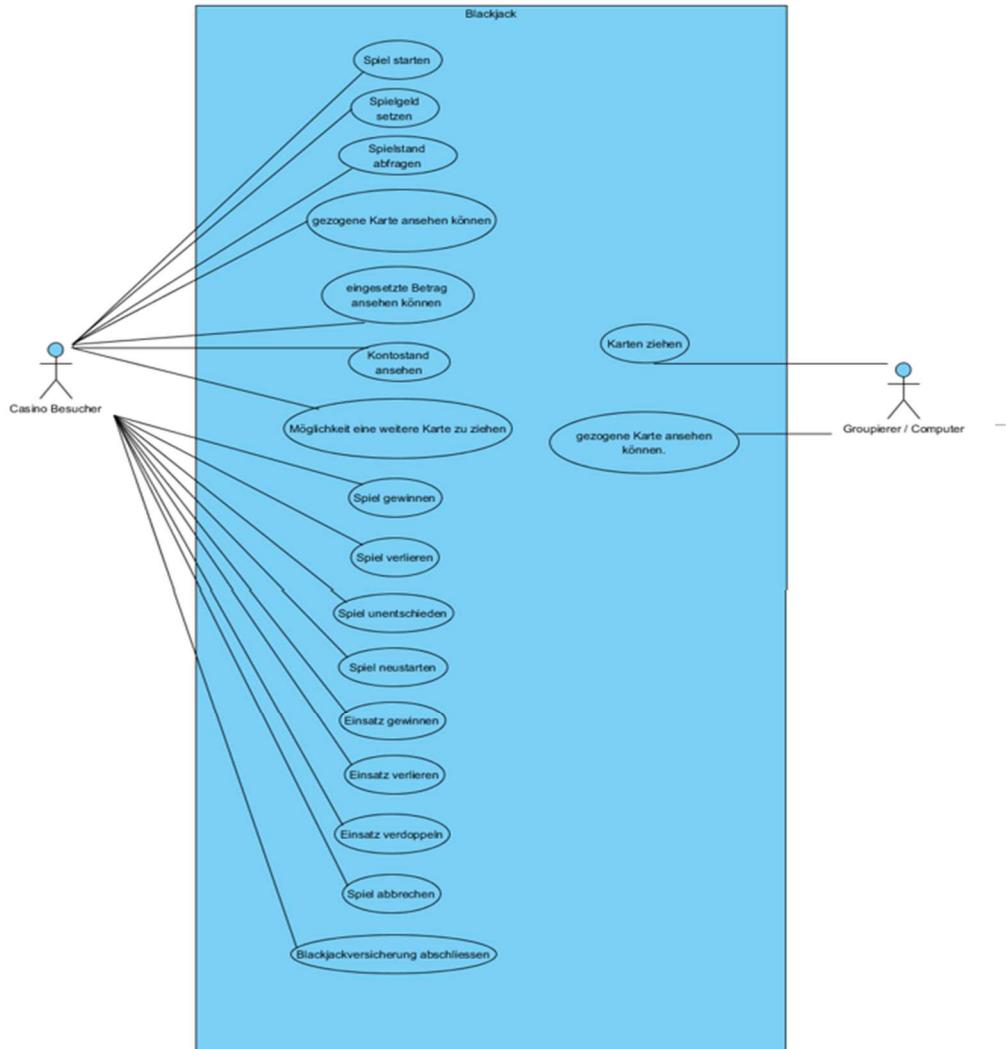


Abbildung 9: Use-Case Blackjack

Hier ist ein Use-Case Diagramm zu sehen, bei dem der Casino Besucher verschiedene Aktionen im Spiel Blackjack.

2.3.3.5 Use Case Roulette

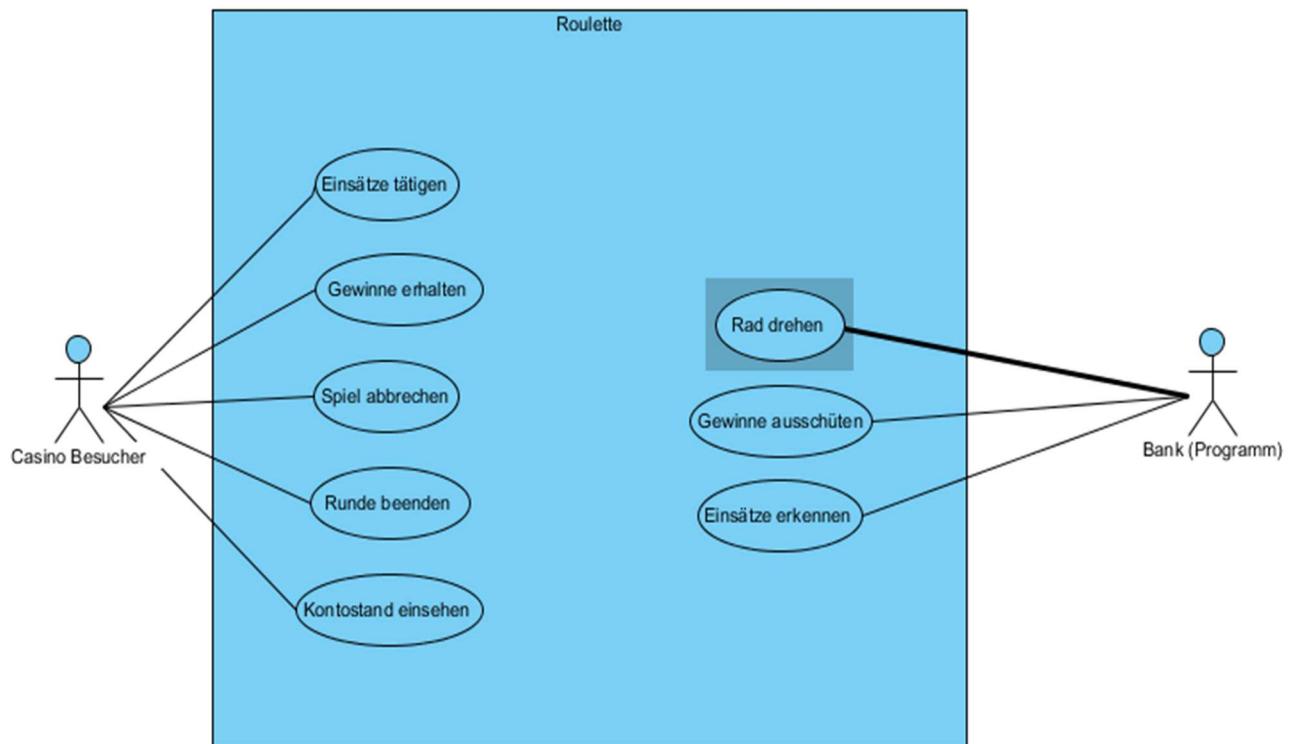


Abbildung 10: Use-Case Roulette

Hier ist ein Use-Case Diagramm zu sehen, bei dem der Casino Besucher verschiedene Aktionen im Spiel Roulette.

2.3.3.6 Use Case Bingo

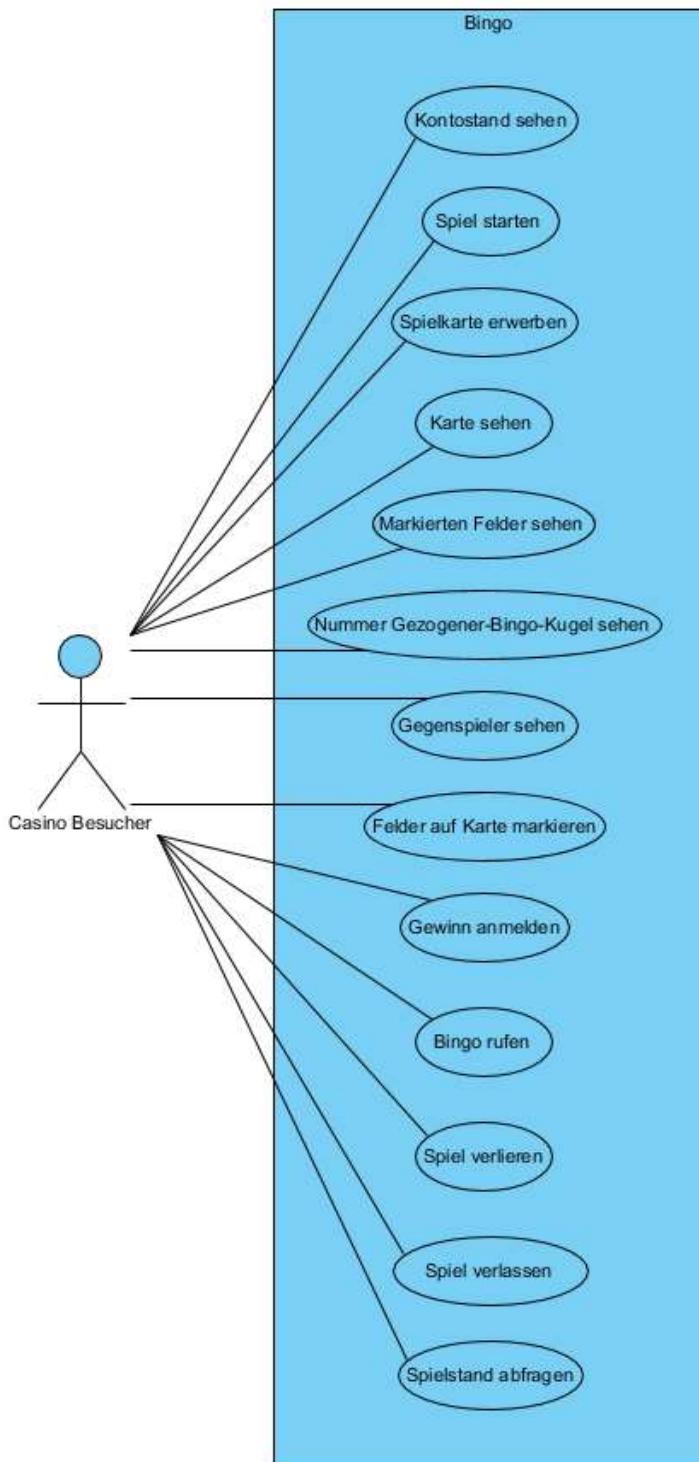


Abbildung 11: Use-Case Bingo

Hier ist ein Use-Case Diagramm zu sehen, bei dem der Casino Besucher verschiedene Aktionen im Spiel Bingo.

2.3.3.7 Use Case Administrator

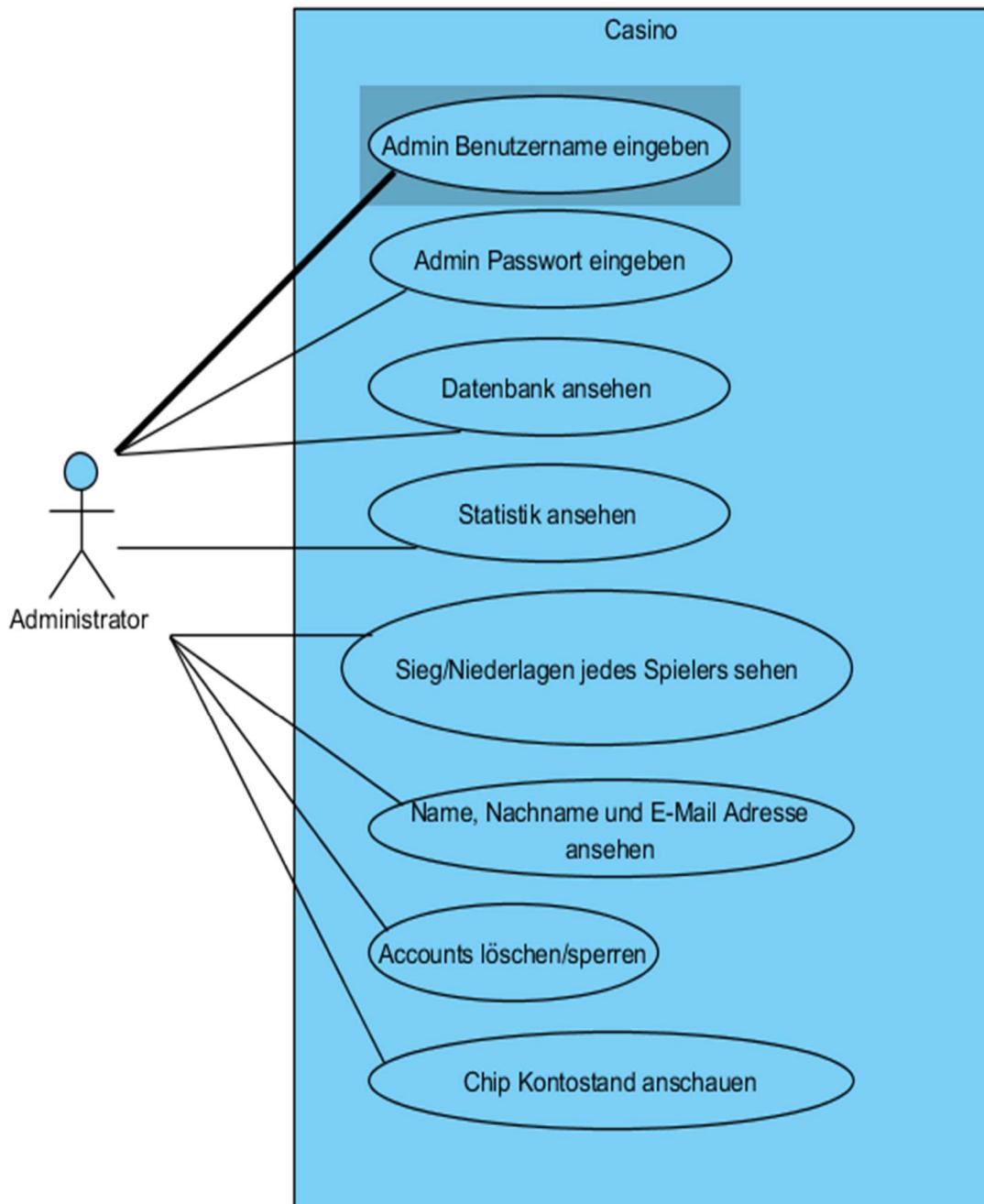


Abbildung 12: Use-Case Admin

Hier ist ein Use-Case Diagramm zu sehen, bei dem die Aktionen dargestellt sind, die der Administrator machen kann.

2.3.4 Szenario

2.3.4.1 Szenario Casino Royale

Titel:

Funktionstüchtiges Online-Casino

Aktoren:

Casino Besucher, Administrator

Situation vor dem Szenario:

Das Casino Royale Programm ist gestartet und der Benutzer kann sich anmelden oder muss sich registrieren.

Situation nach dem Szenario:

Der Benutzer hat ein Glückspiel oder mehrere verschiedene Glücksspiele gespielt. Somit hat der Benutzer Spielgeld gewonnen oder verloren.

Ablauf:

Der Benutzer meldet sich an, und dann kommt er in den Shop wo er Echtgeld in virtuelle Chips umwandeln kann. Nachdem Kaufen von Chips kommt er in die Spielübersichtsseite, wo er ein Glückspiel auswählen kann, die er gerne spielen möchte.

Eventuell:

Kein Spielgeld mehr übrig, somit muss der Benutzer noch mehr Chips kaufen. Es kann auch sein, dass der Benutzer einen grossen Gewinn erzielt hat oder einen grossen Verlust gemacht hat und somit hört er auf.

2.3.4.2 Szenario Baccara

Titel:

Risikospiel Baccara spielen

Aktoren:

Casino Besucher

Situation vor dem Szenario:

Casinospiel Baccara spielen mit der Absicht Spielgeld zu gewinnen. Der Spieler überlegt, wie viel Geld er einzusetzen.

Situation nach dem Szenario:

Risikospiel gespielt Spielgeld gewonnen, verloren oder unentschieden gespielt.

Ablauf:

Der Besucher betritt das Spiel und kann sein Spielgeld setzen. Der Besucher spielt und gewinnt oder verliert Geld. Der Spieler kann auch unentschieden spielen und hat immer die Möglichkeit nochmals eine Runde zu spielen.

Eventuell:

Kein Spielgeld mehr, dann kann der Besucher nicht mehr spielen.

2.3.4.3 Szenario Yatzy

Titel:

Glücksspiel Yatzy

Aktoren:

Casino Besucher

Situation vor dem Szenario:

Yatzy spielen um Gewinn zu bringen. Der Besucher hat sich für das Spiel Yatzy entschieden. Er überlegt wieviel Geld er setzen will.

Situation nach dem Szenario:

Yatzy gespielt Spielgeld gewonnen/verloren.

Ablauf:

Der Besucher betritt das Spiel und kann sein Spielgeld setzen. Er kann Würfeln, seine Figuren auswählen und auch Würfel behalten um seine Zielfigur zu bekommen. Er versucht zu gewinnen, wenn ja verdoppelt sich sein gesetztes Geld, wenn nicht dann verliert er es.

Eventuell:

Kein Spielgeld mehr, dann kann der Besucher nicht mehr spielen.

2.3.4.4 Szenario Blackjack

Titel:

Gewinnbringendes Blackjack Spiel

Aktoren:

Casino Besucher

Situation vor dem Szenario:

Das Casino Royale Programm ist gestartet und der Spieler hat sich angemeldet. Dann spielt der Besucher das Glückspiel Blackjack mit der Absicht das Spielgeld zu vermehren.

Situation nach dem Szenario:

Glückspiel gespielt Spielgeld gewonnen, verloren oder unentschieden gespielt. Lust auf eine weitere Runde.

Ablauf:

Der Besucher betritt das Spiel und kann sein Spielgeld setzen. Der Besucher spielt und gewinnt oder verliert das eingesetztes Geld. Der Spieler kann auch unentschieden spielen und hat immer die Möglichkeit nochmals eine Runde zu spielen.

Eventuell:

Kein Spielgeld mehr übrig, somit kann der Besucher nicht mehr weiterspielen.

2.3.4.5 Szenario Roulette

Titel:

Glücksspiel Roulette spielen

Aktoren:

Casino Besucher

Situation vor dem Szenario:

Roulette spielen mit der Absicht Spielgeld zu gewinnen. Das Casino Royale Programm ist gestartet und der Spieler hat sich angemeldet. Zudem hat man das Spiel Roulette ausgewählt.

Situation nach dem Szenario:

Der Benutzer gespielt Spielgeld gewonnen/verloren.

Ablauf:

Der Besucher betritt das Spiel und kann sein Spielgeld setzen. Er kann verschiedene Einsätze tätigen, zum Beispiel: Einfach, Doppelt, Dreifach. Er kann sagen, wenn er fertig gesetzt hat und die Runde geht zum nächsten Spieler. Der Besucher kann bei jeder Runde gewinnen oder verlieren. Der Spieler kann jeder Zeit das Spiel verlassen.

Eventuell:

Kein Spielgeld mehr, dann kann der Besucher nicht mehr spielen

2.3.4.6 Szenario Bingo

Titel:

Bingo Spiel

Aktoren:

Casino Besucher

Situation vor dem Szenario:

Bingo spielen mit der Absicht gegen die anderen Spieler zu gewinnen und die Beträge der anderen für sich zu gewinnen.

Situation nach dem Szenario:

Bingo gespielt Spielgeld gewonnen oder verloren.

Ablauf:

Der Besucher betritt das Spiel und kann sein Spielgeld setzen. Der Besucher spielt und gewinnt oder verliert Geld. Der Spieler kann das Spiel auch abbrechen, verliert allerdings die gesetzten Beträge.

Eventuell:

Kein Spielgeld mehr, dann kann der Besucher keine weitere Runde mehr spielen.

2.3.5 Klassendiagramm

2.3.5.1 Klassendiagramm Casino Royale

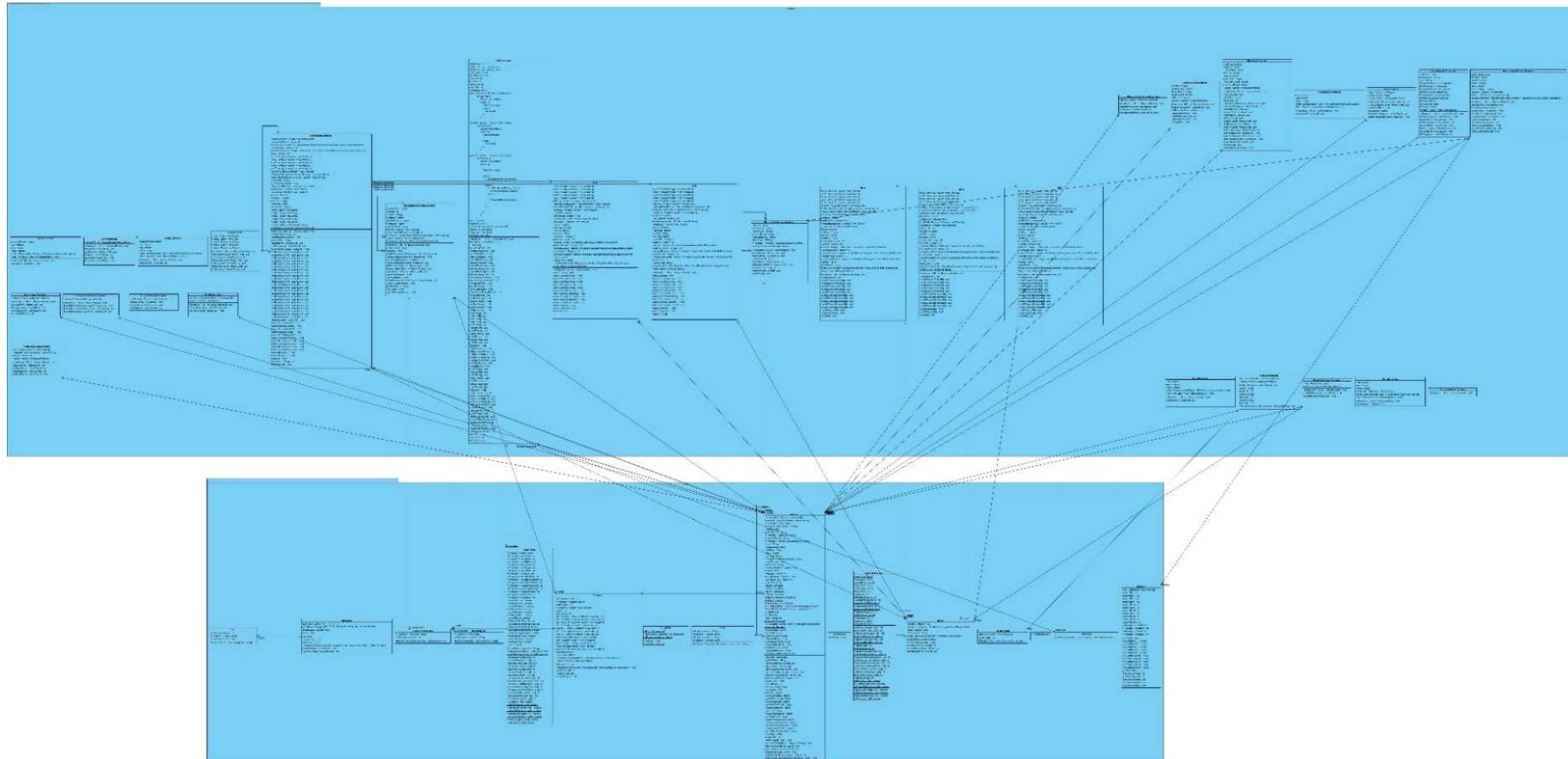


Abbildung 13: UML Casino Royale

Alle Controller Klassen und Models sind in diesem UML zu sehen.

2.3.5.2 Klassendiagramm Baccara

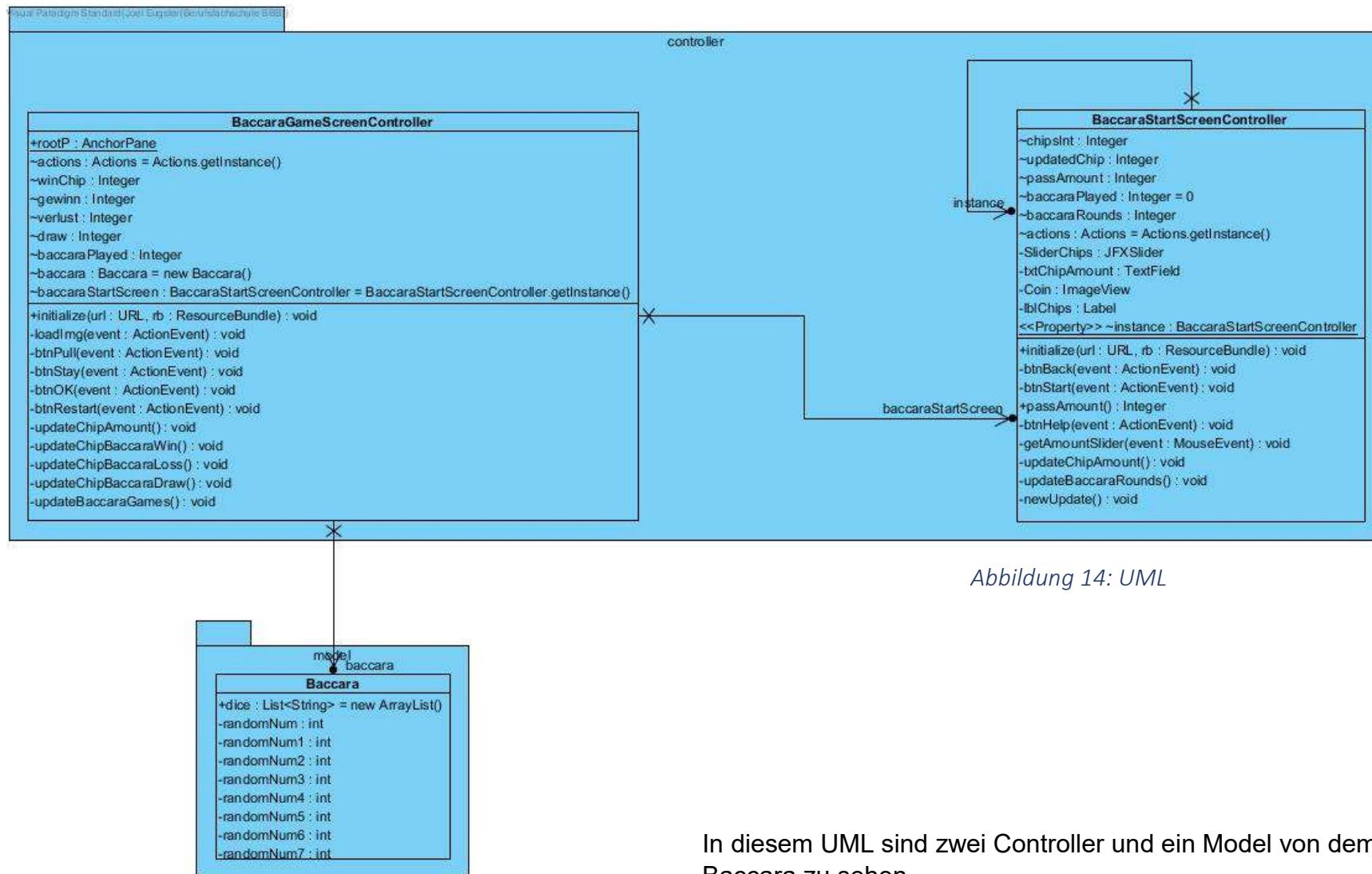


Abbildung 14: UML

In diesem UML sind zwei Controller und ein Model von dem Spiel Baccara zu sehen.

2.3.5.3 Klassendiagramm Yatzy

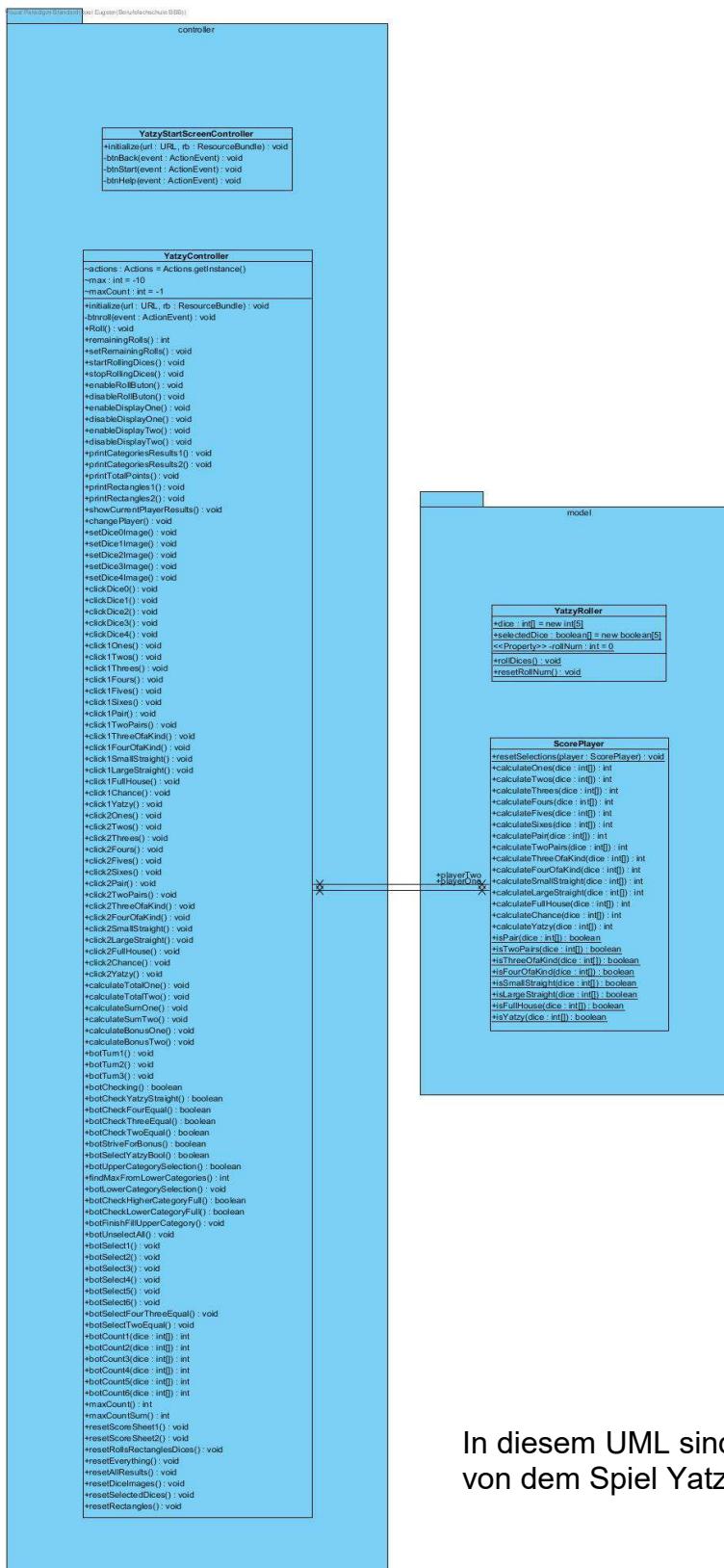


Abbildung 15: UML Yatzy

In diesem UML sind zwei Controller und zwei Model von dem Spiel Yatzy zu sehen.

2.3.5.4 Klassendiagramm Blackjack

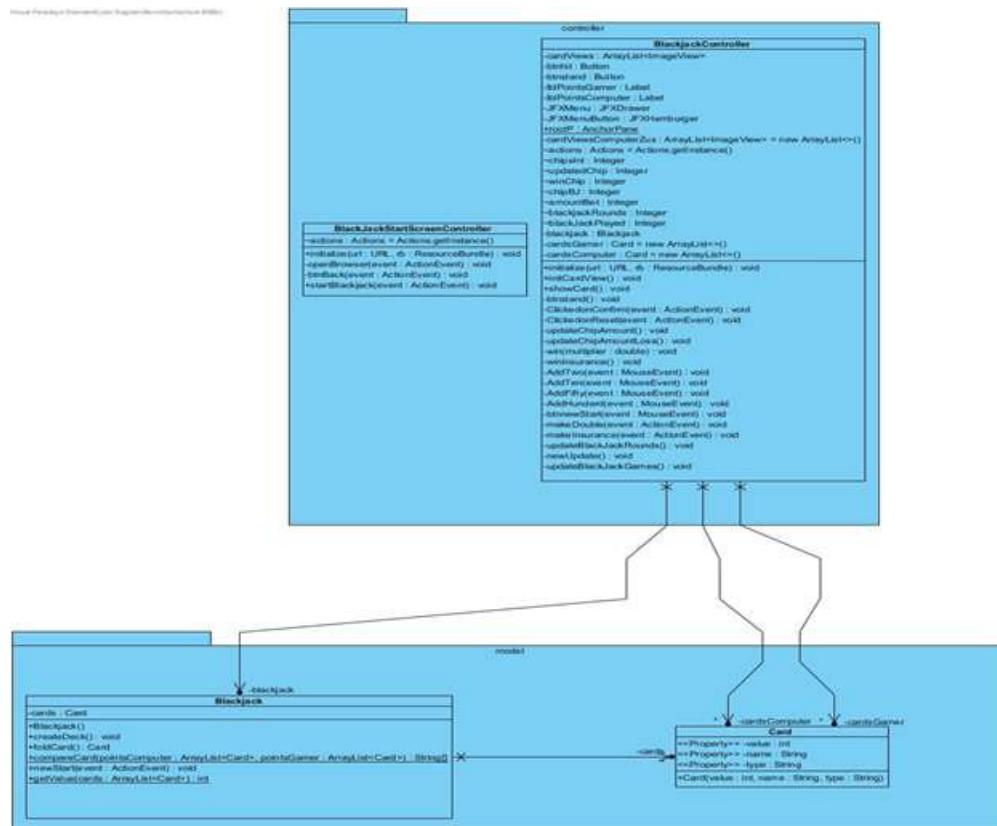


Abbildung 16: UML Blackjack

In diesem UML sind zwei Controller und ein Model von dem Spiel Blackjack zu sehen.

2.3.5.5 Klassendiagramm Roulette

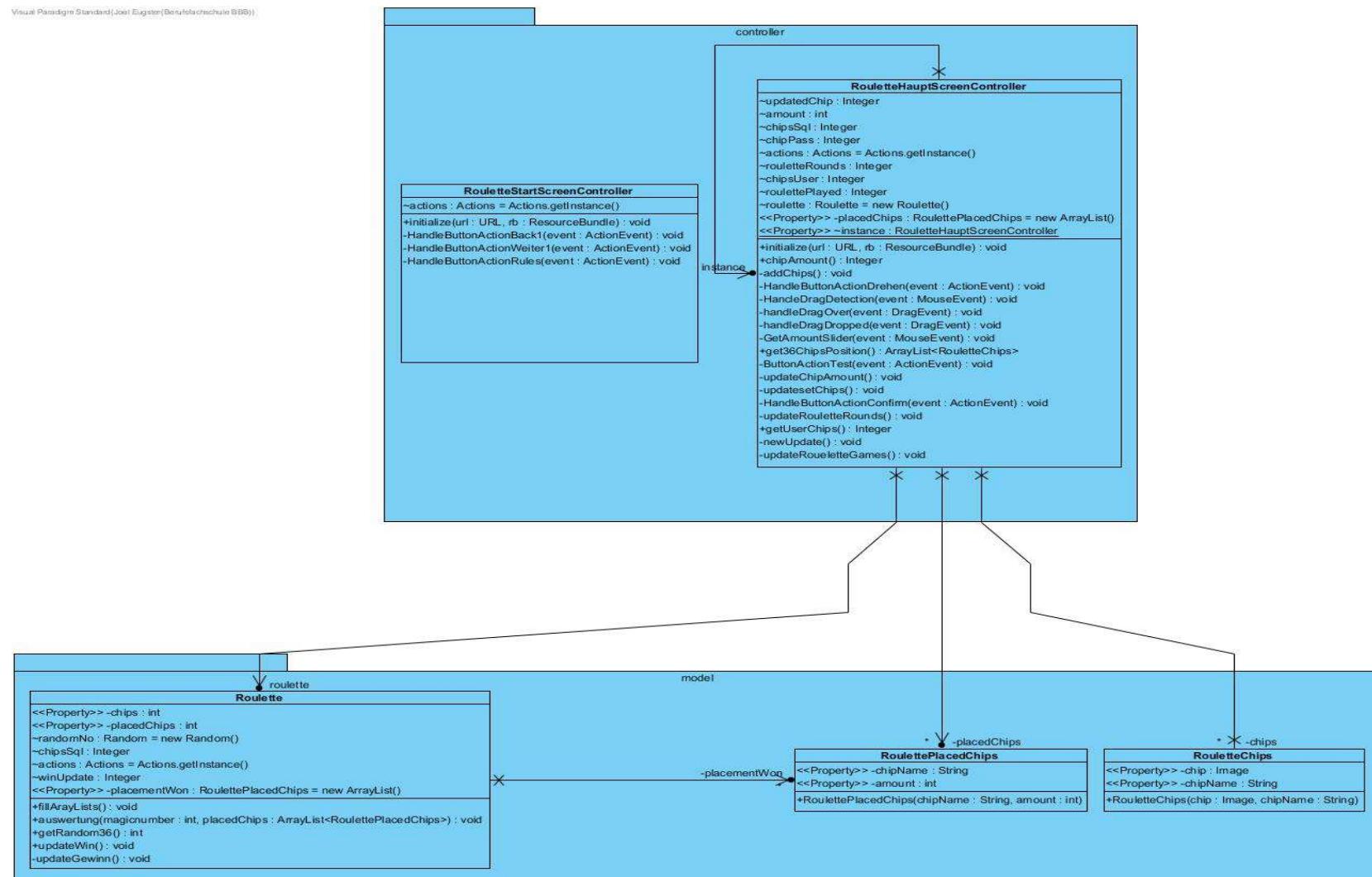


Abbildung 17: UML Roulette

In diesem UML sind zwei Controller und drei Model von dem Spiel Roulette zu sehen.

2.3.5.6 Klassendiagramm Bingo

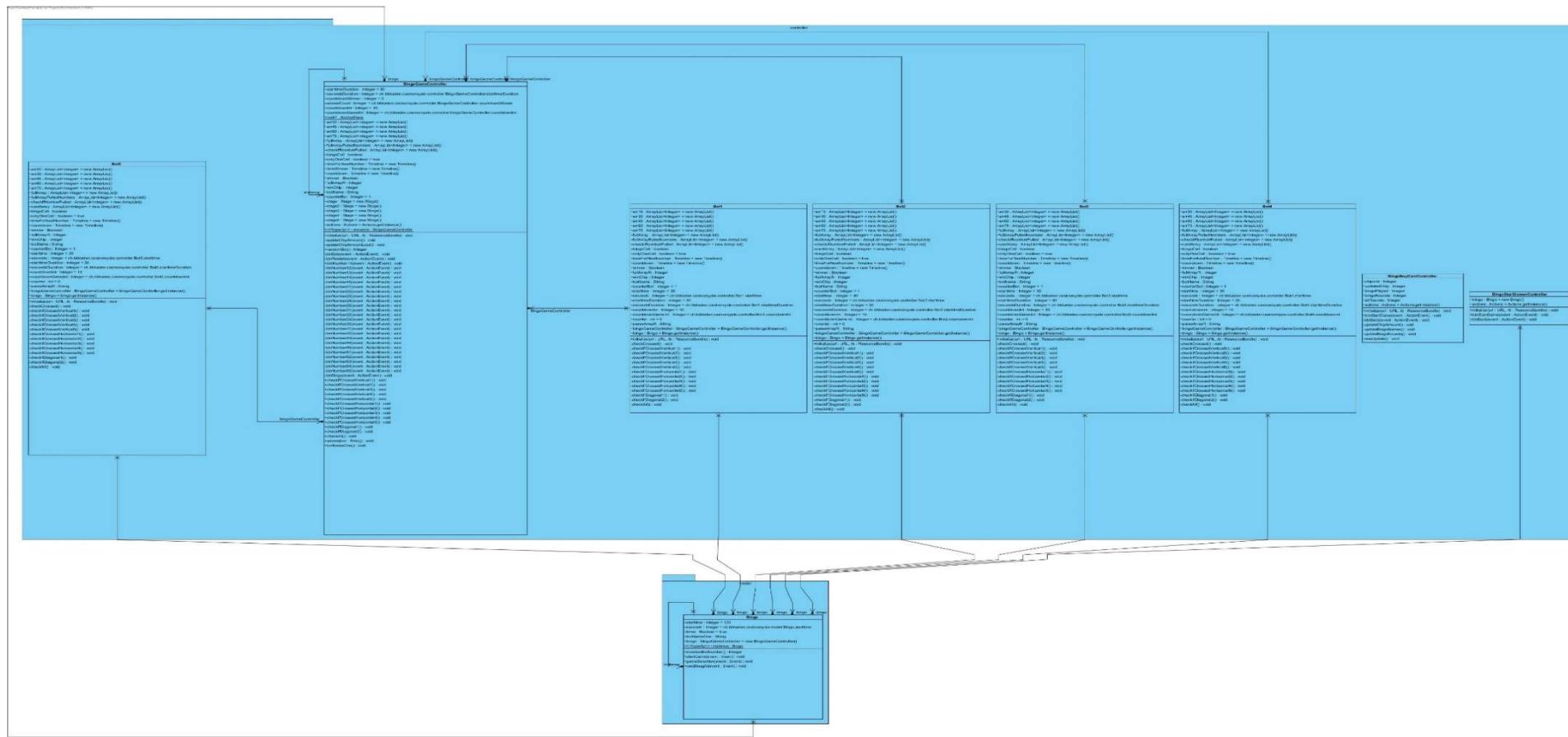


Abbildung 18: UML Bingo

In diesem UML sind drei Controller, ein Model und fünf Bingo Bots von dem Spiel Bingo zu sehen.

2.3.6 GUI-Design

2.3.6.1 Design Login

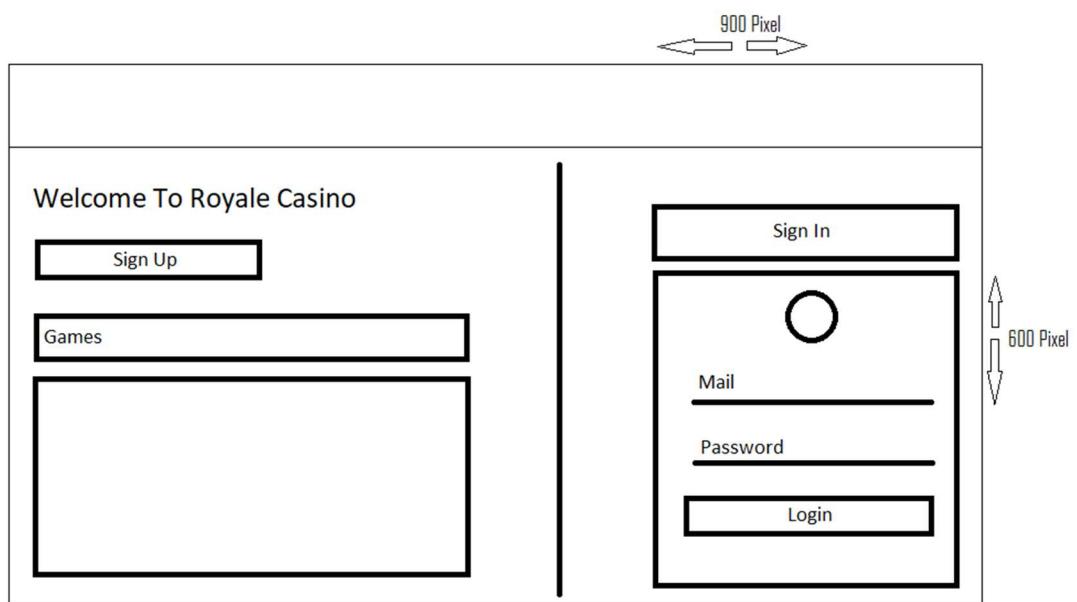
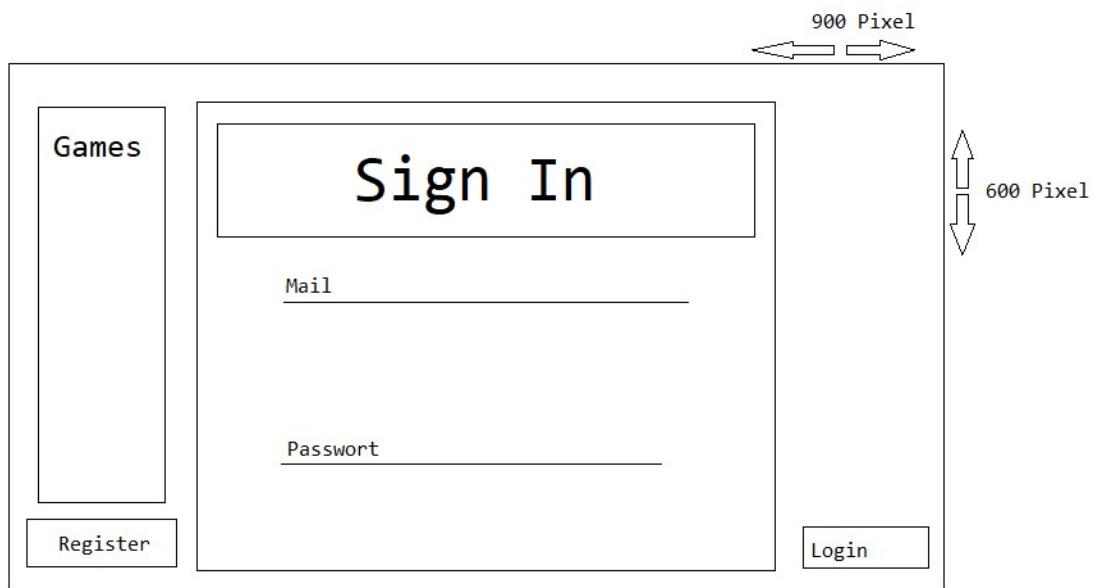


Abbildung 19 und 20: Design Login

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Login Screens zu sehen. Planen: Skizzen

2.3.6.2 Design Home

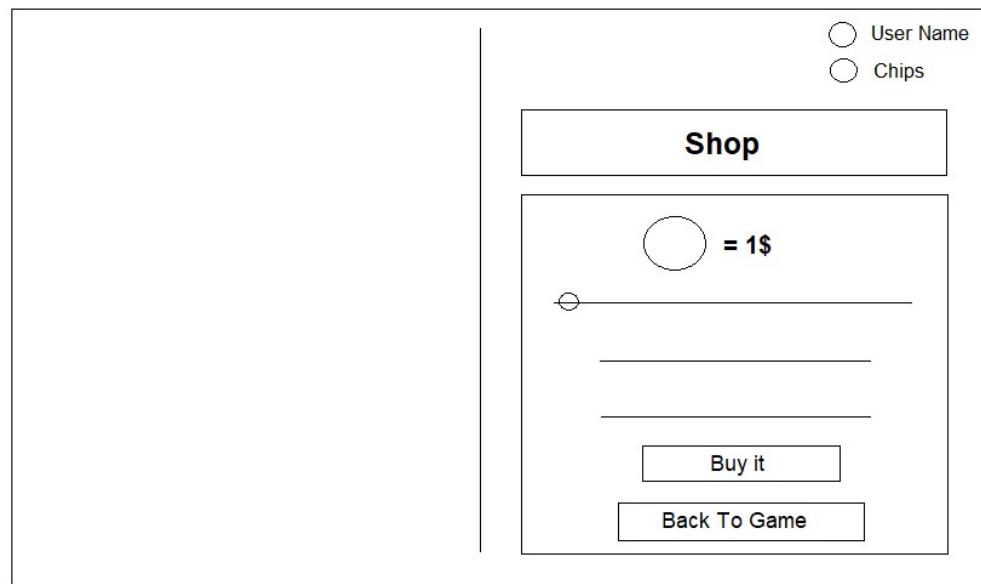


Abbildung 21 und 22: Design Home

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Home Screens zu sehen. Planen: Skizzen

2.3.6.3 Design Shop

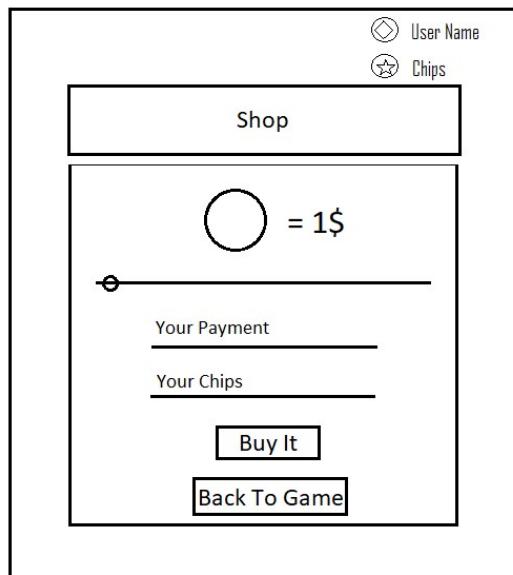
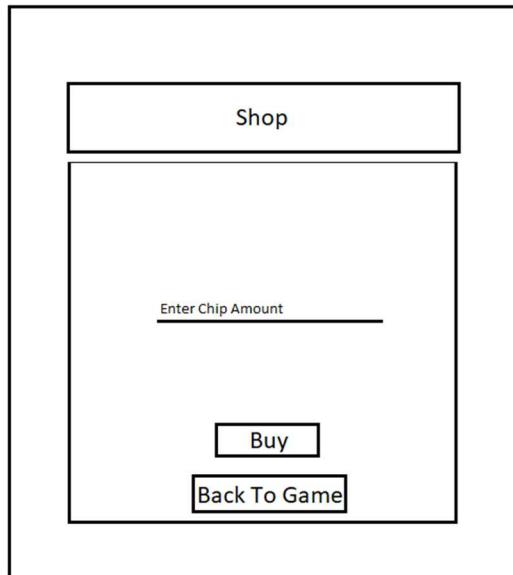


Abbildung 23 und 24: Design Shop

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Shop Screens zu sehen. Planen: Skizzen

2.3.6.4 Design Spielauswahl

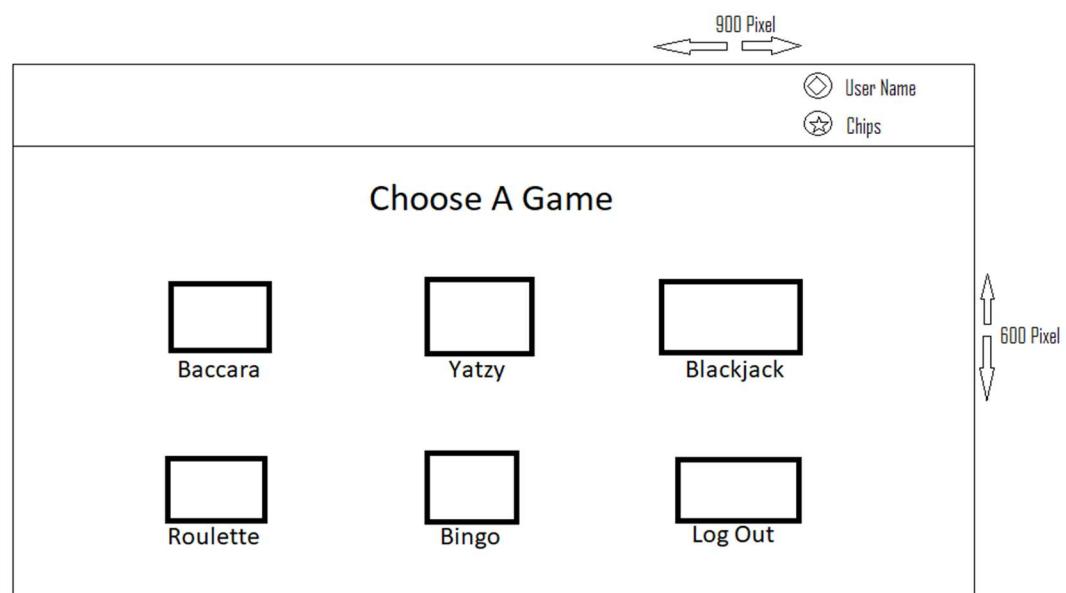
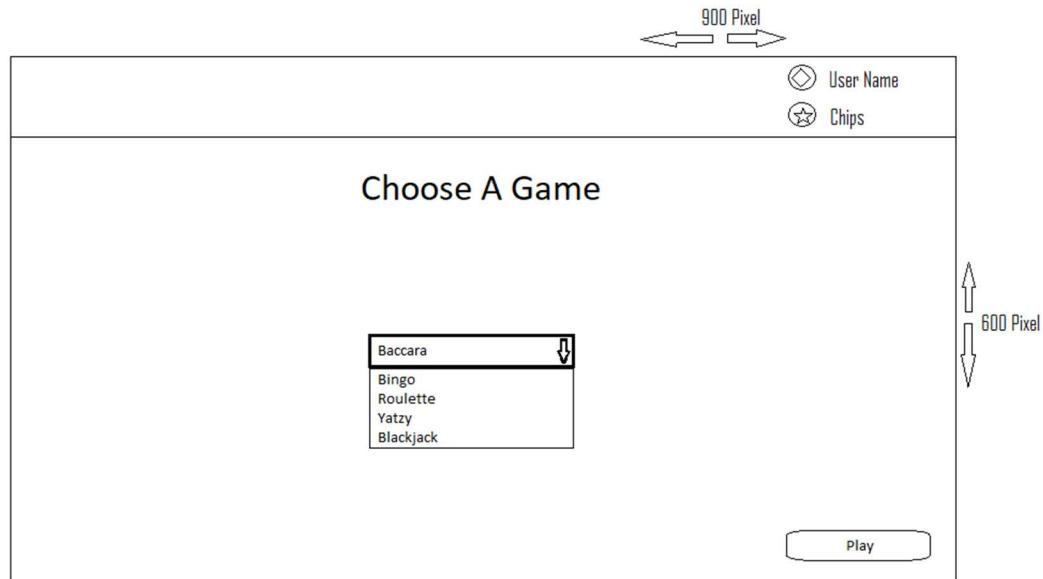


Abbildung 25 und 26: Design Spielauswahl

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spielauswahl Screens zu sehen. Planen: Skizzen

2.3.6.1 Design Baccara

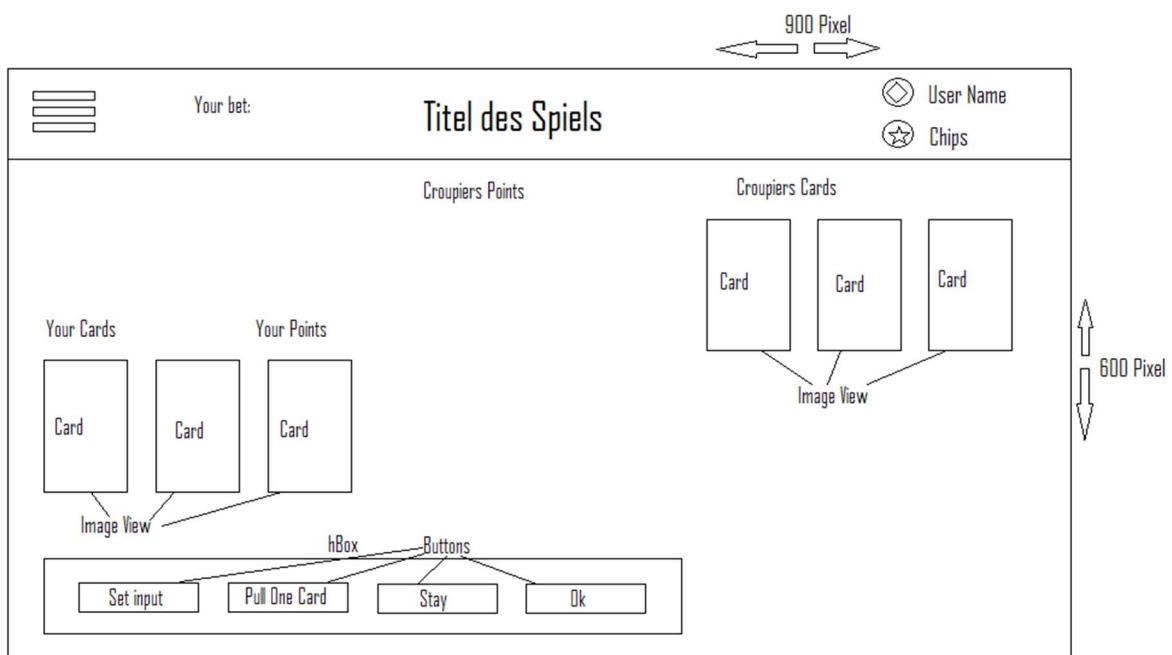
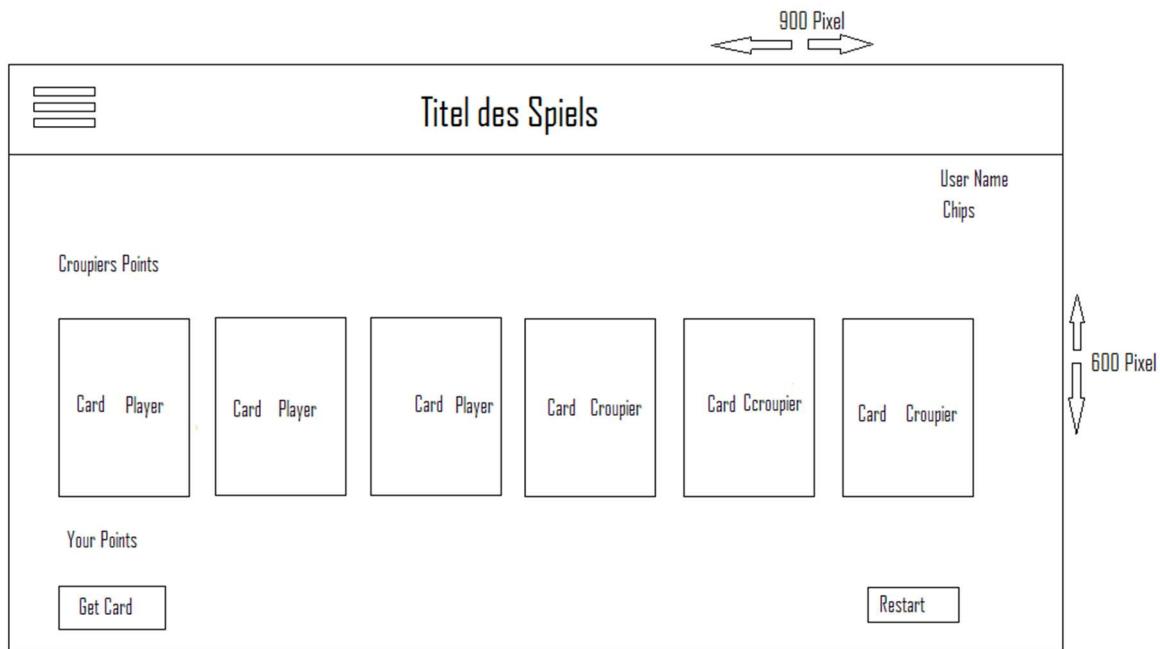


Abbildung 27 und 28: Design Baccara

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spiels Baccara zu sehen. Planen: Skizzen

2.3.6.2 Design Yatzy

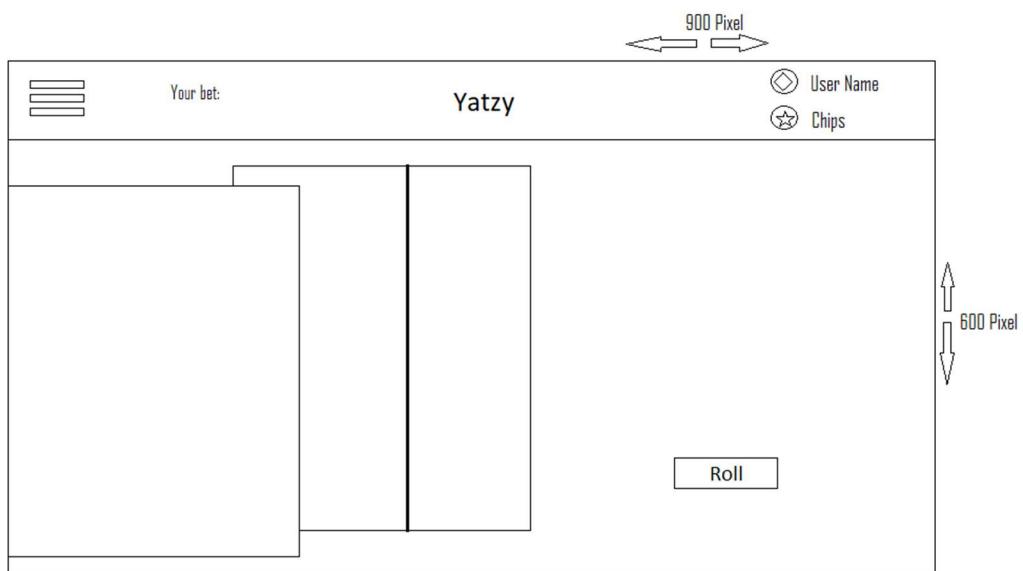
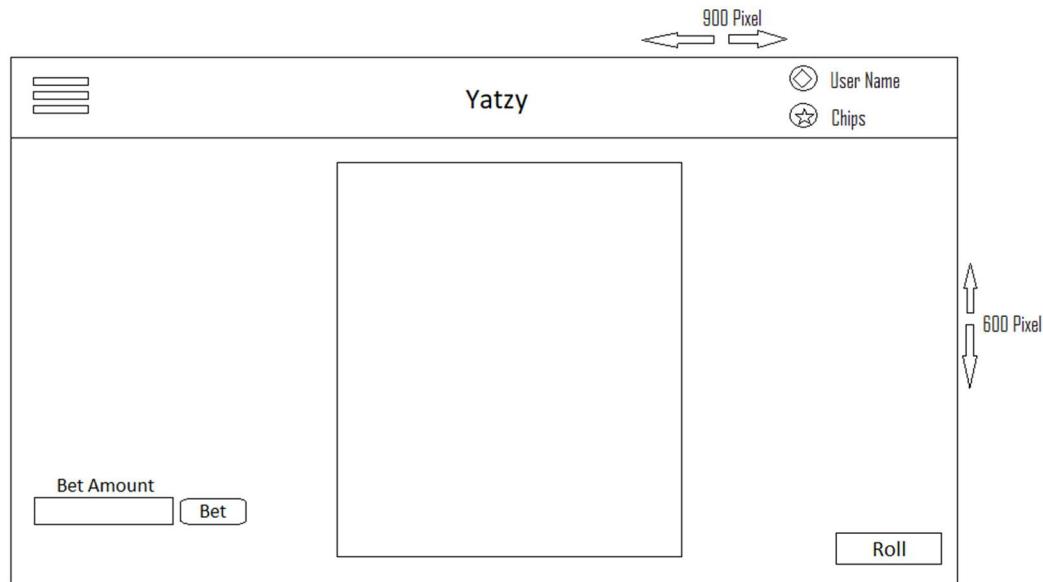


Abbildung 29 und 30: Design Yatzy

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spiels Yatzy zu sehen.
Planen: Skizzen

2.3.6.3 Design Blackjack

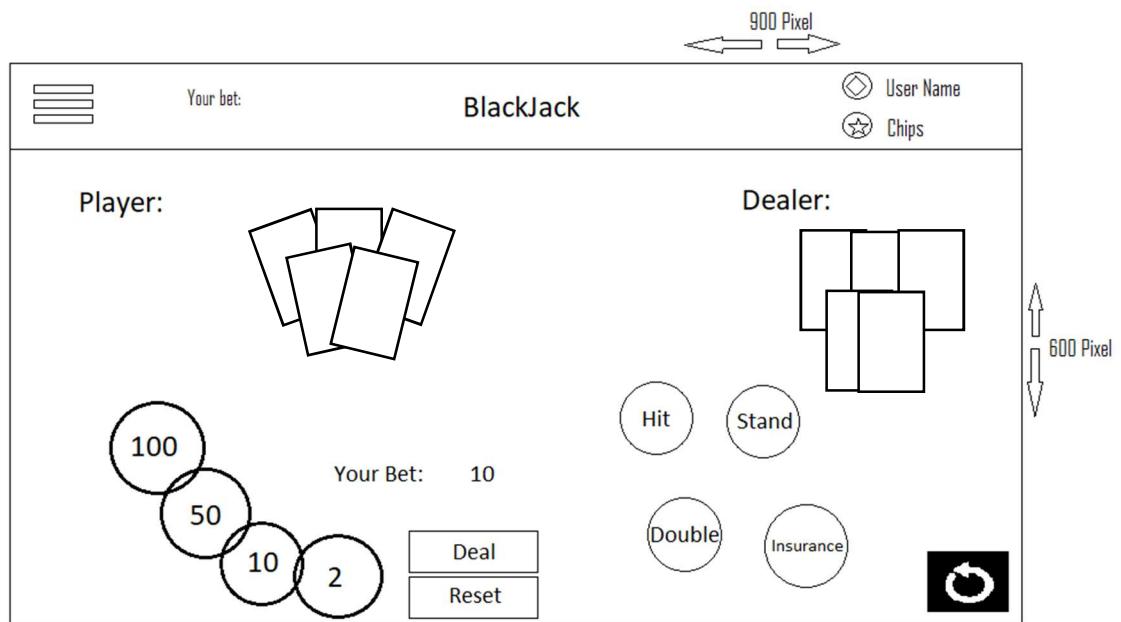
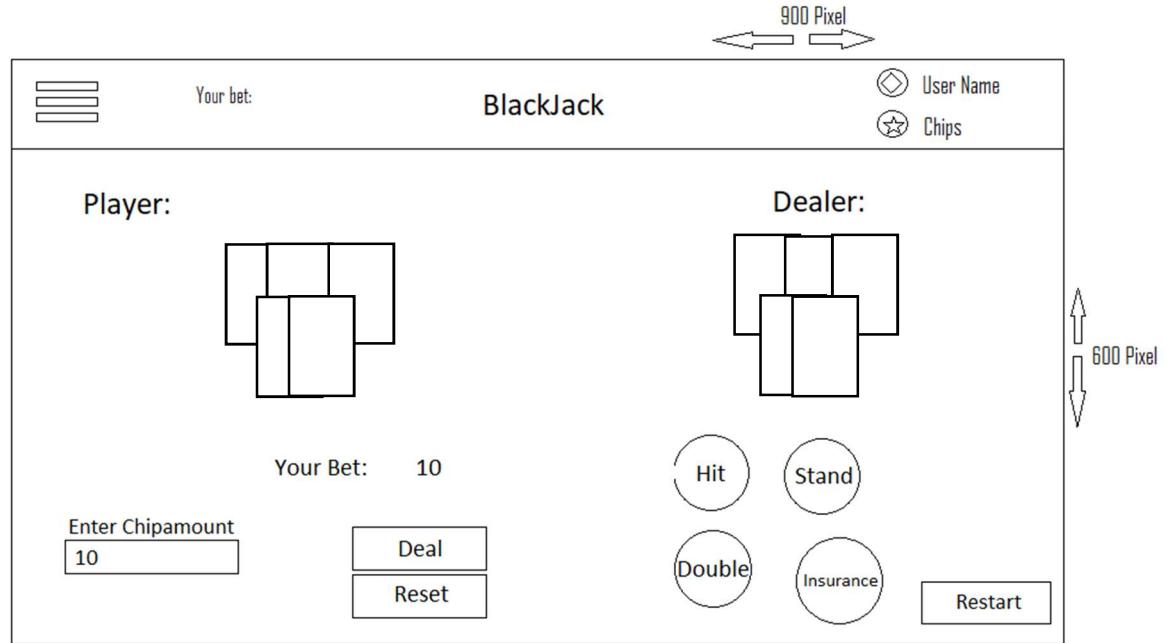


Abbildung 31 und 32: Design Baccara

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spiels Blackjack zu sehen. Planen: Skizzen

2.3.6.4 Design Roulette

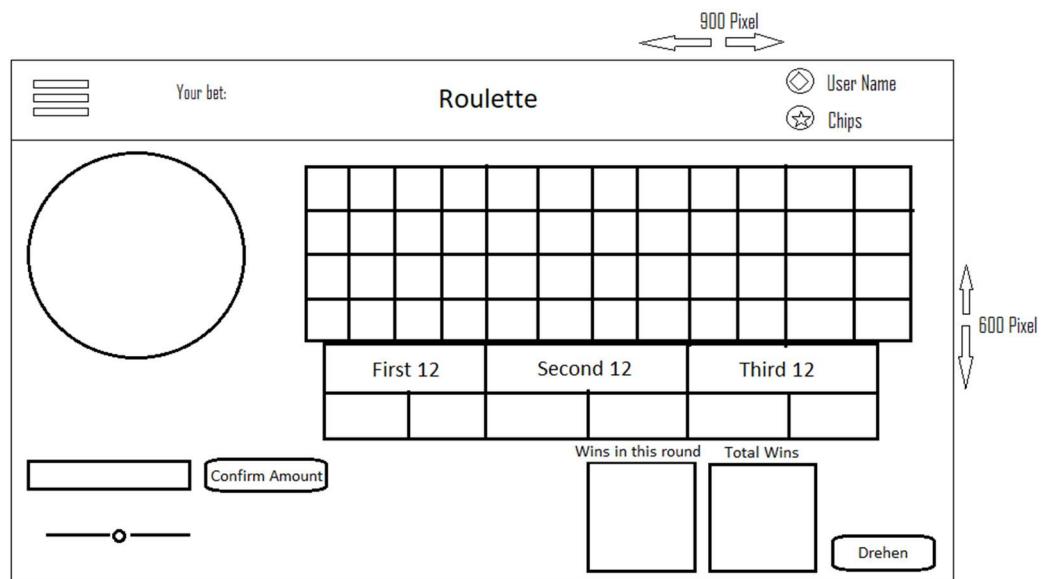
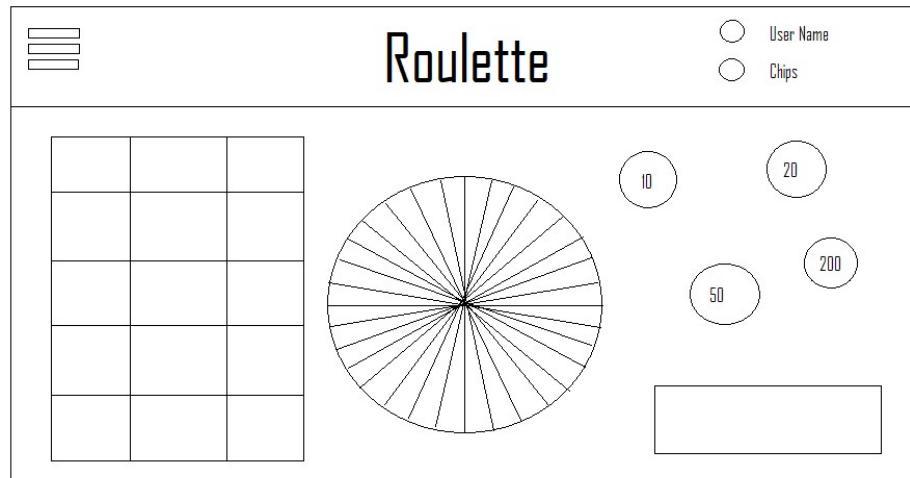


Abbildung 33 und 34: Design Roulette

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spiels Roulette zu sehen. Planen: Skizzen

2.3.6.5 Design Bingo

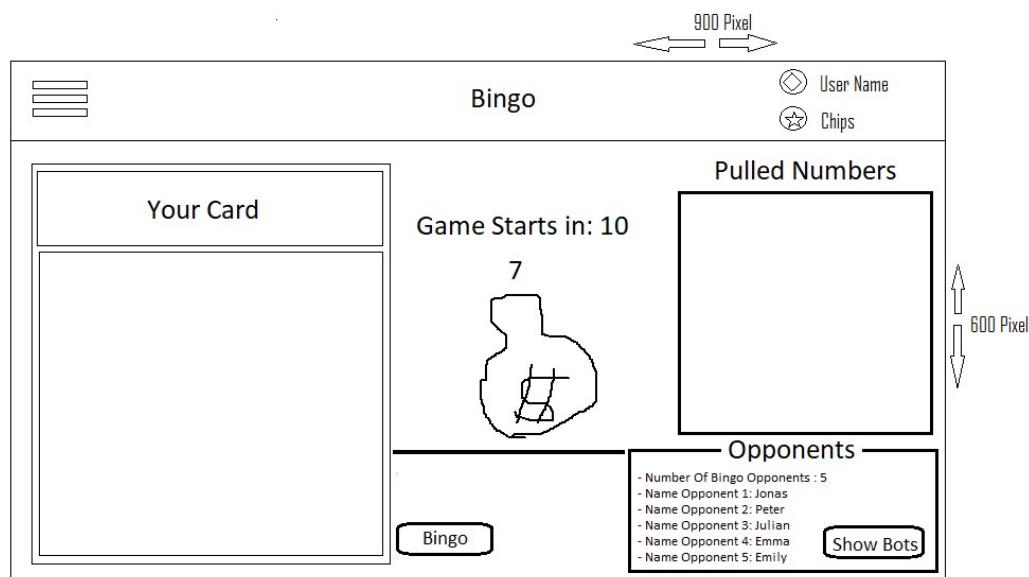
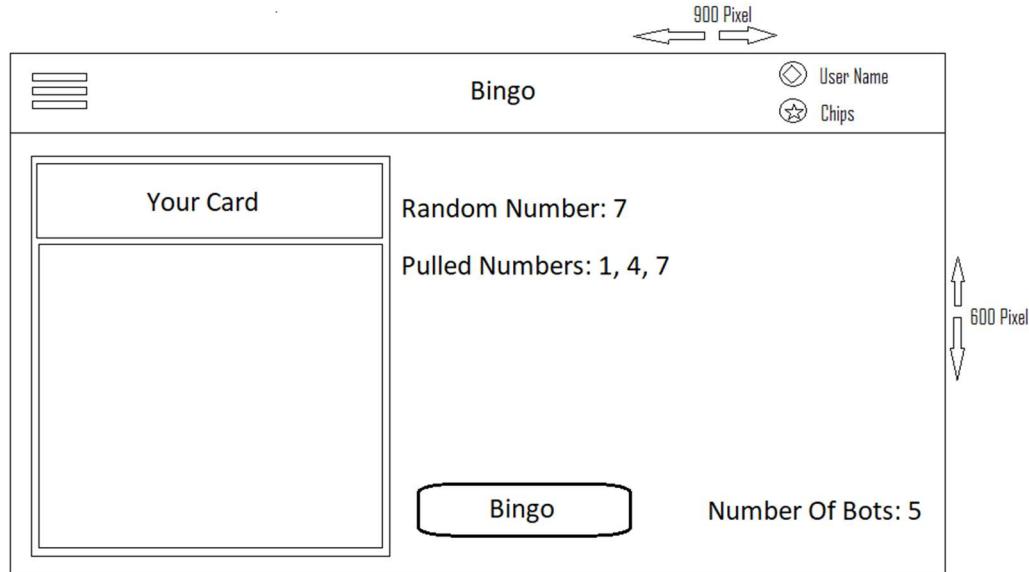


Abbildung 35 und 36: Design Bingo

In diesen zwei Bildern sind die zwei verschiedenen GUI's des Spiels Bingo zu sehen.
Planen: Skizzen

2.3.7 Namenskonventionen

Dateien

Filenamen NameVorname_DateiName

Code

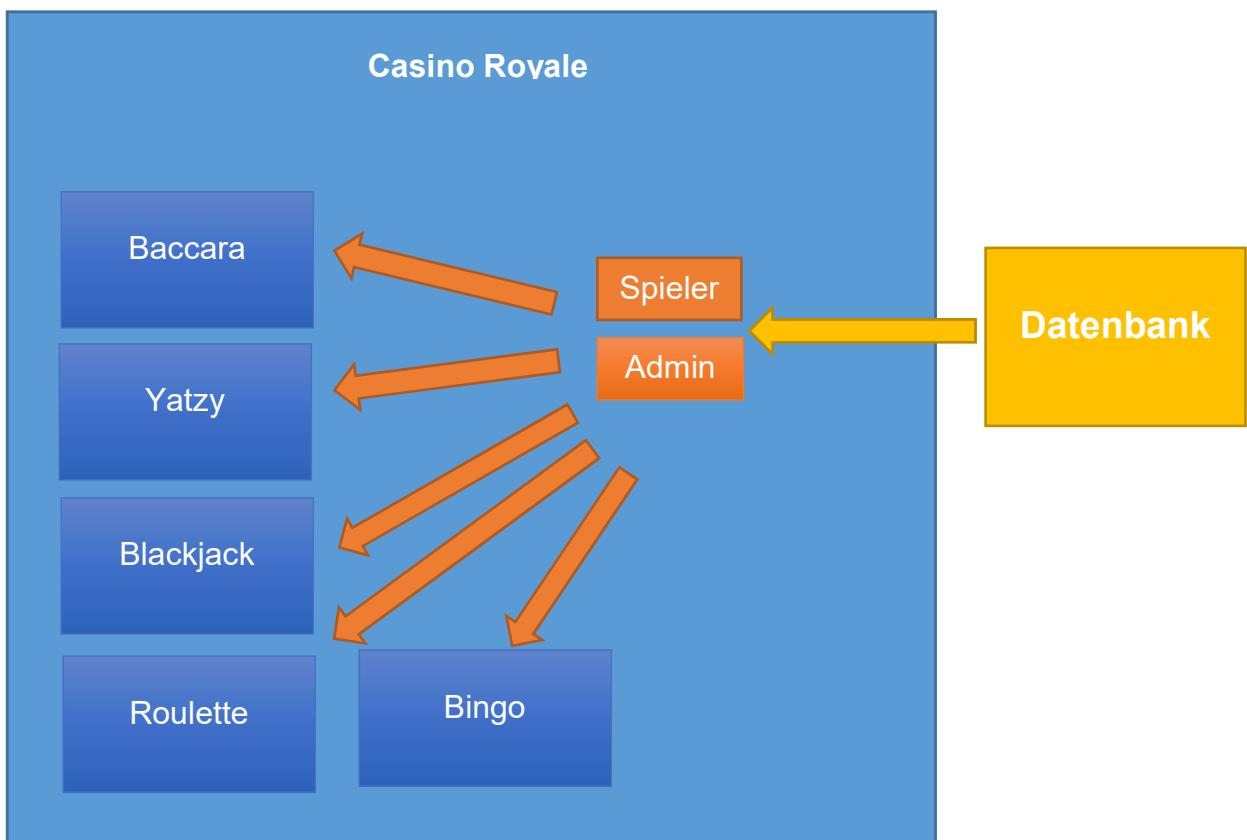
Button	btnFunktion
Normales Label	lblFunktion
Titel Label	lblTitel
Label für Chips	lblChips
TextField	txtFunktion
ImageView	ImageViewFunktion
Slider	sliderFunktion
«Starte Spiel» Button	btnGameName
Vbox	vBoxFunktion

2.3.8 Datenbank

2.3.8.1 Vorgehen

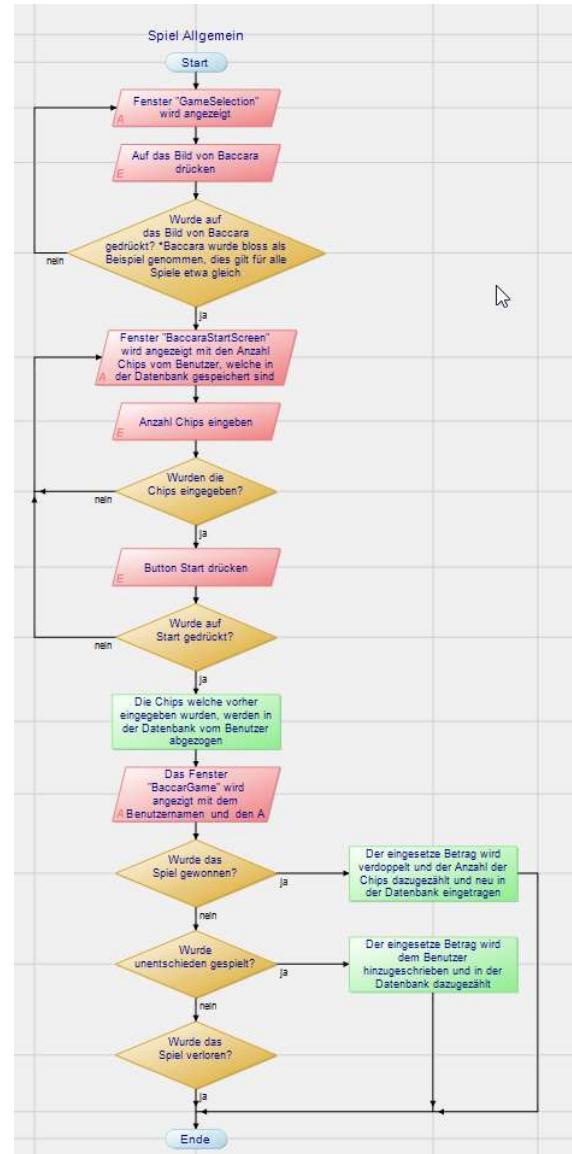
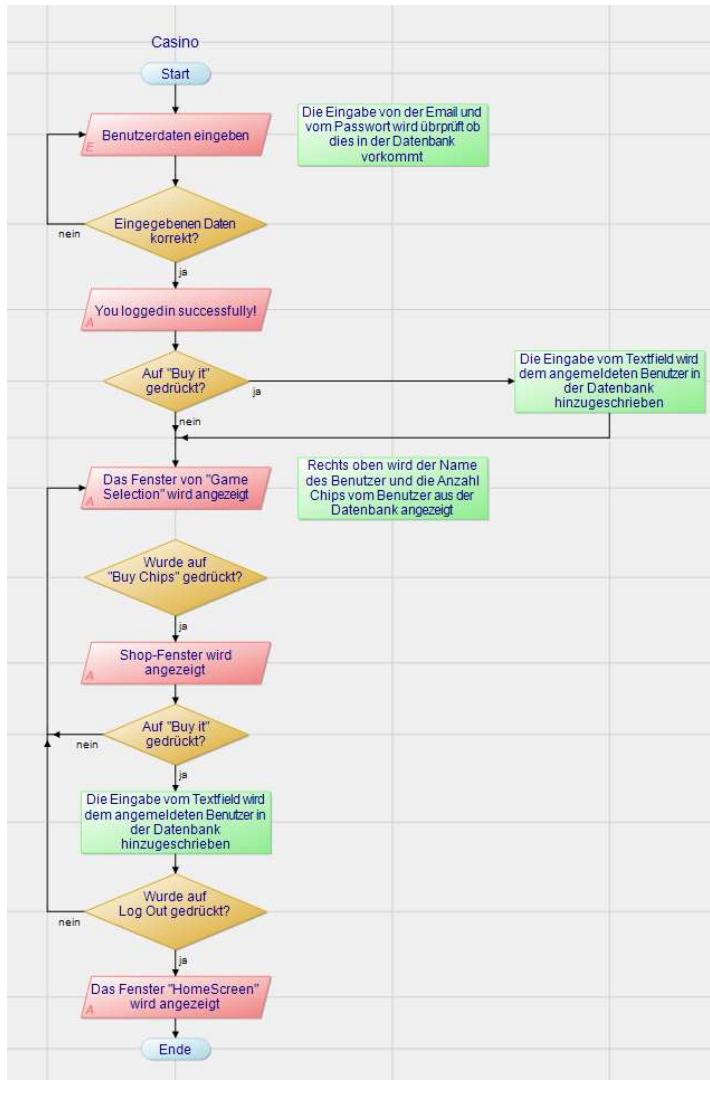
Zuerst wurde das konzeptionelle Datenbankmodell skizziert. Als dies unseren Erwartungen und Vorstellungen entsprach wurde es im Programm «Umlet» digitalisiert. Danach ging es an das logische Datenbankmodell, dieses wurde auch skizziert und anschliessend digitalisiert. Als dies getan war konnte ich die Datenbank auf PhpMyAdmin, auf dem Server von Yannick Huggler einrichten. Beim einrichten der Datenbank wurde das logische Datenbankmodell implementiert.

2.3.8.2 Systemgrenzen / Schnittstellen zur Aussenwelt



Die Datenbank ist auf einem Webserver in der Außenwelt. Sie kommuniziert ständig mit unserem Programm. Es werden Abfragen zu Kontostand und Benutzername getätigt. Zusätzlich überprüft es die Anmelddaten. Alle Gewinne und Verluste des Spielers werden natürlich in der Datenbank gespeichert.

2.3.8.3 Programmablaufplan mit Datenbank



Dieser Programmablaufplan zeigt die möglichen Schritte des Benutzers an und wie sich diese dann auf die Datenbank verhalten. Beim linken PAP wurde das Casino im Allgemeinen dargestellt und auf den rechten Seiten wurde ein Spielverhältnis dargestellt, dies ist nur ein Beispiel und funktioniert bei allen Spielen im Grossen und Ganzen etwa gleich.

Abbildung 37 und 38: PAP mit Datenbank

Das Programm wird in einem PAP dargestellt und auch die Datenbank wird als PAP dargestellt.

2.3.8.4 Konzeptionelles Datenbankmodell

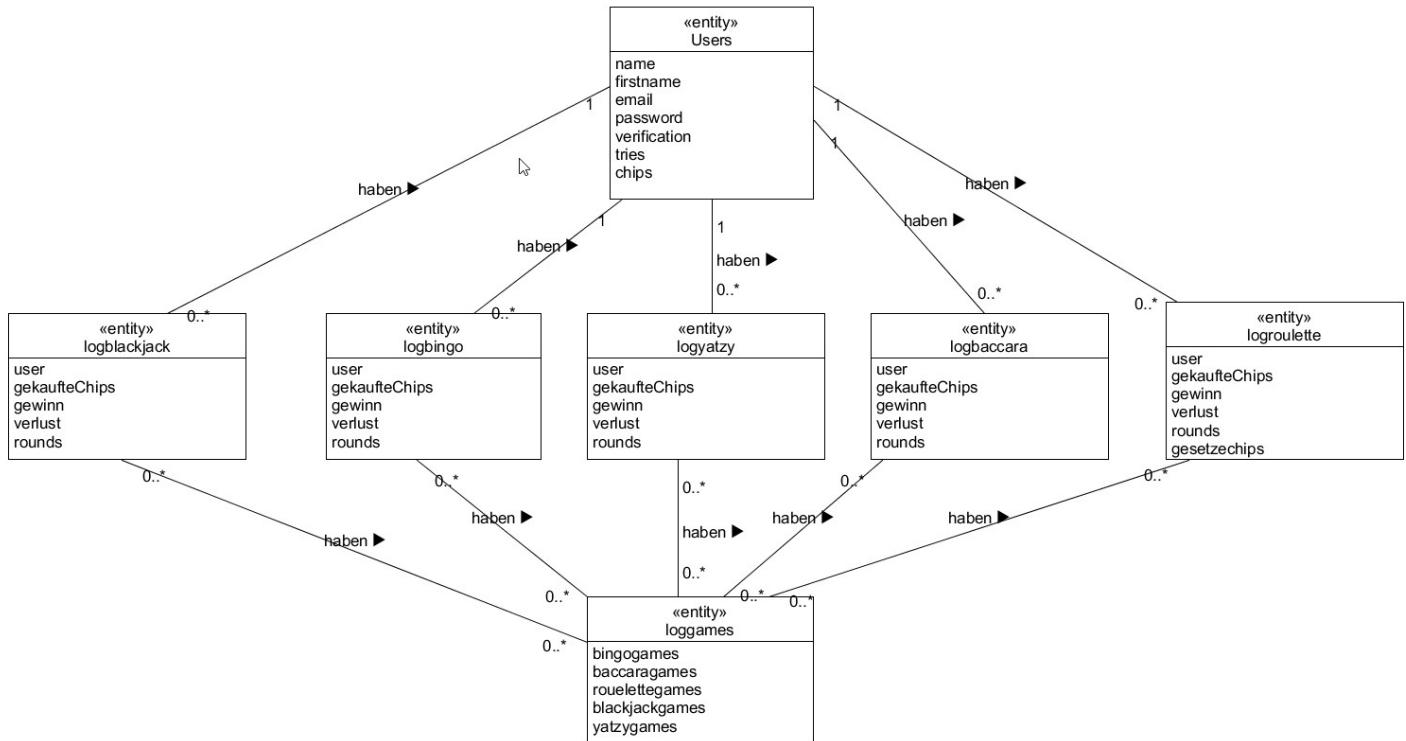


Abbildung 39: Konzeptionelles Datenbankmodell

2.3.8.5 Logisches Datenbankmodell

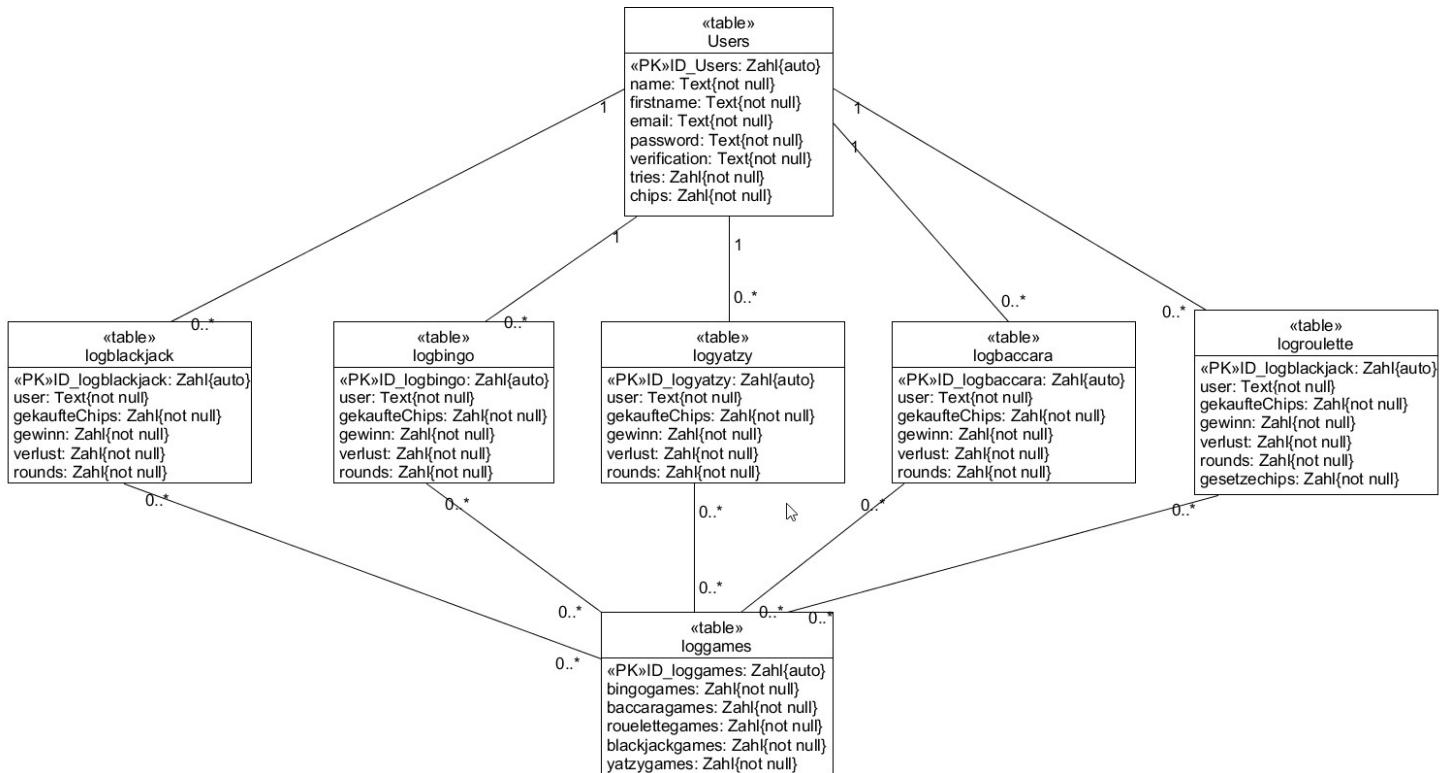


Abbildung 40: Logisches Datenbankmodell

2.3.8.6 Sicherheitsanalyse

```
private void insertIntoSql(Event event) throws SQLException, IOException {
    Statement myStmt = sqlLogin().createStatement();
    String query = "insert into " + sqlGetTable() + " (name, firstname, email, password, verification, tries, chips)"
    + " values (?, ?, ?, ?, ?, ?, ?)";
    PreparedStatement preparedStmt = sqlLogin().prepareStatement(query);
    preparedStmt.setString(1, nameSignUp);
    preparedStmt.setString(2, firstNameSignUp);
    preparedStmt.setString(3, mailSignUp);
    preparedStmt.setString(4, passwordSignUp);
    preparedStmt.setString(5, random);
    preparedStmt.setString(6, "5");
    preparedStmt.setInt(7, 0);
    preparedStmt.execute();
```

Abbildung 41: Code Datenbank (Action)

Für die Sicherheit unserer Benutzer haben wir in Java mit «preparedStatements» gearbeitet. Diese «preparedStatements» verhindern eine SQLInjection, dies ist sehr wichtig, da eine SQLInjection eine ganze Datenbank zerstören kann. Wie eine SQLInjection genau funktioniert wissen wir nicht genau, da wir dieses Thema noch nicht genauer behandelt haben.

	id	name	firstname	email	password	verification	tries	chips
<input type="checkbox"/>	1	Yannick		casion@royale.ch		Verified		3900
<input type="checkbox"/>	2	Eugster	Joel	joel.eugster@stud.bbbaden.ch		Verified	0	
<input type="checkbox"/>	5	admin	admin	admin@admin.ch		Verified	3000	
<input type="checkbox"/>	9	Jelle	Schutter	jelle.schutter@gmx.de		Verified	0	
<input type="checkbox"/>	21	test	user	table@view.ch		Verified	0	
<input type="checkbox"/>	22	Blatter	Hanspeter	y@s.c		Verified	10	
<input type="checkbox"/>	23	test	yannick	yannick10@sunrise.ch		Verified	0	
<input type="checkbox"/>	24	Steinacher Julian		uselesstrash122@gmail.com		Verified	8000000	
<input type="checkbox"/>	28	bbb	demo	joeleugster@hotmail.ch		Verified	1718	
<input type="checkbox"/>	30	test	user	test@user.ch		Locked	0	0
<input type="checkbox"/>	31	test	userzwei	test2@user.ch		721218	2	0

Abbildung 42: Datenbank

Weiter haben wir einen Counter bei der Mailverifizierung erstellt. Dieser ist dafür zuständig, dass der Benutzer nur 5 Möglichkeiten hat seine Mail richtig zu verifizieren. Falls der Verifizierungscode 5 Mal falsch eingegeben wurde wird sich in der Datenbank in der Kolone „Verification“ der Status zu „Locked“ ändern. Dies verhindert Bruteforce-Attacken auf unsere Datenbank und erhöht somit den Sicherheitsfaktor für unsere Benutzer.

Leider konnten wir unsere Passwörter bisher nur als „Plain-Text“ abgespeichert. Dies ist leider nicht sehr sicher und falls unsere Datenbank gehackt wird, würde der Hacker ganz einfach über die Passwörter der Benutzer bescheiden wissen. Dieses Problem könnte man mit einem Salt beheben. So dass das Passwort in einem Hash angezeigt wird und dieser zuerst entschlüsselt werden muss, bevor man das Passwort sehen kann.

2.3.9 Testfallspezifikation

2.3.9.1 Testfallspezifikation Lernende

Testfallspezifikation Lernende an das Programm:

Testfallnummer: AD- 1.1	
Anforderung:	Das Programm sollte eine Help Funktion beinhalten.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Der Benutzer klickt auf den Button «Help».
Ausgabe:	1. Eine Webseite wird aufgerufen mit einer Helpfunktion.

Testfallnummer: AD- 1.2	
Anforderung:	Im Programm sollte man einen frei gewählten Betrag in Chips umwandeln.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Der Benutzer gibt den gewünschten Betrag ein. 2. Der User klickt auf den Button «Buy».
Ausgabe:	1. Die Chips werden dem Account übertragen.

Testfallnummer: AD- 1.3	
Anforderung:	Das Programm sollte keine Kompilierfehler beinhalten.
Voraussetzungen:	NetBeans ist geöffnet.
Eingabe:	1. Quellcode anschauen
Ausgabe:	1. Keine Warnungen → rote «!». Das Programm kann gestartet werden.

Testfallnummer: AD- 1.4	
Anforderung:	Das Programm sollte beim Eingeben des Passwortes, sollte dieses Passwort nicht angezeigt werden.
Voraussetzungen:	Das Programm ist gestartet.
Eingabe:	1. Der Benutzer gibt Benutzernname und Passwort ein.
Ausgabe:	1. Das Passwort wird nicht angezeigt, sondern nur mit Punkten dargestellt.

Testfallnummer: AD- 1.5	
Anforderung:	Das Programm sollte bei allen Spielen ein ausklappbares Menu haben.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User befindet sich in einem Spiel.
Eingabe:	1. Der Benutzer klickt auf das Menüsymbol.
Ausgabe:	1. Das Menu Symbol wird angezeigt. (Quit Game, Exit, Help, Log out)

2.3.9.2 Testfallspezifikation Schule

Testfallspezifikation Schule an das Programm:

Testfallnummer: AD- 2.1	
Anforderung:	Das Programm sollte mit Java realisiert werden.
Voraussetzungen:	NetBeans und Java sind installiert.
Eingabe:	1. Java Programmiersprache
Ausgabe:	1. Programm mit Java realisiert.

Testfallnummer: AD- 2.2	
Anforderung:	Das Programm sollte die CRAP Kriterien erfüllen.
Voraussetzungen:	Die CRAP Methode wird eingeplant. C: Kontrast R: Repetition A: Ausrichtung P: Nähe
Eingabe:	-
Ausgabe:	-

Testfallnummer: AD- 2.3	
Anforderung:	Das Programm sollte eine Anmeldefunktion für Benutzer haben.
Voraussetzungen:	Das Programm ist gestartet. Der Benutzer ist auf der Log-In Seite.
Eingabe:	1. Der Benutzer
Ausgabe:	1. Der Benutzer kann sich Anmelden oder Registrieren.

Testfallnummer: AD- 2.4	
Anforderung:	Das Programm sollte einen frei gewählten Betrag in Jetons umwandeln können.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User ist auf der Seite, wo man Chips kaufen kann.
Eingabe:	1. Der Benutzer gibt den gewünschten Betrag ein. 2. Der User klickt auf den Button «Buy».
Ausgabe:	1. Die Chips werden dem Account übertragen.

Testfallnummer: AD- 2.5	
Anforderung:	Das Programm sollte eine Funktion anbieten, in welcher der Administrator sich anmelden kann, um die Kennzahlen einsehen zu können.
Voraussetzungen:	Das Programm ist gestartet. Der Administrator ist eingeloggt.
Eingabe:	1. Der Administrator klickt auf ein Spiel.
Ausgabe:	1. Er sieht die Kennzahlen der einzelnen Spieler.

s

2.3.9.3 Testfallspezifikation Programm

Testfallspezifikation Benutzer an das Programm:

Testfallnummer: AD- 3.1	
Anforderung:	Ein Benutzer kann sich einen Account erstellen oder Anmelden.
Voraussetzungen:	Das Programm ist gestartet.
Eingabe:	<ol style="list-style-type: none"> 1. Variante <ol style="list-style-type: none"> 1. Der Benutzer gibt Anmelde Daten ein. 2. Der Benutzer klickt auf den Button «Log-In». 2. Variante <ol style="list-style-type: none"> 1. Der Benutzer klickt auf den Button «Sign-Up». 2. Der Benutzer gibt die Daten ein. 3. Der Benutzer klickt «Anmelden».
Ausgabe:	Der Benutzer meldet sich an.

Testfallnummer: AD- 3.2	
Anforderung:	Ein Benutzer kann einen frei gewählten Betrag in Chips umwandeln.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der Benutzer ist angemeldet.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der User gibt den Betrag an Chips ein den er kaufen möchte und klickt dann auf «Buy».
Ausgabe:	<ol style="list-style-type: none"> 1. Die Chips werden dem Account gutgeschrieben.

Testfallnummer: AD- 3.3	
Anforderung:	Ein Benutzer kann Bingo, Baccara, Blackjack, Roulette, Yatzy spielen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der Benutzer ist angemeldet.</p> <p>Der Benutzer ist auf der Seite wo man Spiele auswählen kann.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der User klickt auf ein Spiel
Ausgabe:	<ol style="list-style-type: none"> 1. Die Titelseite des Spiels wird aufgerufen.

Testfallnummer: AD- 3.4	
Anforderung:	Ein Benutzer kann sich Abmelden.
Voraussetzungen:	Das Programm ist gestartet. Der Benutzer ist angemeldet.
Eingabe:	1. Der Benutzer klickt auf «Log-out».
Ausgabe:	1. Der User wird abgemeldet.

Testfallnummer: AD- 3.5	
Anforderung:	Ein Benutzer kann Chips durch das Spielen gewinnen/verlieren.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der Benutzer hat ein Spiel ausgewählt und gestartet.
Eingabe:	1. Der Benutzer verliert/gewinnt ein Spiel.
Ausgabe:	1. Die Chips werden vom Account abgezogen.

Testfallspezifikation Administrator an das Programm:

Testfallnummer: AD- 4.1	
Anforderung:	Ein Administrator kann Anzahl Runden vom jeweiligen Spiel sehen.
Voraussetzungen:	Der Administrator der Datenbank ist eingeloggt.
Eingabe:	1. Der Admin schaut die Datenbank an.
Ausgabe:	1. Die Anzahl Runden vom jeweiligen Spiel werden angezeigt.

Testfallnummer: AD- 4.2	
Anforderung:	Ein Administrator kann die E-Mail-Adresse jedes Spielers ansehen.
Voraussetzungen:	Der Administrator der Datenbank ist eingeloggt.
Eingabe:	1. Der Admin schaut die Datenbank an.
Ausgabe:	1. Die E-Mail-Adresse von jedem Spieler wird angezeigt.

Testfallnummer: AD- 4.3	
Anforderung:	Ein Administrator kann den Kontostand (Chips Anzahl) jedes Spielers ansehen.
Voraussetzungen:	Der Administrator der Datenbank ist eingeloggt.
Eingabe:	1. Der Admin schaut die Datenbank an.
Ausgabe:	1. Die Chipsanzahl von jedem Spieler wird angezeigt.

Testfallnummer: AD- 4.4	
Anforderung:	Ein Administrator kann sehen welcher Spieler am meisten Gewinn gemacht hat.
Voraussetzungen:	Der Administrator der Datenbank ist eingeloggt.
Eingabe:	1. Der Admin schaut die Datenbank an.
Ausgabe:	1. Der Benutzer welcher am meisten Gewinn gemacht hat wird angezeigt.

Testfallnummer: AD- 4.5	
Anforderung:	Ein Administrator kann Sieg und Niederlagen der einzelnen Spieler sehen.
Voraussetzungen:	Das Programm ist gestartet. Der Administrator ist eingeloggt.
Eingabe:	1. Der Administrator klickt auf ein Spiel.
Ausgabe:	1. Er sieht die Siege/Niederlagen der einzelnen Spieler.

2.3.9.4 Testfallspezifikation Desktop Casino-Software

Testfallnummer: AD- 5.1	
Anforderung:	Das Programm muss eine Desktop Casino – Software sein.
Voraussetzungen:	Das Programm ist mit Java als Desktop Casino – Software
Eingabe:	1. Start klicken
Ausgabe:	1. Programm öffnet sich

Testfallnummer: AD- 5.2	
Anforderung:	Die fünf Spiele Blackjack, Roulette, Baccara, Bingo und Yatzy müssen in diesem Programm vorhanden sein.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Der Spieler loggt sich ein
Ausgabe:	1. Die fünf Spiele werden angezeigt.

Testfallnummer: AD- 5.3	
Anforderung:	Eine Statistikfunktion für den Administrator soll das Programm beinhalten.
Voraussetzungen:	Das Programm ist gestartet. Der Admin ist eingeloggt.
Eingabe:	1. Admin meldet sich mit Benutzernamen und Passwort an
Ausgabe:	1. Die Statistikseite wird angezeigt.

Testfallnummer:	AD- 5.4
Anforderung:	Das Programm sollte eine grafische Spieloberfläche haben.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Einloggen
Ausgabe:	1. Grafische Spieloberfläche wird angezeigt.

Testfallnummer:	AD- 5.5
Anforderung:	Das Programm sollte in der Programmiersprache Java realisiert werden.
Voraussetzungen:	1. Java JDK installiert.
Eingabe:	-
Ausgabe:	-

Testfallnummer:	AD- 5.6
Anforderung:	Das Programm sollte eine Registrierfunktion haben, in welcher der Nutzer sich mit Name und Passwort registrieren kann.
Voraussetzungen:	Das Programm ist gestartet. Der Benutzer klickt auf den Button «sign up»
Eingabe:	1. Eingabe von Namen und Passwort
Ausgabe:	1. Mail Verifikation

Testfallnummer:	AD- 5.7
Anforderung:	Die fünf Spiele Blackjack, Roulette, Baccara, Bingo und Yatzy müssen auswählbar sein.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele auswählen kann.
Eingabe:	1. Einloggen
Ausgabe:	1. Der User sieht die 5 Spiele und kann sie auswählen.

Testfallnummer: AD- 5.8	
Anforderung:	Der Nutzer muss nach der Anmeldung einen frei gewählten Betrag in Jetons umwandeln können.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Der Benutzer gibt ein, wie viele Chips er kaufen möchte. 2. Der Spieler klickt den Button «Buy»
Ausgabe:	1. Die Chips werden dem Account gutgeschrieben.

Testfallnummer: AD- 5.9	
Anforderung:	Der Nutzer kann die Jetons in den Spielen benutzen.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele auswählen kann. Der Benutzer klickt auf ein Spiel. Der Benutzer ist auf der Übersichtsseite.
Eingabe:	1. Der Spieler setzt sein Einsatz
Ausgabe:	1. Jetons werden gesetzt und dem Account abgezogen.

Testfallnummer: AD- 5.10	
Anforderung:	Der Spieler kann nach der Auswahl des Spiels, das Spiel spielen.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele auswählen kann. Der Benutzer klickt auf das Spiel Baccara. Der Benutzer ist auf der Übersichtsseite.
Eingabe:	1. Der Spieler muss auf einem gewünschten Spiel klicken.
Ausgabe:	1. Der Spieler wird zur Spiel weitergeleitet.

Testfallnummer: AD- 5.11	
Anforderung:	Der Spieler kann nach persönlichem Abschluss des Spiels zur Spielauswahl zurückkehren und sich neu entscheiden oder aufhören.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ein Spiel beendet.
Eingabe:	1. Der Spieler klickt auf den Button «restart».
Ausgabe:	1. Das Spiel startet neu.

Testfallnummer: AD- 5.12	
Anforderung:	Die Einsätze des Spielers erfolgen in Abhängigkeit der jeweiligen Spielregeln der Spiele.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele auswählen kann. Der Benutzer klickt auf das ein Spiel. Der Benutzer ist auf der Übersichtsseite.
Eingabe:	1. Eingabe von Einsatz
Ausgabe:	1. Einsatz wird nach Spielregeln gesetzt.

Testfallnummer: AD- 5.13	
Anforderung:	Für einen Administrator besteht die Möglichkeit, dass dieser nach erfolgreicher Anmeldung Kennzahlen einsehen kann.
Voraussetzungen:	Das Programm ist gestartet. Der Administrator ist eingeloggt.
Eingabe:	1. Der Administrator klickt auf ein Spiel.
Ausgabe:	1. Der Administrator sieht die Kennzahlen.

Testfallnummer:	AD- 5.14
Anforderung:	Der Administrator kann nach der Anmeldung von jedem Spieler die Werte zu Einsatz, Gewinn und Verlust anschauen können.
Voraussetzungen:	Das Programm ist gestartet. Der Administrator ist eingeloggt.
Eingabe:	1. Der Administrator klickt auf ein Spiel.
Ausgabe:	1. Der Administrator sieht die Werte von jedem Spieler zu Einsatz, Gewinn und Verlust

Testfallnummer:	AD- 5.15
Anforderung:	Die ermittelten Werte der Spiele sollen ebenfalls dokumentiert und dem Administrator dargestellt werden.
Voraussetzungen:	Das Programm ist gestartet. Der Administrator ist eingeloggt.
Eingabe:	1. Der Administrator klickt auf ein Spiel.
Ausgabe:	1. Die ermittelten Werte der Spiele werden angezeigt.

Testfallnummer:	AD- 5.16
Anforderung:	Jeder Lernender muss ein Spiel der Glückspiele selber realisiert haben.
Voraussetzungen:	Das Programm ist gestartet.
Eingabe:	-
Ausgabe:	-

Testfallnummer:	AD- 5.17
Anforderung:	Die Realisierung der übergeordneten Casino-Software muss in Gruppenarbeit realisiert werden.
Voraussetzungen:	-
Eingabe:	-
Ausgabe:	-

Testfallnummer: AD- 5.18	
Anforderung:	Die Administrationsfunktion erfolgt in Gruppenarbeit durch das Projektteam.
Voraussetzungen:	-
Eingabe:	-
Ausgabe:	-

Testfallnummer: AD- 5.19	
Anforderung:	In der Casino-Software darf ein Spiel nur einmal vorkommen.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt.
Eingabe:	1. Der User loggt sich ein
Ausgabe:	1. Der User kann nur 1 Spiel von jedem auswählen.

2.3.9.5 Testfallspezifikation Baccara

Testfallnummer: AD- 6.1	
Anforderung:	Das Spiel enthält eine Titelseite.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Baccara
Ausgabe:	1. Titelseite des Spiels wird angezeigt.

Testfallnummer: AD- 6.2	
Anforderung:	Die Titelseite enthält den Namen des Spiels.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Baccara
Ausgabe:	1. Auf der Titelseite wird der Namen des Spiels angezeigt.

Testfallnummer:	AD- 6.3
Anforderung:	Die Titelseite enthält den aktuellen Kontostand.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Baccara.
Ausgabe:	1. Auf der Titelseite wird der aktuelle Kontostand anzeigen.

Testfallnummer:	AD- 6.4
Anforderung:	Wenn ich auf den Startbutton klicke auf der Titelseite, dann startet das Spiel
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Baccara.
Ausgabe:	1. Ich komme zur nächsten Seite. Das Spiel startet.

Testfallnummer: AD- 6.5	
Anforderung:	Die gezogenen Karten des Spielers werden graphisch dargestellt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start» gedrückt.</p>
Eingabe:	1. Auf der Spielseite den Button «Set Input» drücken.
Ausgabe:	1. Ich sehe die Karten des Spielers graphisch.

Testfallnummer: AD- 6.6	
Anforderung:	Die gezogenen Karten des Croupiers werden graphisch dargestellt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start» gedrückt.</p>
Eingabe:	1. Auf der Spielseite den Button «Set Input» drücken.
Ausgabe:	1. Ich komme zur nächsten Seite und die Karten des Spielers werden graphisch dargestellt.

Testfallnummer: AD- 6.7	
Anforderung:	Während dem Spiel wird der gesetzte Betrag angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start» gedrückt.</p>
Eingabe:	1. Auf der Spielseite den Button «Set Input» drücken.
Ausgabe:	1. Im Spiel wird oben links der gesetzte Betrag angezeigt.

Testfallnummer: AD- 6.8	
Anforderung:	Der Aktuelle Kontostand wird im Spiel angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start» gedrückt.</p>
Eingabe:	1. Auf der Spielseite den Button «Set Input» drücken.
Ausgabe:	1. Der Aktuelle Kontostand wird im Spiel oben rechts angezeigt.

Testfallnummer:	AD- 6.9
Anforderung:	Wenn es die Regeln erlauben, kann ich entscheiden, ob ich eine Karte ziehe oder nicht
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start» gedrückt.</p>
Eingabe:	1. Auf der Spielseite den Button «Set Input» drücken.
Ausgabe:	1. Die ersten beiden Karten werden gezogen.

Testfallnummer:	AD- 6.10
Anforderung:	Ich kann den zu setzenden Betrag selber wählen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer kann einen Slider verwenden oder das Textfield.
Ausgabe:	1. Betrag wird gesetzt

Testfallnummer:	AD- 6.11
Anforderung:	Der Betrag wird meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	2. Betrag setzen und «Start» drücken.
Ausgabe:	Der Betrag wird meinem Konto abgezogen.

Testfallnummer:	AD- 6.12
Anforderung:	Der gesetzte Betrag muss ganzzahlig sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	Ein Betrag grösser als 0 kleiner als Bankguthaben und int.
Ausgabe:	Ganzzahliger Betrag wird abgezogen

Testfallnummer:	AD- 6.13
Anforderung:	Der Betrag muss mindestens 10 Coins betragen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	Eingabe von Betrag und «Start» drücken.
Ausgabe:	Meldung, wenn Betrag nicht mind. 10 ist. Sonst weiterspielen

Testfallnummer: AD- 6.14	
Anforderung:	Der gesetzte Betrag muss kleiner als das Bankguthaben sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer setzt den Betrag.</p>
Eingabe:	1. «Start» drücken
Ausgabe:	1. Wenn Betrag < 0, dann Meldung sonst weiter zu Spielbeginn.

Testfallnummer: AD- 6.15	
Anforderung:	Der Spieler kann die ersten beiden Karten ziehen
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer setzt den Betrag.</p> <p>Der Benutzer drückt «Start»</p>
Eingabe:	1. Der User klickt «Set input»
Ausgabe:	1.

Testfallnummer: AD- 6.16	
Anforderung:	Das Spiel gibt eine Meldung an, wenn ich verloren habe.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel fertig gespielt
Ausgabe:	1. Image View zeigt Bild, dass es verloren ist.

Testfallnummer: AD- 6.17	
Anforderung:	Das Spiel gibt eine Meldung an, wenn ich gewonnen habe.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel fertig gespielt
Ausgabe:	1. Image View zeigt Bild, dass es gewonnen ist.

Testfallnummer: AD- 6.18	
Anforderung:	Alle Karten kommen aus einem gemischten, endlichen Kartenstapel
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Karten ziehen
Ausgabe:	1. Karten anzeigen

Testfallnummer: AD- 6.19	
Anforderung:	Wenn es die Regeln erlauben kann der Spieler ein weiteres Karten ziehen.-
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Auf den Button «Pull Card» drücken
Ausgabe:	1. Dritte Karte wird gezogen.

Testfallnummer: AD- 6.20	
Anforderung:	Wenn ich mit Black Jack gewinne, dann wird mein gesetzter Betrag zu 3:2 ausbezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p> <p>Der Benutzer gewinnt mit Black Jack.</p>
Eingabe:	1. Spieler gewinnt mit Black Jack.
Ausgabe:	1. Betrag wird zu 3:2 abgezogen vom Bankkonto.

Testfallnummer: AD- 6.21	
Anforderung:	Wenn ich einfach so gewinne, dann wird mir mein gesetzter Betrag verdoppelt ausbezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p> <p>Der Benutzer gewinnt ohne Black Jack.</p>
Eingabe:	Spieler gewinnt
Ausgabe:	Betrag wird dem Konto gutgeschrieben.

Testfallnummer: AD- 6.22	
Anforderung:	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel gewonnen
Ausgabe:	1. Der Kontostand und gewonnene Betrag werden angezeigt.

Testfallnummer: AD- 6.23	
Anforderung:	Bei einem Unentschieden wird eine Meldung angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel fertig gespielt
Ausgabe:	1. Image View zeigt Bild, dass es unentschieden ist.

Testfallnummer: AD- 6.24	
Anforderung:	Beim Unentschieden wird der eingesetzte Betrag wieder aufs Bankkonto gezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel unentschieden
Ausgabe:	1. Der Betrag wird wieder dem Konto gutgeschrieben.

Testfallnummer: AD- 6.25	
Anforderung:	Das Spiel kann neu gestartet werden, indem ich eine Meldung starte.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Den Knopf «restart» klicken.
Ausgabe:	1. Spiel wird neu gestartet, man ist wieder auf Titelseite.

Testfallnummer: AD- 6.26	
Anforderung:	Wenn ich verliere wird der Betrag meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Spiel verloren
Ausgabe:	1. Betrag wurde vom Konto abgezogen

Testfallnummer: AD- 6.27	
Anforderung:	Ich kann das Spiel beliebig abbrechen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Den Button Exit, Logout oder quit game klicken
Ausgabe:	1. Spiel wird abgebrochen

Testfallnummer: AD- 6.28	
Anforderung:	Wenn ich das Spiel abbreche, dann wird mein gesetzter Betrag von meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Baccara.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Den Button Exit, Logout oder quit game klicken
Ausgabe:	1. Spiel wird abgebrochen und Bankguthaben angepasst.

2.3.9.6 Testfallspezifikation Yatzy

Testfallnummer: AD- 7.1	
Anforderung:	Es ist eine Übersichtseite vorhanden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite</p>
Eingabe:	1.Der Benutzer klickt auf das Spiel Yatzy.
Ausgabe:	1.Übersichtsseite des Spiels wird angezeigt.

Testfallnummer: AD- 7.2	
Anforderung:	Die Übersichtsseite enthält den Titel des Spiels.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite</p>
Eingabe:	1.Der Benutzer klickt auf das Spiel Yatzy.
Ausgabe:	1.Auf der Übersichtsseite wird der Namen des Spiels angezeigt.

Testfallnummer: AD- 7.3	
Anforderung:	Die Übersichtsseite enthält den aktuellen Kontostand.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1.Der Benutzer klickt auf das Spiel Yatzy.
Ausgabe:	1.Auf der Titelseite wird der aktuelle Kontostand anzeigen.

Testfallnummer: AD- 7.4	
Anforderung:	Mit Klick auf einen Button wird das Spiel gestartet.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Yatzy
Ausgabe:	1.Ich komme zur nächsten Seite. Das Spiel wird gestartet.

Testfallnummer: AD- 7.5	
Anforderung:	Es kann ein Betrag ausgewählt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der Benutzer kann einen Slider verwenden oder das Textfield.
Ausgabe:	1. Betrag wird gesetzt

Testfallnummer: AD- 7.6	
Anforderung:	Der Betrag wird meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Betrag gesetzt und hat «Start» gedrückt.</p>
Eingabe:	1. Betrag setzen 10 und «Start» drücken.
Ausgabe:	1. Der Betrag wurde abgezogen.

Testfallnummer: AD- 7.7	
Anforderung:	Der Betrag muss ganzzahlig sein.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele. Der Benutzer klickt auf das Yatzy. Der Benutzer ist auf der Übersichtsseite.
Eingabe:	1. Betrag setzen, der 1.5 beträgt.
Ausgabe:	1. Der Betrag muss ganzzahlig sein.

Testfallnummer: AD- 7.8	
Anforderung:	Der Betrag muss grösser als Null sein.
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele. Der Benutzer klickt auf das Yatzy. Der Benutzer ist auf der Übersichtsseite.
Eingabe:	1. Man gibt einen Betrag von -1 ein.
Ausgabe:	1. Der Betrag muss über Null sein

Testfallnummer: AD- 7.9	
Anforderung:	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Yatzy.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Man gibt einen Betrag grösser als sein Guthaben ein.
Ausgabe:	1. Der Betrag darf nicht grösser sein als das Guthaben.

Testfallnummer: AD- 7.10	
Anforderung:	Die möglichen Figuren werden dargestellt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	1. Der Benutzer würfelt in dem er auf den Button «Roll» klickt und wählt Würfel aus.
Ausgabe:	1. Die möglichen Figuren für die ausgewählten Würfel werden angezeigt.

Testfallnummer: AD- 7.11	
Anforderung:	Die Punkte der bereits gewählten Figuren werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	1. Der Benutzer wählt eine Figur.
Ausgabe:	1. Die Punkte werden angezeigt.

Testfallnummer: AD- 7.12	
Anforderung:	Die Anzahl verbleibende Würfe werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	1. Der Benutzer würfelt in dem er auf den Button «Roll» klickt.
Ausgabe:	1. Die würfe die noch übrig sind werden angezeigt.

Testfallnummer: AD- 7.13	
Anforderung:	Die Subtotale, der Bonus und die Gesamtanzahl der Punkte werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist Fertiggespielt.</p>
Eingabe:	1.
Ausgabe:	1. Die Subtotale, der Bonus und die Gesamtzahl wird dargestellt.

Testfallnummer: AD- 7.14	
Anforderung:	Die selektierten und gewürfelten Würfel werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Es wurde gewürfelt.</p>
Eingabe:	1. Der Benutzer wählt einen Würfel.
Ausgabe:	1. Der gewählte Würfel wird markiert.

Testfallnummer: AD- 7.15	
Anforderung:	Die selektierten Würfel werden nicht neu gewürfelt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Es wurde gewürfelt.</p> <p>Ein Würfel wurde ausgewählt.</p>
Eingabe:	1. Der Benutzer würfelt in dem er auf den Button «Roll» klickt.
Ausgabe:	1. Der ausgewählte Würfel bleibt bestehen.

Testfallnummer: AD- 7.16	
Anforderung:	Die neu gewürfelten Würfel werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	1. Der Benutzer würfelt in dem er auf den Button «Roll» klickt.
Ausgabe:	1. Es sind neue Würfel dargestellt.

Testfallnummer:	AD- 7.17
Anforderung:	Die Punkte der selektierten Würfel werden bei der Kategorie angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Es wurde gewürfelt.</p> <p>Würfel wurden ausgewählt.</p>
Eingabe:	1.
Ausgabe:	1. Auf dem Spielblatt werden die möglichen Punkte angezeigt.

Testfallnummer:	AD- 7.18
Anforderung:	Die entsprechende Kategorie wird gesperrt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Es wurde gewürfelt.</p> <p>Würfel wurden ausgewählt.</p> <p>Eine Figur wurde benutzt.</p>
Eingabe:	1. Man klickt auf eine Figur, die schon benutzt wird.
Ausgabe:	1. Diese Figur wurde bereits ausgewählt.

Testfallnummer: AD- 7.19	
Anforderung:	Meine Runde wird beendet und der nächste Spieler kommt dran.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Eine Runde wurde gespielt.</p>
Eingabe:	1. Man wählt eine Figur.
Ausgabe:	1. Der nächste Spieler kommt dran.

Testfallnummer: AD- 7.20	
Anforderung:	Gewinnt der Spieler, so wird ihm der eingesetzte Betrag verdoppelt und ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist fertiggespielt.</p> <p>Der Benutzer hat gewonnen.</p>
Eingabe:	
Ausgabe:	1. Er bekommt den doppelten Betrag zurück.

Testfallnummer: AD- 7.21	
Anforderung:	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist fertiggespielt und der Benutzer hat gewonnen.</p>
Eingabe:	1.
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn das Spiel mit Yatzy gewonnen wurde, wird ein Text angezeigt, in welcher steht: «Sie haben gewonnen». 2. Der gewonnene Betrag wird unten rechts angezeigt und der Kontostand oben rechts wird aktualisiert

Testfallnummer: AD- 7.22	
Anforderung:	Beim Unentschieden erscheint eine Meldung.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist fertiggespielt und es ist unentschieden.</p>
Eingabe:	1.
Ausgabe:	1. Es erscheint eine Meldung: «Es ist unentschieden».

Testfallnummer: AD- 7.23	
Anforderung:	Beim Unentschieden wird der eingesetzte Betrag wieder auf das Konto gezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist fertiggespielt und es ist unentschieden.</p> <p>Die Meldung ist erschienen.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Die Meldung muss mit einem Klick auf «OK» bestätigt werden.
Ausgabe:	<ol style="list-style-type: none"> 1. Der eingesetzte Betrag wieder auf das Konto gezahlt. 2. Der Kontostand oben rechts wird aktualisiert.

Testfallnummer: AD- 7.24	
Anforderung:	Beim Bestätigen der Meldung kann das Spiel neu gestartet werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Eine Meldung ist erschienen.</p>
Eingabe:	1. Man Klickt auf «Quit Game» oben links wenn man die drei Striche gedrückt hat und startet dann das Spiel nochmals.
Ausgabe:	1. Das Spiel wird neugestartet.

Testfallnummer: AD- 7.25	
Anforderung:	Beim Verlieren des Spiels wird der eingesetzte Betrag gelöscht.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p> <p>Das Spiel ist fertiggespielt und der Benutzer hat verloren.</p>
Eingabe:	1.
Ausgabe:	1. Ihr eingesetzter Betrag wird gelöscht. 2. Der Kontostand oben rechts wird aktualisiert.

Testfallnummer: AD- 7.26	
Anforderung:	Das Spiel kann jederzeit abgebrochen werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	<ol style="list-style-type: none"> Der Benutzer drückt auf dem Menu Bar oben links. In dem Menu Bar drückt der Benutzer auf «Quit Game».
Ausgabe:	1. Das Spiel wird abgebrochen.

Testfallnummer: AD- 7.27	
Anforderung:	Der gesetzte Betrag geht beim Abbruch verloren.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Yatzy.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Yatzy.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler bricht das Spiel ab in dem er oben Links auf die drei Striche drückt und dann «Quit Game» anklickt.
Ausgabe:	1. Der eingesetzte Betrag geht verloren.

2.3.9.7 Testfallspezifikation Blackjack

Testfallnummer: AD- 8.1	
Anforderung:	Das Spiel hat eine Übersichtseite, welches den Titel des Spiels und den aktuellen Kontostand anzeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtseite.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Die Übersichtsseite hat einen Titel: Blackjack. 2. Die Übersichtsseite zeigt den aktuellen Kontostand.

Testfallnummer: AD- 8.2	
Anforderung:	Die Übersichtsseite sollte ein Button beinhalten, mit dem das Spiel gestartet werden kann.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der Benutzer ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer geht auf der Übersichtseite.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der Benutzer muss auf den Button «Start Game» klicken.
Ausgabe:	<ol style="list-style-type: none"> 1. Die Übersichtsseite enthält einen Button. Wenn man auf diesen klickt kommt man auf das eigentliche Spiel Blackjack.

Testfallnummer: AD- 8.3	
Anforderung:	Die gezogenen Karten des Spielers müssen angezeigt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p>
Eingabe:	<ol style="list-style-type: none"> Auf der Spielseite den Button «Deal» drücken. Den Button «Hit» drücken für eine weitere Karte.
Ausgabe:	<ol style="list-style-type: none"> Der Benutzer sieht die ersten zwei Karten des Spielers graphisch auf der Spielfläche. Der Benutzer sieht eine weitere Karte des Spielers graphisch auf der Spielfläche.

Testfallnummer: AD- 8.4	
Anforderung:	Die gezogenen Karten des Groupiers müssen angezeigt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p>
Eingabe:	<ol style="list-style-type: none"> Auf der Spielseite den Button «Deal» drücken. Der Benutzer drückt auf den Button «Stand».
Ausgabe:	<ol style="list-style-type: none"> Der Benutzer sieht das erste Karte des Dealers graphisch auf der Spielfläche. Der Benutzer sieht die Karten des Dealers graphisch auf der Spielfläche.

Testfallnummer:	AD- 8.5
Anforderung:	Der eingesetzte Betrag muss angezeigt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehenen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert. Der Benutzer drückt den Button «Deal».
Ausgabe:	<ol style="list-style-type: none"> Der eingesetzte Betrag wird unten links bei «Your Bet» angezeigt.

Testfallnummer:	AD- 8.6
Anforderung:	Der aktuelle Kontostand muss angezeigt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> Der aktuelle Kontostand wird oben rechts im Spiel stets angezeigt.

Testfallnummer: AD- 8.7	
Anforderung:	Der Spieler kann wählen, ob eine weitere Karte gezogen werden soll oder nicht.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User drückt auf den Button «Hit». Der User drückt den Button «Stand»
Ausgabe:	<ol style="list-style-type: none"> Am User wird eine weitere Karte wird gezogen, falls es die Regeln zulassen. Am User werden keine weiteren Karten mehr gezogen.

Testfallnummer: AD- 8.8	
Anforderung:	Der Spieler kann ein Betrag auswählen, welches er einsetzen möchte.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert. Der Benutzer drückt den Button «Deal», um es zu bestätigen.
Ausgabe:	<ol style="list-style-type: none"> Der eingesetzte Betrag wird unten links bei «Your Bet» angezeigt.

Testfallnummer: AD- 8.9	
Anforderung:	Der eingesetzte Betrag wird von dem aktuellen Kontostand abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehenen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert. Der Benutzer drückt den Button «Deal», um es zu bestätigen.
Ausgabe:	<ol style="list-style-type: none"> Der aktuelle Kontostand wird oben rechts im Spiel, nach dem Einsetzen von Betrag, aktualisiert.

Testfallnummer: AD- 8.10	
Anforderung:	Der ausgewählte Betrag muss ganzzahlig sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehenen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert.
Ausgabe:	<ol style="list-style-type: none"> Falls der eingesetzte Betrag nicht ganzzahlig ist erscheint eine Fehlermeldung.

Testfallnummer: AD- 8.11	
Anforderung:	Der ausgewählte Betrag muss grösser als Null.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert.
Ausgabe:	<ol style="list-style-type: none"> Falls der eingesetzte Betrag nicht grösser als 0 ist erscheint eine Fehlermeldung.

Testfallnummer: AD- 8.12	
Anforderung:	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt den Betrag auf dem vorgesehen Feld unten links per Hand ein oder klickt auf den gewünschten Button mit dem entsprechendem Wert.
Ausgabe:	<ol style="list-style-type: none"> Falls der eingesetzte Betrag grösser als das Guthaben auf dem Konto ist erscheint eine Fehlermeldung.

Testfallnummer:	AD- 8.13
Anforderung:	Wenn das Spiel verloren wurde wird eine Meldung angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt.</p>
Eingabe:	
Ausgabe:	1. Wenn das Spiel verloren wurde wird ein Text unten recht angezeigt angezeigt, in welcher steht: «Sie haben verloren».

Testfallnummer:	AD- 8.14
Anforderung:	Wenn das Spiel gewonnen wurde wird eine Meldung angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt.</p>
Eingabe:	
Ausgabe:	1. Wenn das Spiel gewonnen wurde wird ein Text unten recht angezeigt, in welcher steht: «Sie haben gewonnen».

Testfallnummer:		AD- 8.15
Anforderung:	Alle gezogenen Karten kommen aus einem gemischten, endlichen Kartenstapel.	
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele. Der Benutzer klickt auf das Spiel Blackjack. Der Benutzer ist auf der Übersichtsseite. Der Benutzer hat den Button «Start Game» gedrückt. Der Benutzer ist auf der Spielseite für Blackjack.	
Eingabe:	1. Auf der Spielseite den Button «Deal» drücken. 2. Den Button «Hit» drücken für eine weitere Karte. 3. Den Button «Stand» drücken um selber keine Karte mehr zu ziehen, somit kann man die Karten des Dealers sehen.	
Ausgabe:	1. Der Benutzer sieht die ersten zwei Karten des Spielers und die erste Karte des Dealers graphisch auf der Spielfläche, welche aus einen gemischten endlichen Kartenstapel erscheinen. 2. Der Benutzer sieht eine weitere Karte des Spielers graphisch auf der Spielfläche, welche aus einen gemischten endlichen Kartenstapel erscheinen. 3. Der Benutzer sieht die Karten des Dealers graphisch auf der Spielfläche, welche aus einen gemischten endlichen Kartenstapel erscheinen.	

Testfallnummer: AD- 8.16	
Anforderung:	Gewinnt der Spieler mit Black Jack, wird der von ihm eingesetzte Betrag im Verhältnis 3:2 ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und mit einem Blackjack gewonnen.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn das Spiel mit Blackjack gewonnen wurde, wird ein Text unten recht angezeigt, in welcher steht: «Sie haben Blackjack». 2. Ihr eingesetzter Betrag wird mit 1.5 multipliziert und im Kontostand oben rechts aktualisiert.

Testfallnummer: AD- 8.17	
Anforderung:	Gewinnt der Spieler nicht mit Black Jack, so wird ihm der eingesetzte Betrag verdoppelt und ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und normal gewonnen.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn das Spiel mit Blackjack gewonnen wurde, wird ein Text unten recht angezeigt, in welcher steht: «Sie haben gewonnen». 2. Ihr eingesetzter Betrag wird mit 2 multipliziert und im Kontostand oben rechts aktualisiert.

Testfallnummer: AD- 8.18	
Anforderung:	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und normal gewonnen oder durch ein Blackjack.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn das Spiel mit Blackjack gewonnen wurde, wird ein Text unten recht angezeigt, in welcher steht: «Sie haben Blackjack» oder «Sie haben gewonnen». 2. Der gewonnene Betrag wird unten rechts angezeigt und der Kontostand oben rechts wird aktualisiert

Testfallnummer: AD- 8.19	
Anforderung:	Beim Unentschieden erscheint eine Meldung.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und es war ein Unentschieden.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn das Spiel mit Blackjack gewonnen wurde, wird ein Text unten recht angezeigt, in welcher steht: «Sie haben Blackjack». 2. Ihr eingesetzter Betrag wird wieder zurückgegeben.

Testfallnummer: AD- 8.20	
Anforderung:	Beim Bestätigen der Meldung von Unentschieden kann das Spiel neu gestartet werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und es war ein Unentschieden.</p> <p>Eine Meldung erscheint.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Die Meldung muss mit einem Klick auf «OK» bestätigt werden.
Ausgabe:	<ol style="list-style-type: none"> 1. Der Kontostand oben rechts wird aktualisiert. 2. Das Spiel wird neu gestartet.

Testfallnummer: AD- 8.21	
Anforderung:	Beim Unentschieden wird der eingesetzte Betrag wieder auf das Konto gezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und es war ein Unentschieden.</p> <p>Eine Meldung erscheint.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Die Meldung muss mit einem Klick auf «OK» bestätigt werden.
Ausgabe:	<ol style="list-style-type: none"> 1. Der eingesetzte Betrag wieder auf das Konto gezahlt. 2. Der Kontostand oben rechts wird aktualisiert.

Testfallnummer: AD- 8.22	
Anforderung:	Beim Verlieren des Spiels wird der eingesetzte Betrag gelöscht.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Die Runde wurde fertig gespielt und der Spieler verlor.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Ihr eingesetzter Betrag wird gelöscht. 2. Der Kontostand oben rechts wird aktualisiert.

Testfallnummer:	AD- 8.23
Anforderung:	Das Spiel kann jederzeit abgebrochen werden, mit der dann das eingesetzte Betrag verloren geht.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der Benutzer drückt auf dem Menu Bar oben links. In dem Menu Bar drückt der Benutzer auf «Quit Game».
Ausgabe:	<ol style="list-style-type: none"> Das Spiel wird abgebrochen. Der eingesetzte Betrag geht verloren.

Testfallnummer:	AD- 8.24
Anforderung:	Falls es die Regeln zulassen, erscheint ein Button für die Versicherung.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Der Dealer hat im Spiel eine Aces gezogen.</p>
Eingabe:	
Ausgabe:	1. Der Spieler drückt auf den Button «Insurance».

Testfallnummer:		AD- 8.25
Anforderung:	Mit einem Klick auf dem Versicherungsbutton, sollte der Spieler ein Betrag setzen können. Dieser Betrag muss positiv, ganzzahlig sein und es darf nicht grösser als das Kontostand sein.	
Voraussetzungen:	Das Programm ist gestartet. Der User ist eingeloggt. Der User hat ist auf der Seite wo man Spiele. Der Benutzer klickt auf das Spiel Blackjack. Der Benutzer ist auf der Übersichtsseite. Der Benutzer hat den Button «Start Game» gedrückt. Der Benutzer ist auf der Spielseite für Blackjack.	
Eingabe:	1. Der Benutzer klickt auf den Button «Insurance». 2. Der Benutzer muss ein Betrag einsetzen, welches positiv, ganzzahlig und nicht grösser als der Kontostand ist. 3. Der Benutzer klickt wieder auf den Button «Insurance».	
Ausgabe:	1. Der Benutzer sieht seinen eingesetzten Betrag.	

Testfallnummer: AD- 8.26	
Anforderung:	Hat der Groupier im nächsten Zug ein Black Jack, wenn der Versicherungsbutton geklickt wurde, wird eine Versicherungsbetrag von 2:1 ausbezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der Benutzer hat auf den Button «Insurance» geklickt. Der Benutzer hat den Betrag für die Versicherung eingesetzt.
Ausgabe:	1. Der eingesetzte Betrag wird verdoppelt ausbezahlt.

Testfallnummer: AD- 8.27	
Anforderung:	Hat der Groupier im nächsten Zug kein Black Jack, wenn der Versicherungsbutton geklickt wurde, wird das eingesetzte Betrag verloren.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Der User hat auf den Button «Insurance» geklickt.</p> <p>Der User hat den Betrag für die Versicherung eingesetzt.</p> <p>Der Groupier hat Blackjack gezogen.</p>
Eingabe:	
Ausgabe:	1. Der eingesetzte Betrag geht verloren.

Testfallnummer: AD- 8.28	
Anforderung:	Falls es die Regeln zulassen erscheint ein Button zur Verdoppelung des Einsatzes.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Der Dealer hatte bei dem Klick auf den Button «Deal» eine 9, 10, J, Q, K und Aces gezogen.</p>
Eingabe:	
Ausgabe:	1. Es erscheint ein Button «Double».

Testfallnummer: AD- 8.29	
Anforderung:	Beim Klick auf den Verdopplungsbutton wird der bereits eingesetzte Betrag nochmals dem Konto abgezogen und dem eingesetzten Betrag hinzugefügt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p> <p>Der Dealer hatte bei dem Klick auf den Button «Deal» eine 9, 10, J, Q, K und Aces gezogen.</p>
Eingabe:	1. Der Spieler klickt auf den Button «Double».
Ausgabe:	1. Der bereits eingesetzte Betrag wird nochmals dem Konto abgezogen und dem eingesetzten Betrag hinzugefügt.

Testfallnummer: AD- 8.30	
Anforderung:	Wenn der Spieler den Betrag durch den Verdopplungsbutton verdoppelt hat, wird nur noch eine Karte gezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Blackjack.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p> <p>Der Benutzer hat den Button «Start Game» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Blackjack.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler klickt auf den Button «Double».
Ausgabe:	<ol style="list-style-type: none"> Der bereits eingesetzte Betrag wird nochmals dem Konto abgezogen und dem eingesetzten Betrag hinzugefügt. Eine weiter Karte wird automatisch gezogen.

2.3.9.8 Testfallspezifikation Roulette

Testfallnummer: AD- 9.1	
Anforderung:	Es ist eine Übersichtseite vorhanden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p>
Eingabe:	1. Der Benutzer klickt auf das Spiel Roulette.
Ausgabe:	1. Der Benutzer kommt auf eine Übersichtsseite.

Testfallnummer: AD- 9.2	
Anforderung:	Die Übersichtsseite enthält den Titel des Spiels.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer ist auf der Übersichtsseite</p>
Eingabe:	
Ausgabe:	1. Die Übersichtsseite hat einen Titel: Roulette.

Testfallnummer: AD- 9.3	
Anforderung:	Die Übersichtsseite enthält den aktuellen Kontostand.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	
Ausgabe:	1. Die Übersichtsseite enthält oben rechts eine Zahl mit dem Kontostand des Benutzers.

Testfallnummer:	AD- 9.4
Anforderung:	Mit Klick auf einen Button wird das Spiel gestartet.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User er hat ist auf der Seite wo man Spiele.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer ist auf der Übersichtsseite.</p>
Eingabe:	1. Der User klickt auf das Spiel Roulette.
Ausgabe:	1. Die Übersichtsseite enthält einen Button. Wenn man diesen klickt kommt man auf das eigentliche Spiel Roulette.

Testfallnummer:	AD- 9.5
Anforderung:	Es kann eine Zahl, Zahlengruppe, Farbe oder Art ausgewählt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der Benutzer hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der User klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der User ist auf der Spielseite für Roulette.</p> <p>Der Benutzer hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	1. Der Benutzer setzt den Chip auf der gewünschten Position auf dem Roulette Tisch.
Ausgabe:	1. Der Chip bleibt an diesem Ort.

Testfallnummer:	AD- 9.6
Anforderung:	Es kann ein Betrag ausgewählt werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der Benutzer ist eingeloggt.</p> <p>Der Benutzer hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der User gibt den Betrag auf dem vorgesehenen Feld unten links per Hand ein oder benutzt den Schieberegler. 2. Der Benutzer drückt den Button «Confirm».
Ausgabe:	

Testfallnummer:	AD- 9.7
Anforderung:	Der Betrag wird meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der Benutzer ist eingeloggt.</p> <p>Der Benutzer hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der User klickt auf das Spiel Roulette.</p> <p>Der User hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der Benutzer setzt seinen Chip auf das gewünschte Feld.
Ausgabe:	<ol style="list-style-type: none"> 1. Die Anzeige für den Kontostand wird mit dem neuen Kontostand aktualisiert.

Testfallnummer: AD- 9.8	
Anforderung:	Der Betrag muss ganzzahlig sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p>
Eingabe:	<ol style="list-style-type: none"> Der User gibt eine Kommazahl in das Eingabefeld ein. Der Benutzer drückt den «Confirm» Button.
Ausgabe:	<ol style="list-style-type: none"> Das Programm gibt eine Meldung aus, welche besagt, es dürfen nur Ganzzahlen eingegeben werden. Der Betrag im Eingabefeld wird wieder zurückgesetzt.

Testfallnummer: AD- 9.9	
Anforderung:	Der Betrag muss grösser als Null sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p>
Eingabe:	<ol style="list-style-type: none"> Der Benutzer gibt einen Betrag ein der Null ist oder kleiner als Null Der Benutzer drückt den «Confirm» Button gedrückt.
Ausgabe:	<ol style="list-style-type: none"> Das Programm gibt eine Meldung aus, dass der Betrag grösser als Null sein muss.

Testfallnummer: AD- 9.10	
Anforderung:	Der Betrag darf nicht grösser als das Guthaben auf dem Konto sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p>
Eingabe:	<ol style="list-style-type: none"> Der Benutzer gibt einen Betrag ein, der grösser ist als sein Kontostand. Der Benutzer drückt den «Confirm» Button gedrückt.
Ausgabe:	<ol style="list-style-type: none"> Das Programm gibt eine Meldung aus, dass der Betrag grösser als Null sein muss.

Testfallnummer: AD- 9.11	
Anforderung:	Einsatz auf eine einzelne Zahl (Straight Up Bets): Die Auszahlung liegt hier bei 35:1.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzten gewählt.</p> <p>Der Chip zum setzten ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler setzt den Chip auf eine Zahl auf dem Roulette Tisch.
Ausgabe:	<ol style="list-style-type: none"> Wenn die Roulette Kugel auf der Zahl stehen bleibt, auf die man gesetzt hat. Wird dem Benutzer ein Gewinn von 35 zu 1 ausbezahlt.

Testfallnummer: AD- 9.12	
Anforderung:	Einsatz auf zwei benachbarte Zahlen (Split Bets): Die Gewinnauszahlung beträgt 17:1.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler setzt den Chip in die Mitte von zwei Zahlen, auf die Linie. Das geht horizontal wie vertikal.
Ausgabe:	<ol style="list-style-type: none"> Wenn die Roulette Kugel auf einer der beiden Zahlen stehen bleibt, auf die man gesetzt hat. Wird dem Benutzer ein Gewinn von 17 zu 1 ausbezahlt.

Testfallnummer: AD- 9.13	
Anforderung:	Einsatz auf drei Zahlen (Street Bets): Die Gewinnauszahlung beträgt 11 zu 1.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler setzt den Chip auf die Linie unter drei Zahlen.
Ausgabe:	<ol style="list-style-type: none"> Wenn die Roulette Kugel auf einer der drei Zahlen stehen bleibt, auf die man gesetzt hat. Wird dem Benutzer ein Gewinn von 11 zu 1 ausbezahlt.

Testfallnummer:	AD- 9.14
Anforderung:	Einsatz auf vier Zahlen (Corner Bets): Die Gewinnauszahlung beträgt 8:1.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der Benutzer hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler setzt den Chip in die Mitte von vier Zahlen auf das Kreuz.
Ausgabe:	<ol style="list-style-type: none"> Wenn die Roulette Kugel auf einer der vier Zahlen stehen bleibt, auf die man gesetzt hat. Wird dem Benutzer ein Gewinn von 8 zu 1 ausbezahlt.

Testfallnummer:	AD- 9.15
Anforderung:	Einsatz auf fünf Zahlen (Five Number Bets): Wird das mit 6:1 ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> Der Spieler setzt den Chip in die Mitte von zwei Zahlen auf das untere Kreuz in der untersten Reihe.
Ausgabe:	<ol style="list-style-type: none"> Wenn die Roulette Kugel auf einer der beiden Zahl stehen bleibt, auf die man gesetzt hat. Wird dem Benutzer ein Gewinn von 6 zu 1 ausbezahlt.

Testfallnummer: AD- 9.16	
Anforderung:	Rot oder Schwarz: Wird das mit 1:1 ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der Spieler setzt den Chip auf das Feld «Rot». 2. Der Spieler setzt den Chip auf das Feld «Schwarz».
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn die Roulette Kugel auf einer der gesetzten Farbe stehen bleibt. Wird dem Benutzer ein Gewinn von 1 zu 1 ausbezahlt.

Testfallnummer: AD- 9.17	
Anforderung:	Ungerade oder Gerade: Wird das mit 1:1 ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der Spieler setzt den Chip auf das Feld «Ungerade». 2. Der Spieler setzt den Chip auf das Feld «Gerade».
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn die Roulette Kugel auf einer geraden Zahl stehen bleibt und der Spieler hat auf «Gerade» gesetzt hat. Wird dem Benutzer ein Gewinn von 1 zu 1 ausbezahlt. Wenn aber die Roulette Kugel auf einer ungeraden Zahl stehen bleibt und der Spieler hat auf «Ungerade» gesetzt hat. Wird dem Benutzer ein Gewinn von 1 zu 1 ausbezahlt.

Testfallnummer: AD- 9.18	
Anforderung:	Niedrig oder Hoch: Im Gewinnfall wird Ihnen der gleiche Betrag ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzen gewählt.</p> <p>Der Chip zum setzen ist aufgetaucht.</p>
Eingabe:	<ol style="list-style-type: none"> 1. Der Spieler setzt den Chip auf das Feld «Niedrig». 2. Der Spieler setzt den Chip auf das Feld «Hoch».
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn die Roulette Kugel auf einer der ersten 18 Zahlen stehen bleibt und der Spieler hat auf «Niedrig» gesetzt hat. Wird dem Benutzer ein Gewinn von 1 zu 1 ausbezahlt. Wenn aber die Roulette Kugel auf einer Zahl zwischen 19 und 36 stehen bleibt und der Spieler hat auf «Hoch» gesetzt hat. Wird dem Benutzer ein Gewinn von 1 zu 1 ausbezahlt.

Testfallnummer:	AD- 9.19
Anforderung:	Spalten: Wird das mit 2:1 ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzten gewählt.</p> <p>Der Chip zum setzten ist aufgetaucht.</p>
Eingabe:	1. Der Spieler setzt den Chip auf eines der Felder «2:1»
Ausgabe:	1. Wenn die Roulette Kugel auf eine Zahl, welche auf der gesetzten Spalte vorkommt landet. Wir ihm der Einsatz im Verhältnis 2:1.

Testfallnummer:	AD- 9.20
Anforderung:	Dutzend: Im Gewinnfall wird der doppelte Einsatz ausgezahlt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat einen gewünschten Betrag zum setzten gewählt.</p> <p>Der Chip zum setzten ist aufgetaucht.</p>
Eingabe:	1. Der Spieler setzt den Chip auf eines der Felder «First 12», «Second 12», «Third 12»
Ausgabe:	1. Wenn die Roulette Kugel auf einer Zahl, welche auf einem der drei Felder ist, und man auf das Feld gesetzt hat. Wir ihm der Einsatz im Verhältnis 2:1.

Testfallnummer: AD- 9.21	
Anforderung:	Der gewonnene Betrag sowie der neue Kontostand werden angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User hat ist auf der Seite wo man Spiele auswählen kann.</p> <p>Der Benutzer klickt auf das Spiel Roulette.</p> <p>Der Benutzer hat auf der Übersichtsseite den Button: «Spielen» gedrückt.</p> <p>Der Benutzer ist auf der Spielseite für Roulette.</p> <p>Der User hat seinen Zug durchgeführt.</p> <p>Der Spieler hat das Rad gedreht.</p>
Eingabe:	
Ausgabe:	<ol style="list-style-type: none"> 1. Wenn der Spieler eine gewinnende Wette gemacht hat. Wird ihm der Gewinn ausgezahlt. 2. Sein neuer Kontostand wird im angezeigt.

2.3.9.9 Testfallspezifikation Bingo

Testfallnummer: AD- 10.1	
Anforderung:	Es ist eine Übersichtseite vorhanden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p>
Eingabe:	Der Benutzer klickt auf das Spiel Bingo.
Ausgabe:	Übersichtsseite wird angezeigt.

Testfallnummer: AD- 10.2	
Anforderung:	Die Übersichtsseite enthält den Titel des Spiels.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p>
Eingabe:	Der Benutzer klickt auf das Spiel Bingo.
Ausgabe:	Titel wird angezeigt.

Testfallnummer:	AD- 10.3
Anforderung:	Die Übersichtsseite enthält den aktuellen Kontostand.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p>
Eingabe:	Der Benutzer klickt auf das Spiel Bingo und den Button «Start Game»
Ausgabe:	Rechts oben wird der aktuelle Kontostand angezeigt.

Testfallnummer:	AD- 10.4
Anforderung:	Mit Klick auf einen Button wird das Spiel gestartet.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p>
Eingabe:	Der Benutzer klickt auf den Button «Buy Card».
Ausgabe:	Ein neues Fenster mit dem Spiel wird angezeigt.

Testfallnummer:	AD- 10.5
Anforderung:	Der Betrag für die Karte darf nicht grösser als das Guthaben auf dem Konto sein.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p>
Eingabe:	Der Benutzer klickt auf den Button «Buy Card».
Ausgabe:	Wenn der Betrag vom Benutzer kleiner wie der Betrag der Karte ist, erscheint eine Fehlermeldung, ansonsten startet das Spiel ganz normal.

Testfallnummer:	AD- 10.6
Anforderung:	Der Betrag wird meinem Konto abgezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p>
Eingabe:	Der Benutzer klickt auf den Button «Buy Card».
Ausgabe:	Wenn der Betrag vom Benutzer grösser wie der Betrag der Karte ist, wird vom Benutzer 100 Chips abgezogen und im nächsten Fenster sieht man den Stand der Spielchips.

Testfallnummer: AD- 10.7	
Anforderung:	Die erworbene Karte wird angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p>
Eingabe:	Der Benutzer klickt auf den Button «Buy Card».
Ausgabe:	Wenn der Betrag ausreicht wird man zum nächsten Fenster weitergeleitet und die erworbene Karte wird angezeigt.

Testfallnummer: AD- 10.8	
Anforderung:	Die markierten Felder sind jederzeit ersichtlich.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer klickt auf eine Zahl, welche sich auf der Karte befindet.
Ausgabe:	Auf der Zahl wird nun ein rotes Kreuz erscheinen.

Testfallnummer: AD- 10.9	
Anforderung:	Die Nummer auf der aktuell gezogenen Bingo-Kugel ist ersichtlich.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine Eingabe tätigen, da die Zahlen alle 30 Sekunden automatisch generiert werden.
Ausgabe:	Über dem Sack wird die Zahl angezeigt.

Testfallnummer:	AD- 10.10
Anforderung:	Die Anzahl der virtuellen Gegenspieler ist ersichtlich.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Die Gegenspieler werden rechts unten angezeigt.

Testfallnummer:	AD- 10.11
Anforderung:	Der aktuelle Kontostand wird angezeigt.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Rechts oben wird der aktuelle Spielstand angezeigt.

Testfallnummer:	AD- 10.12
Anforderung:	Es wird alle 30 Sekunden eine neue Bingo-Kugel aus der Trommel gezogen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Wenn der Timer im Programm von 30 Sekunden abgelaufen ist, wird oberhalb vom Sack die Zahl neu aktualisiert.

Testfallnummer:	AD- 10.13
Anforderung:	Die Nummer auf der aktuell gezogenen Bingo-Kugel wird gezeigt und bleibt für den Rest der Runde ersichtlich.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Wenn eine neue Kugel aus der Trommel gezogen wird, wird im Fenster unter «Pulled Numbers» angezeigt.

Testfallnummer:	AD- 10.14
Anforderung:	Die Felder auf den Karten können ohne Prüfung oder Restriktionen Markiert werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer klickt auf eine der Zahlen auf der Karte.
Ausgabe:	Auf der Zahl erscheint nun ein rotes Kreuz.

Testfallnummer:	AD- 10.15
Anforderung:	Das Spiel bietet die Möglichkeit, den geglaubten gewinn anzumelden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss auf den Knopf «Bingo» drücken.
Ausgabe:	Es erscheint eine Meldung ob man gewonnen hat oder das Spiel noch weiter geht, da der geglaubte Gewinn falsch ist.

Testfallnummer:	AD- 10.16
Anforderung:	Meine Karte ist auf die Legitimität des Gewinnes validiert.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss auf den Knopf «Bingo» drücken.
Ausgabe:	Wenn die angekreuzten Zahlen stimmen, wird dem Betrag der Betrag von den Bots hinzugeschrieben.

Testfallnummer:	AD- 10.17
Anforderung:	Es wurde von keinem meiner Gegenspieler vor mir «Bingo» angemeldet.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer kann auf den Knopf «Bingo» drücken.
Ausgabe:	Wenn die angekreuzten Zahlen stimmen, wird dem Betrag der Betrag von den Bots hinzugeschrieben.

Testfallnummer:	AD- 10.18
Anforderung:	Einer meiner 5-10 Gegenspieler mit je einer Karte, brechen das Spiel mit «Bingo» mit einer Verzögerung von 15 Sekunden vor mir ab, wenn sie einen Gewinn auf der Karte verzeichnen.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Es erscheint ein Fenster, wo steht, dass man verloren hat.

Testfallnummer:	AD- 10.19
Anforderung:	Mein mit «Bingo» angemeldeter Gewinn bestätigt sich als falsch.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss auf den Knopf «Bingo» drücken.
Ausgabe:	Ein Fenster erscheint, wo steht das Der «Bingo-Call» falsch ist und das Spiel geht ganz normal weiter.

Testfallnummer:	AD- 10.20
Anforderung:	Der Gewinn wird meinem Konto gutgeschrieben.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss auf den Knopf «Bingo» drücken.
Ausgabe:	Wenn der «Bingo-Call» korrekt ist wird dem Betrag des Benutzer 250 hinzugeschrieben und rechts oben angezeigt.

Testfallnummer:	AD- 10.21
Anforderung:	Das Spiel kann jederzeit abgebrochen werden.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer kann links oben auf die 3 Linien links oben drücken und anschliessend auf «Quit Game».
Ausgabe:	Man kommt ins Fenster der Spielübersicht.

Testfallnummer: AD- 10.22	
Anforderung:	Bereits gesetzte Beträge gehen beim Abbruch verloren.
Voraussetzungen:	<p>Das Programm ist gestartet.</p> <p>Der User ist eingeloggt.</p> <p>Der User ist auf der Seite wo man alle Spiele sieht.</p> <p>Der Benutzer klickt auf das Spiel Bingo.</p> <p>Der Benutzer ist auf der Übersichtsseite von Bingo.</p> <p>Der Benutzer muss auf den Button «Start Game» klicken.</p> <p>Der Benutzer muss auf den Button «Buy Card» klicken.</p> <p>Der Benutzer befindet sich nun im Hauptspiel.</p>
Eingabe:	Der Benutzer muss keine weiteren Eingaben tätigen.
Ausgabe:	Der Spielbetrag des Spielers bleibt gleich wie zuvor.

2.4 Entscheiden

2.4.1 Vorgehen für GUI Entscheidung

Die Entscheidungswege und Kompetenzen sind klar, es werden mehrere Designs entworfen und falls einem das eine Design passt, kann man damit fortfahren. Ansonsten muss man es nochmals überarbeiten. Abgemacht ist, dass alles möglichst ähnlich gemacht wird, damit es einfacher ist den Vorschlag auf das Gemeinsame zu übertragen. Deshalb versuchen wir die Designs in all unseren Spielen gleichzuhalten. Unsere Besprechungen waren, dass unsere Spiele einen blauen Hintergrund haben muss. Dazu muss auch jedes Spiel links oben einen Menübar haben. Rechts oben bestimmten wir zwei Icons für den User Name und Kontostand, die vorhanden sein müssen. Den Rest konnten wir selber darstellen.

Die Ressourcen, welche uns bereitgestellt werden, versuchen wir intensiv zu nutzen. Natürlich werden wir auch immer auf das Internet zurückgreifen oder uns an die Klassenmitglieder und Lehrpersonen wenden.

2.4.2 Entscheidung des Designs

2.4.2.1 Entscheidung Login

Das zweite Login GUI fanden wir am besten, da es in zwei Teilen aufgeteilt ist. Das macht die Schalteroberfläche direkt übersichtlicher. Links oben kann sicher der Benutzer, der noch keine Anmelddaten zu Verfügung hat, mit dem Button «Sign In» registrieren. Auf der rechten Hälfte kann sich der Benutzer anmelden. Dadurch ist die Login Maske verständlicher für den Benutzer. Die Funktionen sind in Blöcken aufgeteilt, welches auch dem Benutzer alles vereinfacht.

Das erste GUI ist dagegen weniger übersichtlicher, da es in keine Blöcke aufgeteilt ist. Es sind nur Texte und Buttons vorhanden. Da sich auch alles in der Mitte befindet ist dieses Design nicht das Beste.

2.4.2.2 Entscheidung Home

Bei diesen zwei GUI's haben wir uns für die zweite Version entschieden, weil es viel übersichtlicher ist. Man hat links die Einleitung und kann sich erkundigen. Auf der rechten Seite kann man Chips kaufen. Der Shop sieht viel besser aus als bei der ersten Variante. Auch wenn man auf den Button «Buy it» drückt dann kauft man Chips und geht direkt weiter zum nächsten Screen. Beim anderen muss man noch den Button «Next» drücken.

2.4.2.3 Entscheidung Shop

Wir haben uns für das untere Design entschieden. Erstens haben wir dort rechts oben Icons mit dem Nutzernamen und dem Kontostand. Dazu gibt es einen Slider der das auswählen von Beträgen verschönert und auch vereinfacht. Außerdem kann mit seiner Zahlungsmethode Geld direkt in Spielgeldumwandeln. Man hat man eine Übersicht wieviel 1 Chip wert hat. Es bietet allgemein mehr Übersicht und Benutzerfreundlichkeit.

2.4.2.4 Entscheidung Spielauswahl

Die Gruppe hat sich für das untere Design entschieden, weil es übersichtlicher ist. Bei diesem Design sieht man direkt, was man alles machen kann. Man sieht alle Spiele auf einen Blick, das ist viel benutzerfreundlicher. Beim oberen Design ist es viel umständlicher für den Benutzer. Der Benutzer muss hier viel mehr Klicks machen bis er spielen kann. Ebenfalls ist der Bildschirm nicht voll ausgefüllt.

2.4.2.5 Entscheidung Baccara

Das erste GUI kam für mich überhaupt nicht in Frage, weil alle Karten in der Mitte dargestellt werden. Das bedeutet die Übersichtlichkeit ist nicht gut. Zudem kam noch dazu, dass der Username und Chips Kontostand nicht ganz oben rechts angezeigt werden. Es wird unter dem Absatz angezeigt, was auch nicht gut aussieht. Das 2. GUI gefällt mir besser und es ist ziemlich übersichtlicher. Man hat auch mehrere Möglichkeiten, weil es mehrere Buttons gibt.

2.4.2.6 Entscheidung Yatzy

Ich habe mich für das untere entschieden. Es ist einfacher oberhalb von «Roll» die Würfel darzustellen da dort mehr Platz ist. Beim oben ist es unübersichtlich da die Figuren in der Mitte wären und die Würfel müsste man links und rechts von denen machen was für Verwirrung sorgt. Nur der Vorteil beim oberen ist, dass man einen «Bet» direkt auf der Spielseite tätigen kann.

2.4.2.7 Entscheidung Blackjack

Ich habe mich für das untere Design entschieden, da es klar, einfach und übersichtlich ist. Ich setze einen grossen Wert auf Übersichtlichkeit. Damit alles verständlich für den Benutzer ist. Der Aufbau des Designs ähnelt auch einem richtigen Casino durch die Chips unten links. Denn wenn man auf einer der Chips klickt, setzt man den Wert des Chips als Einsatz ein. Das sieht sehr schön aus und macht es dem Benutzer viel einfacher den Betrag einzusetzen. Die Darstellung der Karten sieht auch sehr viel realer und schöner aus als die bei dem oberen Design.

Das von mir präferierte Design ist sehr ähnlich zu dem anderen, trotzdem unterscheidet es sich in kleinen erweiterten Funktionen.

2.4.2.8 Entscheidung Roulette

Ich habe mich für das untere Design entschieden, da es viel einfacher und intuitiver ist für den Benutzer. Er fühlt sich an wie in einem echten Casino. Denn da ist es nicht eine vereinfachte Version von Roulette, sondern man sieht hier alle möglichen Wetten.

Beim ersten Design kann man zum Beispiel eine Street Bet über einen Knopf links auf dem Brett auswählen. Dann wird eine Aufforderung angezeigt wo man auswählen kann auf welchen Zahlen man setzen möchte. Beim zweiten Design habe ich es mir nicht einfacher gemacht, dafür dem Benutzer. Hier kann der Spieler einen freiwillbaren Betrag in vorgesehenen Fenster eingeben. Dann bekommt er einen, vorher nicht angezeigten, Chip. Welchen er auf dem Spielfeld, per Drag and Drop, auf eine beliebige Zahl ziehen kann.

Ich habe mich schlussendlich für die Benutzerfreundlichere Variante entschieden. Deshalb setzte ich das zweite Design ins Programm um.

2.4.2.9 Entscheidung Bingo

Ich habe mich für das untere Design entschieden, da es klar, einfach zu verstehen und modern ist. Die gezogenen Nummern werden schön in einem Kästchen aufgelistet, somit ist es für den Benutzer einfach einen guten Überblick zu haben, welche Karten bereits gezogen wurden und welche nicht. Im Vergleich zum vorherigen Design gibt es nun einen Button, bei dem man die Karten der Gegenspieler anzeigen kann, dies ist für den Spieler ein großartiges Feature, da er den Fortschritt vom Bot sehen kann. Die Gegenspieler haben in diesem Design auch zufällig einen Namen bekommen, dies ist nicht weiter ein wichtiger Bestandteil, ist aber gut für die Benutzerfreundlichkeit, weshalb ich mich schlussendlich auch für dieses Design entschieden habe.

2.4.3 Nutzwertanalyse

2.4.3.1 Entscheidung Branches

Ein Branch ist eine andere Version eines Programmes.

Kriterien	Gewichtung	Variante 1 (1 Branch)		Variante 2 (Mehrere Branches)	
		Faktor	Total V1	Faktor	Total V2
Einrichtung	10%	500%	0.5	300%	0.3
Übersichtlich	15%	500%	0.75	400%	0.6
Benutzerfreundlich	25%	500%	1.25	400%	1
Aufwand	30%	300%	0.9	500%	1.5
Nützlichkeit	20%	400%	0.8	300%	0.6
	100%		4.2		4

Abbildung 43: Entscheidungsmatrix Branches

Ein wichtiger Entscheidungsfaktor für uns war der Aufwand. Da man bei mehreren Branches schlussendlich jedes Programm auf den Master-Branch bringen muss, ist dies deutlich aufwändiger. Der einzige Nachteil bei einem Branch ist, dass man nicht gleichzeitig an einer Datei arbeiten kann, da es sonst zu Fehlern kommt.

2.4.3.2 Entscheidung Pattern

Fassade und MVC sind Code Patterns, d.h. es ist schon vorgegebener Code, welchen man verwenden kann. Jeder Programmierer sollte die Patterns kennen und mit diesen arbeiten können. Mit Fassade kann man eine vereinheitlichte und/oder vereinfachte Schnittstelle zu einem komplexen System, im Code einbauen.

Bei MVC wird das ganze Programm in drei Komponenten aufgeteilt. Damit ist das Programm flexibler. Die drei Komponenten sind Model, View und Controller.

Kriterien	Gewichtung	Variante 1 (MVC)		Variante 2 (Fassade)	
		Faktor	Total V1	Faktor	Total V2
Hilfestellungen	10%	5	0.5	4	0.4
Übersichtlich	15%	5	0.75	4	0.6
Benutzerfreundlich	25%	3	0.75	2	0.5
Aufwand	30%	4	1.2	2	0.6
Nützlichkeit	20%	4	0.8	3	0.6
	100%		4		2.7

Abbildung 44: Entscheidungsmatrix Pattern

Auch hier war für uns der wichtigste Punkt der Aufwand. Da man beim MVC Pattern eine schöne strukturierte Ordnerstruktur hat, dachten wir es gibt weniger Aufwand, wie wenn alles über einen Punkt ausläuft, wie bei der Fassade Pattern. Auch die Übersicht ist daher höher als bei der Fassade.

2.5 Realisieren

2.5.1 Kurzfassung

Während dem Umsetzen unserer Casinos arbeiten wir nebenbei auch an unserer Dokumentation. Wir kommen ehrlich gesagt, nicht immer mit der Dokumentation mit, weil wir zu tief in das Programmieren versunken sind. Trotzdem haben wir versucht unser Bestes zu geben. In diesem Abschnitt kann man sehen, ob wir uns beim Programmieren des gesamten Projektes an die Pläne gehalten haben. Wir haben das Programm und den Code mit Screenshot dokumentiert.

2.5.1 Implementation

2.5.2.1 Implementation Login

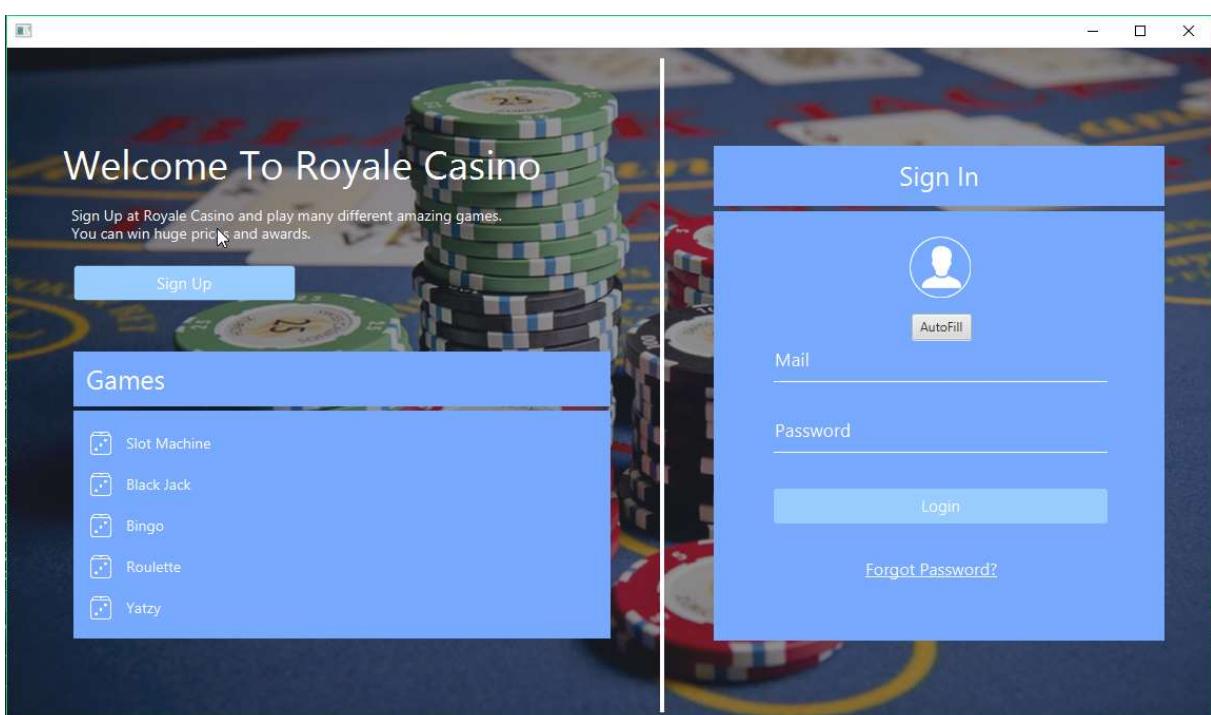


Abbildung 45: Login Casino Royale

Das Login ist übersichtlich, klar und modern. Für den Benutzer ist es nicht kompliziert, Entscheidungen zu treffen, was er machen muss. Auf der linken Seite kann man sich mit dem Button «Sign Up» registrieren, wenn man noch kein Konto hat. Weiter unten sieht man die Auflistung der verschiedenen Spiele. Auf der rechten Seite kann man sich einloggen, wenn man bereits einen Benutzer bei Royale Casino ist. Falls man sein Passwort vergessen hat, kann man auf «Forgot Password?» drücken und man wird zu einem neuen Fenster weitergeleitet, welches alle Schritte erklärt.

Stolpersteine:

Zuerst Probleme mit dem verifizieren, das senden der E-Mail mit dem Verifizierungscode war etwas kompliziert.

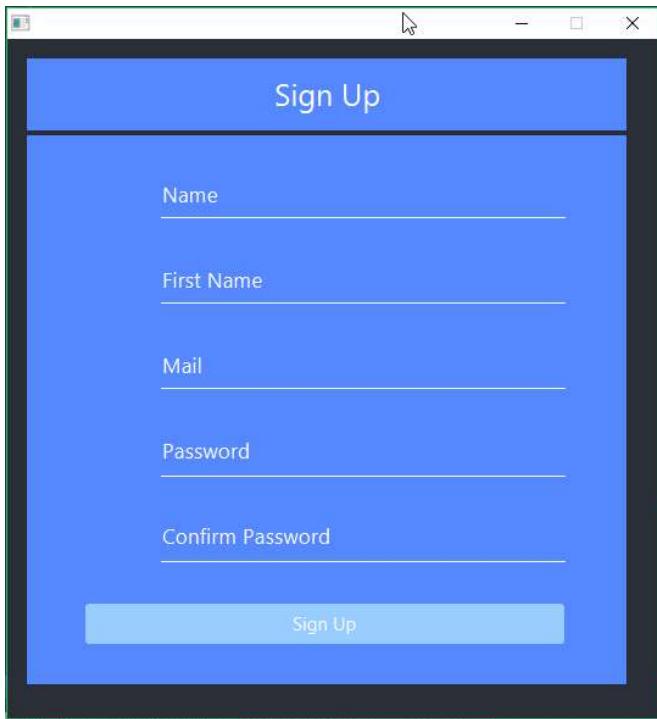


Abbildung 46: Sign Up Casino Royale

Dies ist das Fenster, falls man noch kein Benutzer bei Royale Casino ist. Man muss seinen Namen, Vornamen, seine Mail und sein Passwort eingeben, Danach, wenn alles richtig eingegeben wurde, muss man noch seine Email verifizieren und dann ist man bereits registriert.

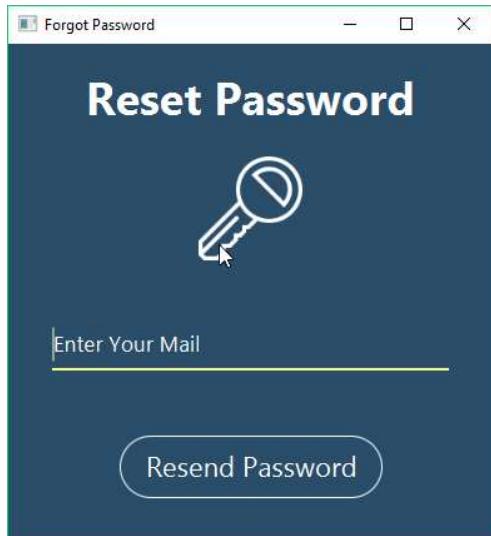


Abbildung 47: Forgot Password

Dies ist Fenster für den Fall, dass man sein Passwort vergessen hat. Man muss blos seine Mail-Adresse eingeben und dann auf den Button «Resend Password» drücken. Danach bekommt man eine Mail mit dem Passwort.

Stolpersteine:

Die verschiedenen Überprüfungen ob die richtigen Zeichen eingegeben wurden bei der Mail Adresse gab zuerst Probleme. Dies konnte ich jedoch beheben.

2.5.2.2 Implementation Datenbank

	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>	<input type="button" value="Edit"/> <input type="button" value="Copy"/> <input type="button" value="Delete"/>
		1	Meyner	Yannick	casion@royale.ch				
		2	Eugster	Joel	joel.eugster@stud.bbbaden.ch				
		5	admin	admin	admin@admin.ch				
		9	Jelle	Schutter	jelle.schutter@gmx.de				
		21	test	user	table@view.ch				
		22	Blatter	Hanspeter	y@s.c				
		23	test	yannick	yannick10@sunrise.ch				
		24	Steinacher Julian		uselesstrash122@gmail.com				

Abbildung 48: Datenbank

Dies ist die Datenbank für alle Benutzer, welche man bei PhpMyAdmin aufrufen kann. Man kann den Namen, Vornamen, Email, Passwort, Verifizierungsstatus, Versuche für die Verifizierung und die Anzahl Chips sehen. Ausserdem kann man die Einträge ändern, Konten sperren oder gar löschen.

```
Statement myStmt = sqlLogin().createStatement();
String query = " insert into " + sqlGetTable() + " (name, firstname, email, password, verification, tries, chips)"
    + " values (?, ?, ?, ?, ?, ?, ?)";
PreparedStatement preparedStmt = sqlLogin().prepareStatement(query);
preparedStmt.setString(1, nameSignUp);
preparedStmt.setString(2, firstNameSignUp);
preparedStmt.setString(3, mailSignUp);
preparedStmt.setString(4, passwordSignUp);
preparedStmt.setString(5, random);
preparedStmt.setString(6, "5");
preparedStmt.setInt(7, 0);
preparedStmt.execute();
```

Abbildung 49: Code Datenbank

Dies ist der Code in Java welcher für die Einträge in die Datenbank verantwortlich ist. Mit dem Befehl «preparedStmt.execute();» werden die Einträge an die Datenbank übergeben.

```
public Integer getGekauftechips() throws SQLException {
    Statement statementChips;
    ResultSet resultSetChips;
    statementChips = sqlLogin().createStatement();
    String verifyChips = "select gekauftechips from logbaccara where user like '%" + sqlMail;
    resultSetChips = statementChips.executeQuery(verifyChips);
    while ((resultSetChips.next())) {
        gekauftechips = Integer.valueOf(resultSetChips.getString("gekauftechips"));
    }
    return gekauftechips;
}
```

Abbildung 50: Code Datenbank

Dies ist der Code in Java welcher für Auslesen von Einträgen in der Datenbank verantwortlich ist.

2.5.2.3 Implementation Shop

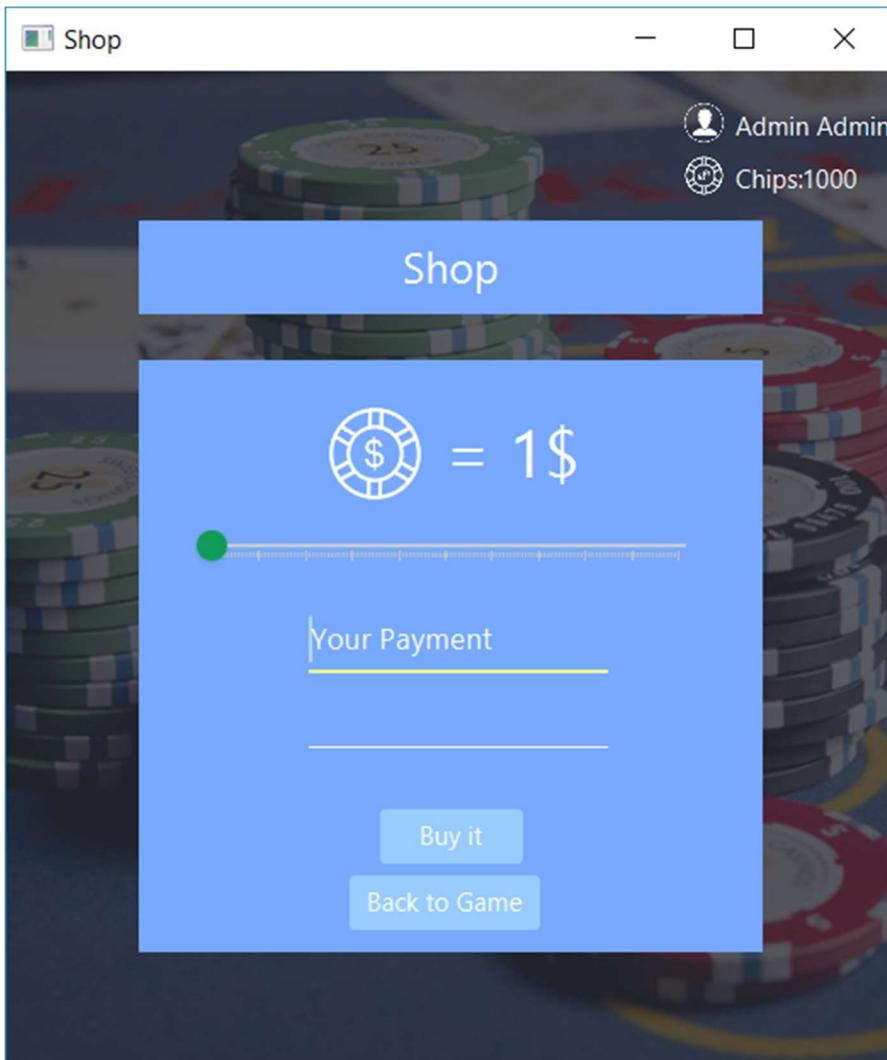


Abbildung 51: Shop

Wenn der Benutzer sich Chips kaufen möchte, dann wird er zu dieser Seite gelangen. Man kann mit einem Slider ein Betrag wählen oder man kann den Betrag selber von Hand in das Text-Field schreiben. Mit einem Klick auf „Buy it“ wird der Betrag dem Account hinzugefügt. Mit dem Button „Back to Game“ gelangt man wieder zurück auf die Seite. Der Benutzername und die Chips werden im Label angezeigt. Wir hatten noch ein Problem, nämlich wenn man im Shop Chips kauft und dann wieder zurück zum Fenster geht wird das Label nicht aktualisiert. Das bedeutet es wird immer noch der Kontostand angezeigt, den man vor dem Kauf hatte.

```
@Override  
public void initialize(URL url, ResourceBundle rb) {  
    try {  
        gekaufteChips = actions.getGekauftChips();  
    } catch (SQLException ex) {  
        Logger.getLogger(ShopController.class.getName()).log(Level.SEVERE, null, ex);  
    }  
    instance = this;  
    try {  
        // TODO  
        lblChipss.setText(actions.getChips());  
    } catch (SQLException ex) {  
        Logger.getLogger(HomescreenController.class.getName()).log(Level.SEVERE, null, ex);  
    }  
    lblActiveUser.setText(actions.getUsername());  
}  
  
@FXML  
private void GetAmountSlider(MouseEvent event) {  
    int toInt = (int) sliderChips.getValue();  
    txtPayment.setText(Integer.toString(toInt));  
    lblChips.setText(Integer.toString(toInt));  
}
```

Abbildung 52: Code Shop

In der initialize Methode wird zuerst das Label so angepasst, dass die aktuellen Chips im Label angezeigt werden. In der zweiten Methode wird vom Slider der Betrag genommen und in das Text-Field „Your Payment“ geschrieben und ins Label.

2.5.2.4 Implementation Baccara

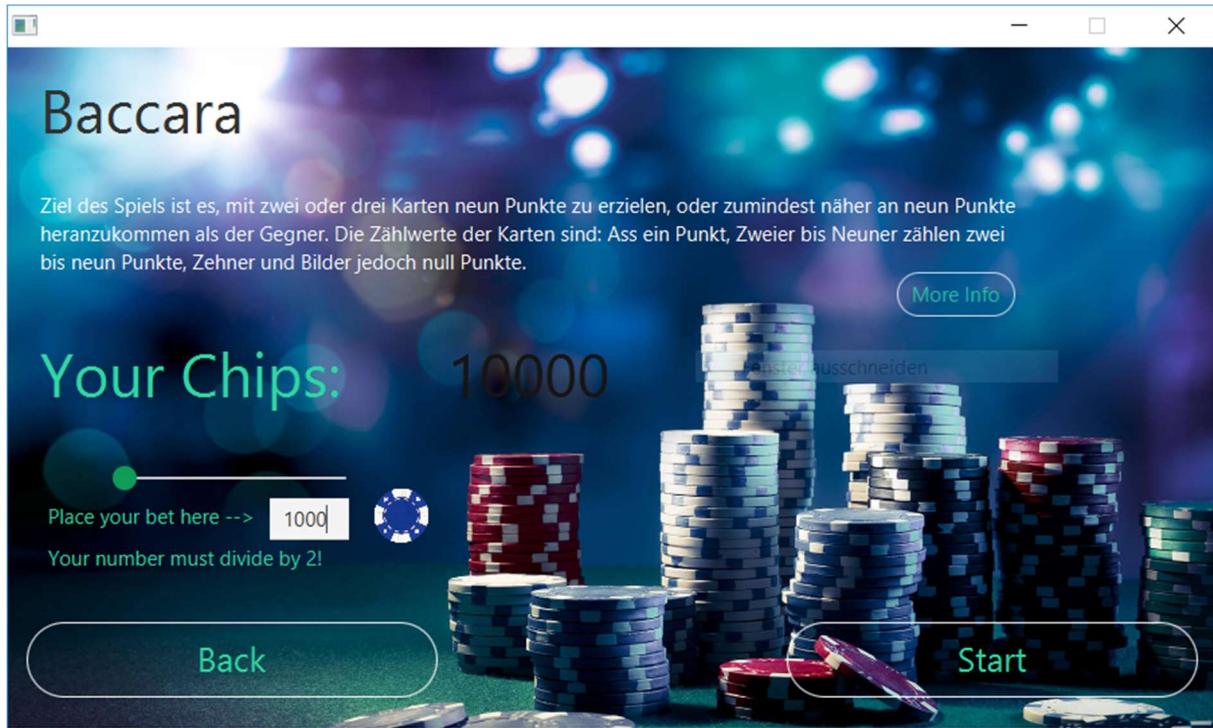


Abbildung 53: Start Screen Baccara

Der Name des Spiels ist zu sehen und auf der Seite wird das Spiel, mit einer kleinen Beschreibung, beschrieben. Man kann entweder Mehr Informationen zum Spiel erhalten, man kann zurück zu den Spielen und man kann das Spiel, wenn man seinen Einsatz gesetzt hat starten. Der Einsatz muss mindestens 10 Chips sein und nur ganzzahlige Chips dürfen eingegeben werden. Das habe ich so gemacht, damit der Betrag schon bevor das Spiel startet abgezogen wird, damit wenn der Spieler das Spiel verlässt, der Betrag schon abgezogen wurde.

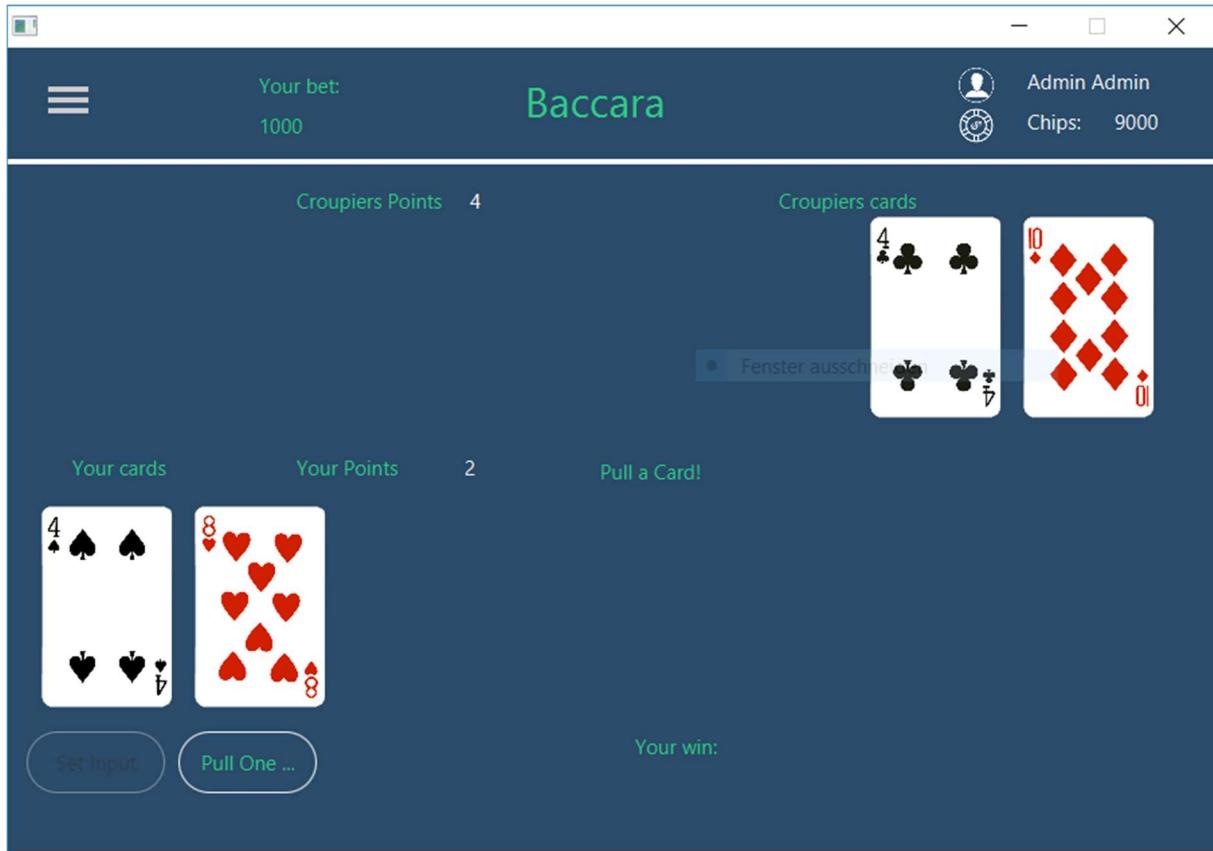


Abbildung 54: in game Baccara

Wenn man auf der Titelseite den Betrag eingegeben hat und auf den Button «Start» gedrückt hat, dann wird das Spiel gestartet. Wenn man dann den Button «Set Input» drückt, dann werden die ersten zwei Karten des Spielers und des Croupiers angezeigt. Man kann entweder noch eine Karte ziehen, wenn man weniger als sechs Punkte hat oder man kann so bleiben. Es wird in einem Label direkt die Punkte angezeigt. Das habe ich so gemacht, weil es sonst für ein paar Nutzer unverständlich wäre. Auch Benutzername, Chips und gesetzter Betrag werden auch angezeigt.

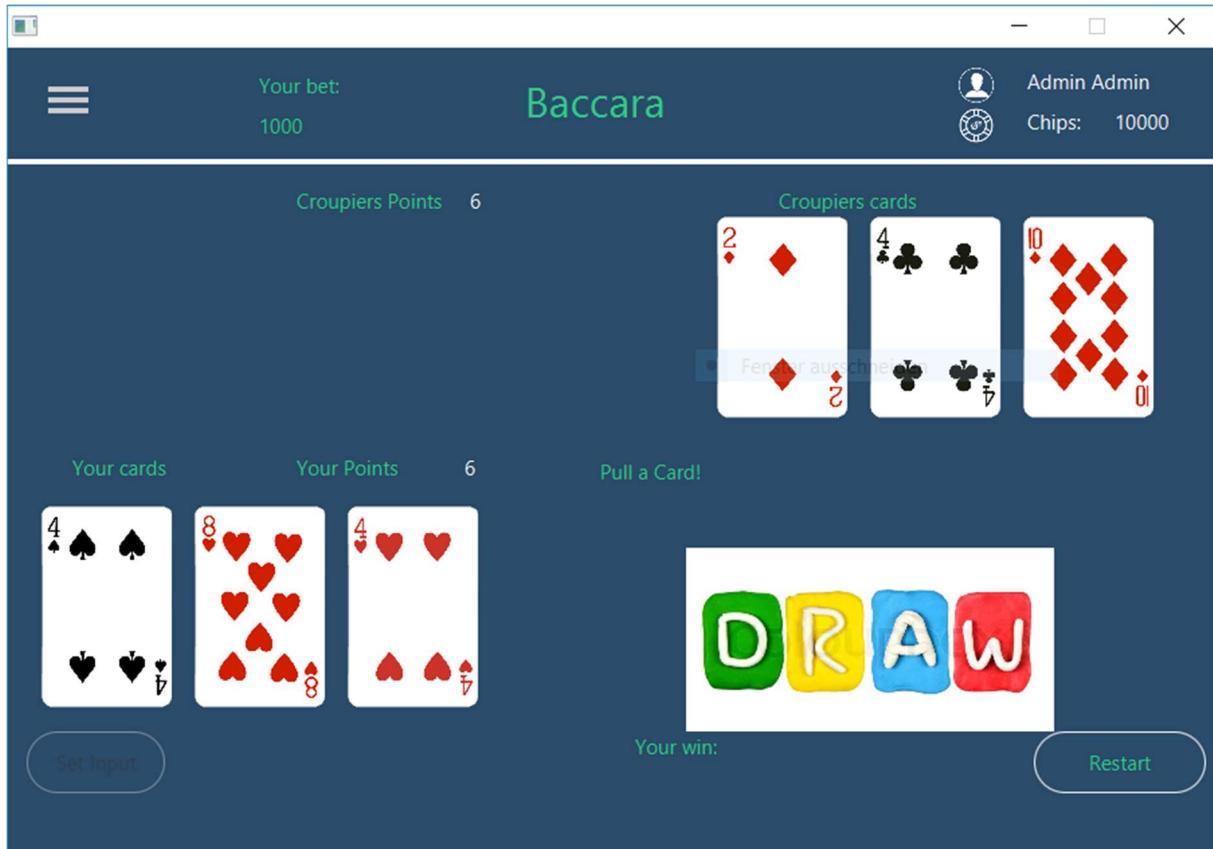


Abbildung 55: in game Baccara

Wenn der Spieler keine weitere Karte ziehen darf/kann/will dann wird nachher ausgerechnet, wer gewonnen hat wer verloren hat. Wenn man gewinnt wird der Betrag verdoppelt auf den Account gutgeschrieben. In einem Label wird angezeigt wieviel man gewonnen hat. Auch ein Bild wie hier wird angezeigt. Beim Unentschieden wird der Betrag wieder auf das Konto gutgeschrieben. Wenn man verloren hat verliert man den Einsatz und ein Bild wird angezeigt.

Stolpersteine:

In diesem Spiel war mein größtes Problem, dass das Spiel nicht ganz nach den Regeln läuft. Nämlich kann der Spieler immer wieder ziehen, wenn er weniger als sechs Punkte hat. Das bedeutet, wenn er die ersten beiden Karten zieht und dann 4 Punkte hat dann erstelle ich so lange neue Karte bis der Spieler mindestens sechs oder mehr Punkte hat.

2.5.2.5 Implementation Yatzy

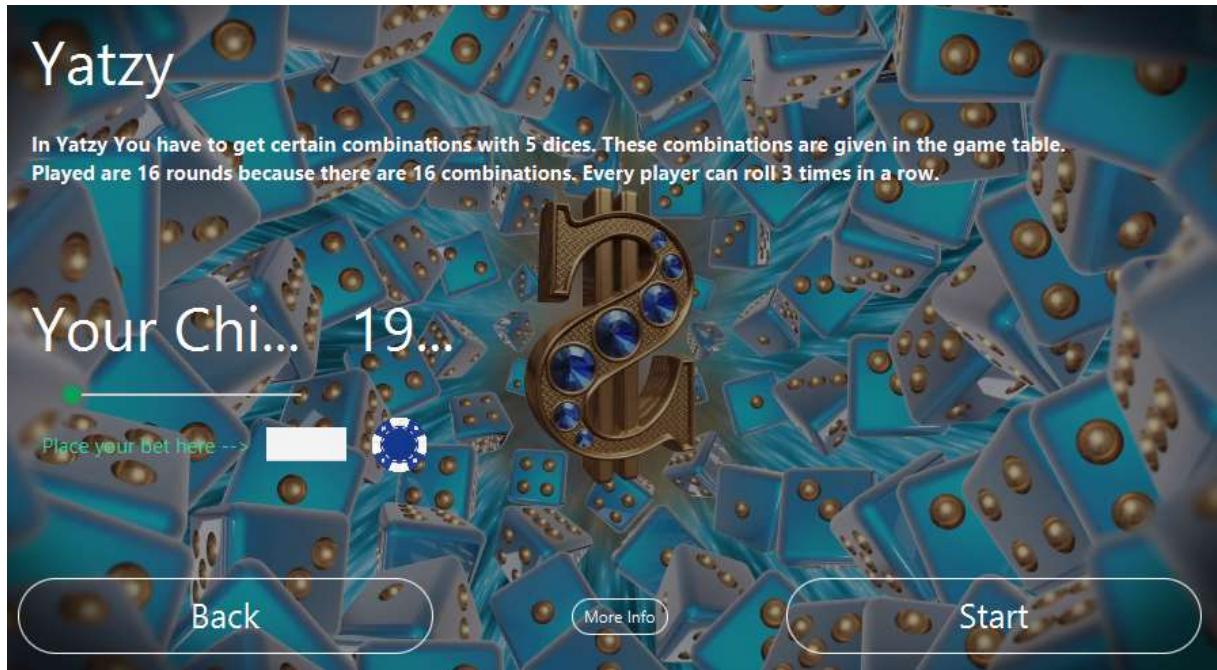


Abbildung 56: Start Screen Yatzy

Der Name des Spiels ist zu sehen und auf der Seite wird das Spiel, mit einer kleinen Beschreibung, beschrieben. Man kann entweder Mehr Informationen zum Spiel erhalten, man kann zurück zu den Spielen, man kann einen Einsatz tätigen und man kann das Spiel starten. Der Einsatz muss mindestens 10 Chips sein und nur ganzzahlige Chips dürfen eingegeben werden. Das habe ich so gemacht, damit der Betrag schon bevor das Spiel startet abgezogen wird, damit wenn der Spieler das Spiel verlässt, der Betrag schon abgezogen wurde.



Abbildung 57: in game Yatzy

Wenn man auf der Titelseite den Betrag eingegeben hat und auf den Button «Start» gedrückt hat, dann wird das Spiel gestartet. Wenn man dann den Button «Roll» drückt, wird gewürfelt. Man kann Würfel auswählen, eine Figur, die man will oder nochmals Würfeln, maximal 2-Mal mehr. Die Figuren werden links angezeigt, ich habe auch eingefügt, dass alle möglichen Figuren dargestellt werden um es dem Nutzer zu erleichtern. Auch Benutzername, Chips und wievielmal man Rollen kann werden gezeigt.

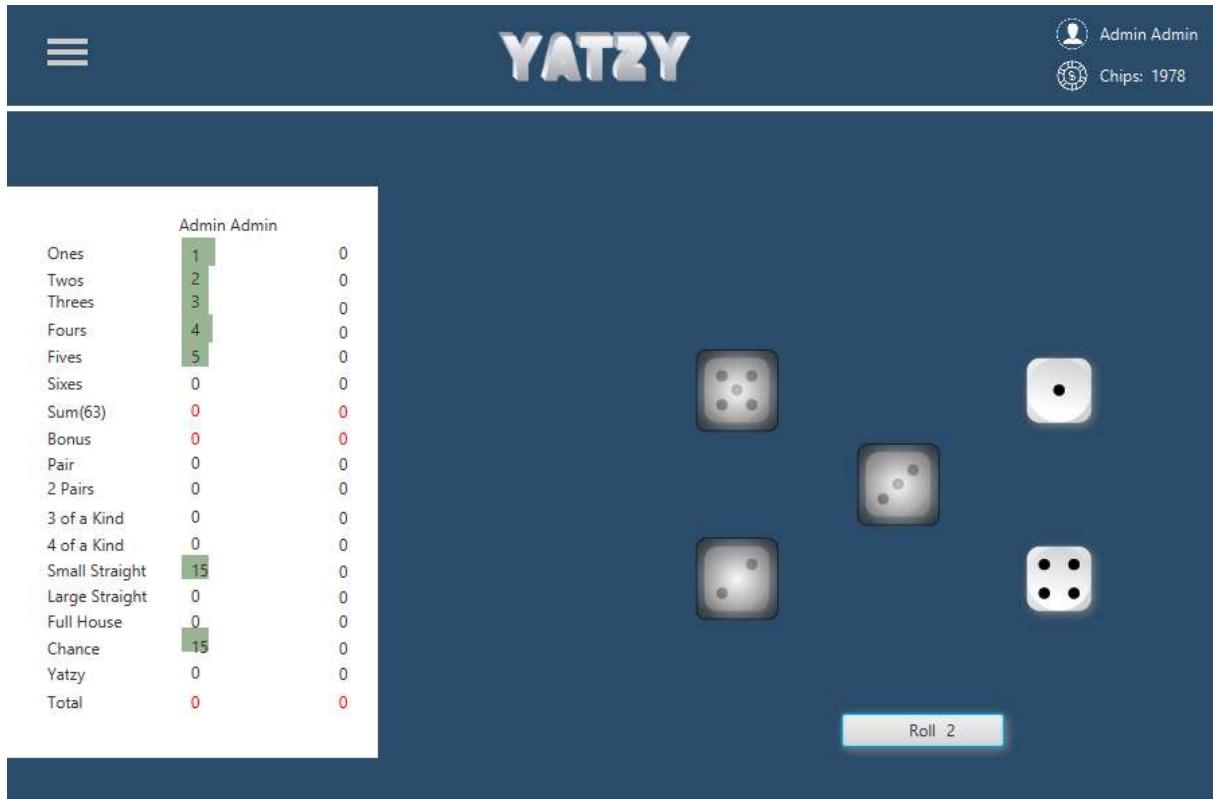


Abbildung 58: in game Yatzy

Wenn man « Roll » gedrückt hat wird gewürfelt. Es werden die fünf Würfel angezeigt, die jeweils zufällig generiert wurden. Die grauen Würfel sind ausgewählt und bleiben so wenn man nochmals « Roll » drücken würde. Links sieht man grün markiert was es für Möglichkeiten gibt mit den Figuren. Wenn die 16 Runden gespielt sind, wird überprüft ob der Spieler über 250 Punkte hat, wenn ja bekommt er das Doppelte von seinem Einsatz zurück über. Wenn er darunter ist wird ihm der Einsatz von seinem Konto abgezogen.

Stolpersteine:

Die grösste Mühe hatte ich mit der Überlegung wie es funktionieren sollte. Was ich damit meine ist, ob es ein zweispielerspiel ist oder ob man gegen einen « Bot » spielt. Ich wagte mich dann an den « Bot », der mir sehr viel Mühe bereitete. Es war sehr schwer zu verstehen wie er umzusetzen ist und was er alles können muss. Am Ende war es doch zu schwer und ich entschied mich dafür, dass man eine Mindestpunktzahl erreichen muss, also doch die zweispielervariante. Bei den Würfeln funktioniert das klicken nicht ordentlich, den Fehler habe ich nicht rausgefunden. Ausserdem habe ich es nicht ganz hingekriegt, dass wenn es ein Unentschieden gibt das Geld zurückgegeben wird und auch direkt neugestartet wird.

2.5.2.6 Implementation Blackjack

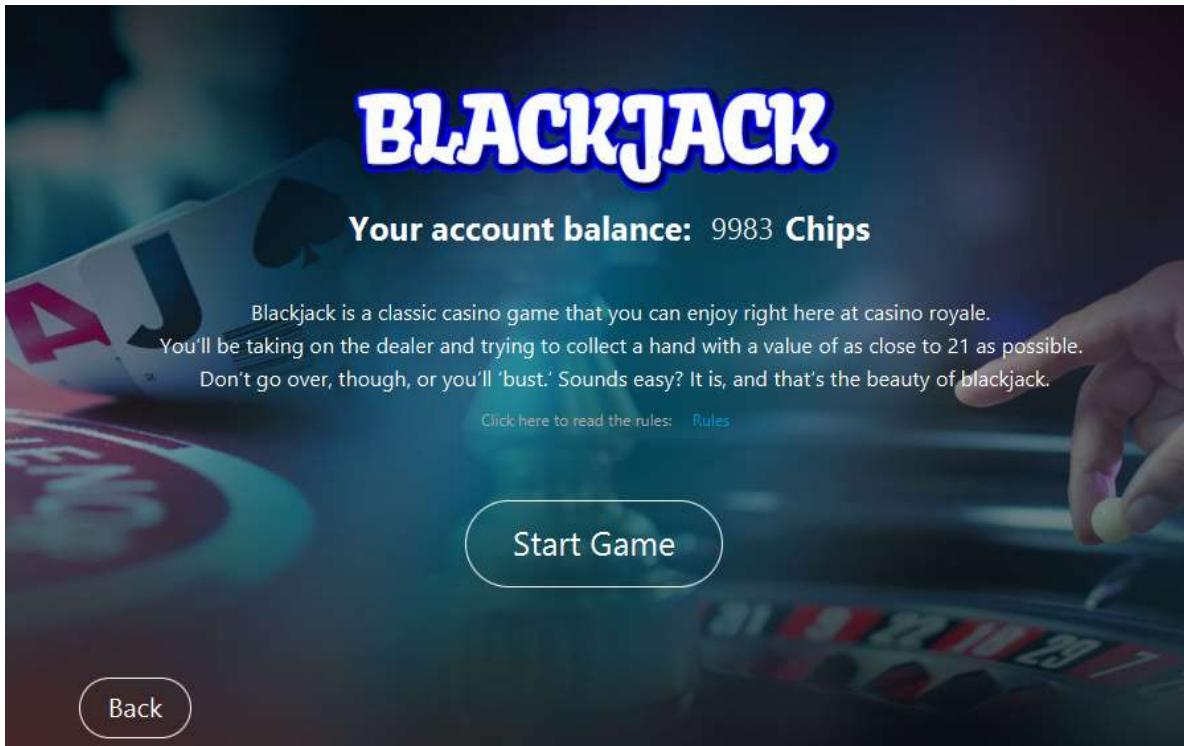


Abbildung 59: Start Screen Blackjack

Der Name des Spiels ist zu sehen und auf der Seite wird das Spiel, mit einer kleinen Beschreibung, beschrieben. Man kann entweder mehr Informationen zum Spiel erhalten, man kann zurück zu der Spielauswahl und man kann das Spiel starten.

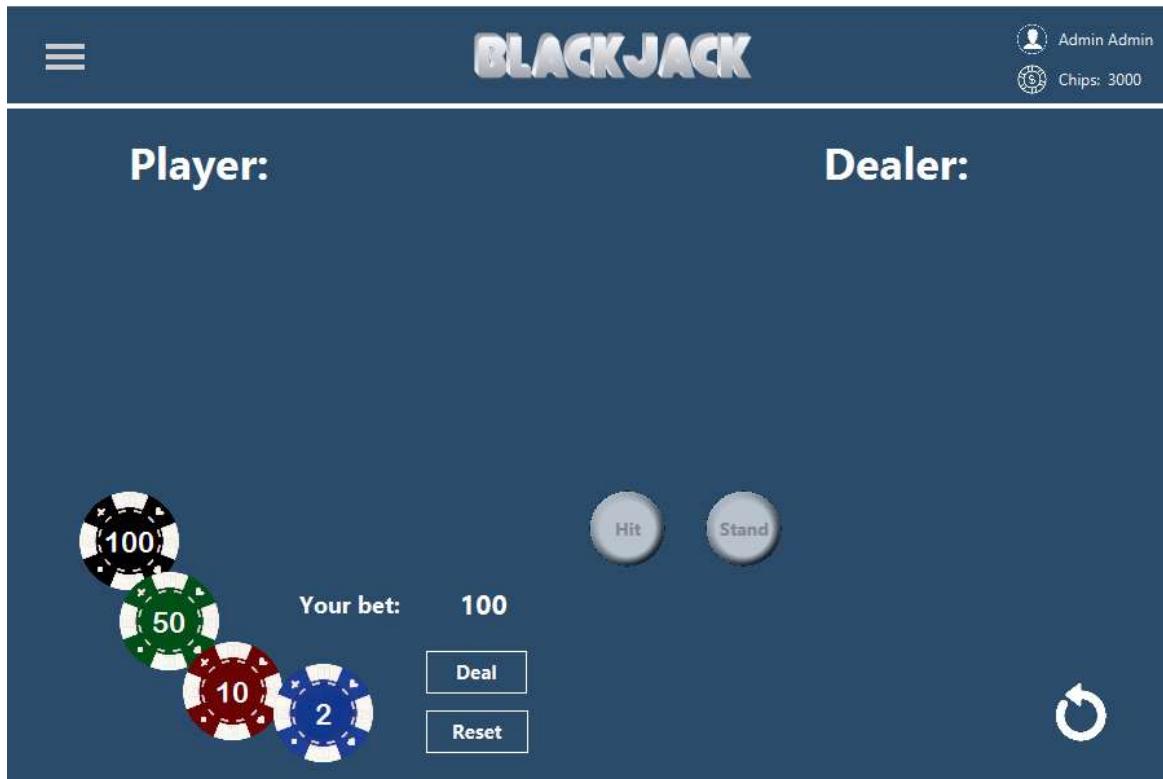


Abbildung 60: in game Blackjack

Man muss einen Einsatz t tigen und man kann das mit dem Button „Deal“ starten. Der Einsatz muss mindestens 10 Chips sein und nur ganzzahlige Chips d rigen eingegaben werden. Der Einsatz darf auch nicht gr sser als der Kontostand sein. Jede Fehleingabe wird mit einer Meldung abgefangen.

Wenn man auf den Button „Deal“ gedr ckt wird der Einsatz gespeichert und man kann es nicht mehr  ndern. Die Buttons Deal und Reset werden auch gesperrt.

Der Einsatz wird von dem Kontostand abgezogen und wenn der Benutzer gewinnt, erh lt er das gewonnene Betrag mit dem Einsatz zur ck.

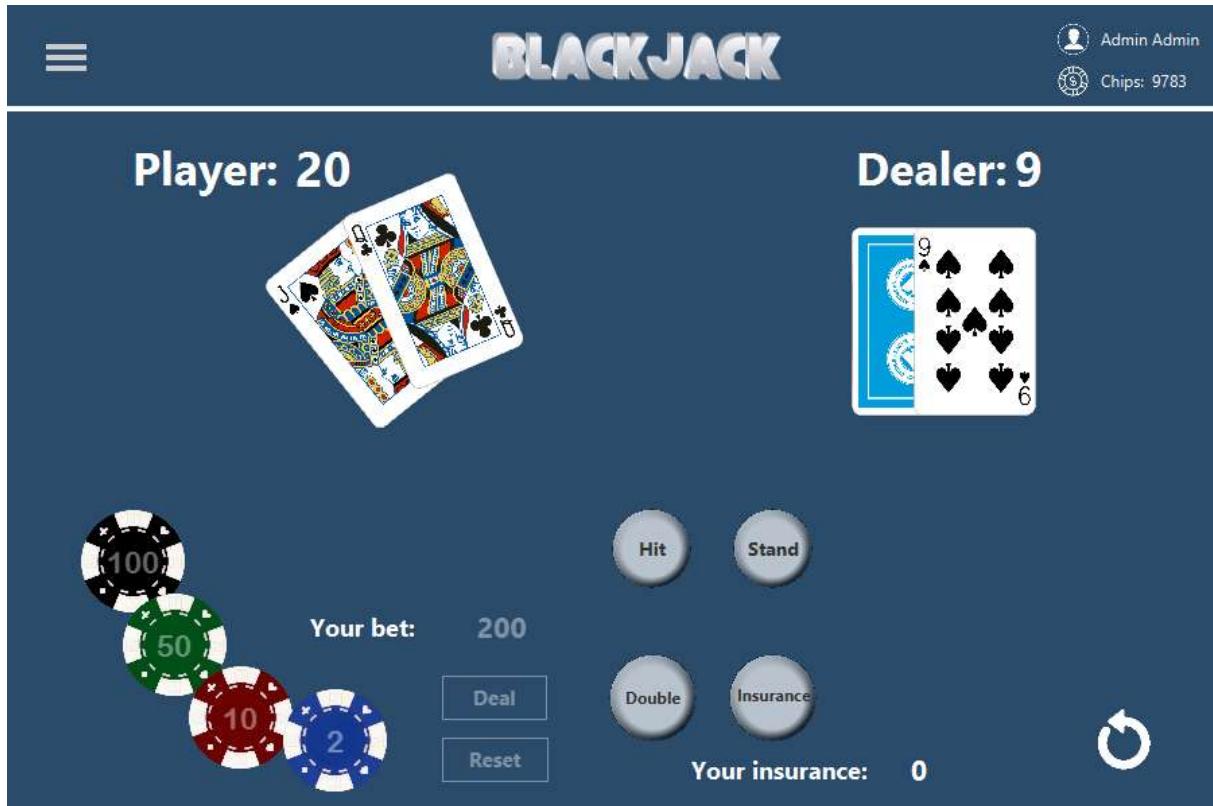


Abbildung 61: in game Blackjack

Die Karten werden dargestellt, die zweite Karte des Dealers bleibt gedeckt. Mit dem Button Stand kann man aufhören eine weitere Karte zu ziehen. Dann wird geprüft, ob der Spieler gewonnen hat oder nicht. Dieser Button ruft eine Methode auf, welche es ausführt.

Hier ist der Code dazu:

```
/*Static weil wir vom BlackjackController aus auf diese Methode zugreifen müssen, jedoch werden keine Attribute der Blackjack Klasse benötigt.
public static String[] compareCard(ArrayList<Card> pointsComputer, ArrayList<Card> pointsGamer) {
    /*Hier haben wir ein Array, in welcher das Faktor für den Gewinn/Verlust benutzt wird um ein Text dazu.
    Weil wir hier wieder vom vom BlackjackController aus auf diese Methode zugreifen müssen, musste ich dies in einem Array speichern.*/
    String[] output = new String[2];
    if (getValue(pointsComputer) == 21) {
        output[0] = "0";
        output[1] = "Dealer has Blackjack!";
    } else if (getValue(pointsGamer) == 21) {
        output[0] = "1.5";
        output[1] = "You have Blackjack!";
    } else if (getValue(pointsComputer) == getValue(pointsGamer)) {
        output[0] = "1";
        output[1] = "It's a draw!";
    } else if (getValue(pointsComputer) > getValue(pointsGamer) && getValue(pointsComputer) < 21) {
        output[0] = "0";
        output[1] = "Dealer has won!";
    } else if (getValue(pointsComputer) > 21) {
        output[0] = "2";
        output[1] = "Dealer is busted!";
    } else if (getValue(pointsGamer) > 21) {
        output[0] = "0";
        output[1] = "You are busted!";
    } else {
        output[0] = "2";
        output[1] = "You have won";
    }
    return output;
}
```

Abbildung 62: Code Blackjack

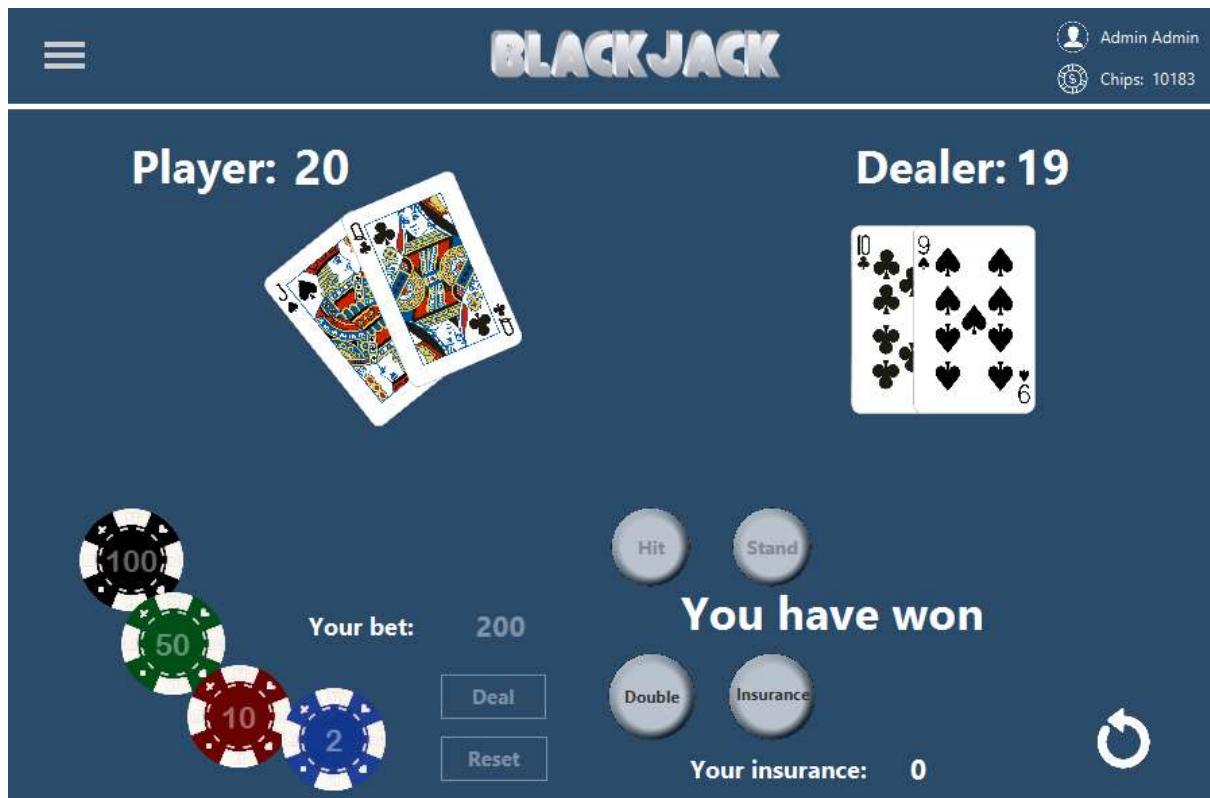
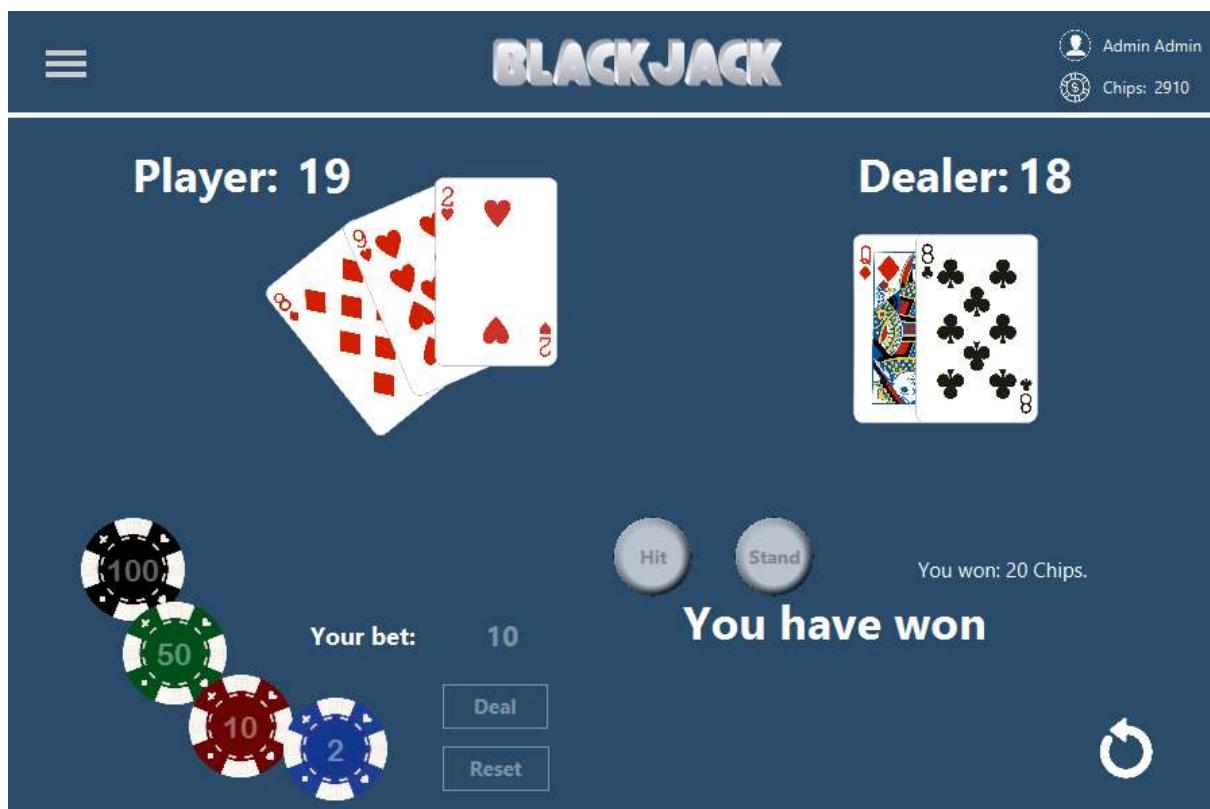


Abbildung 63: in game Blackjack

Ob er dann gewonnen hat oder nicht wird, dann in einer Textform unten rechts ausgegeben. Wenn er gewonnen hat wird der gewonnene Betrag auch unten rechts angezeigt.



Freitag, 4. Mai 2018

Abbildung 64: in game Blackjack

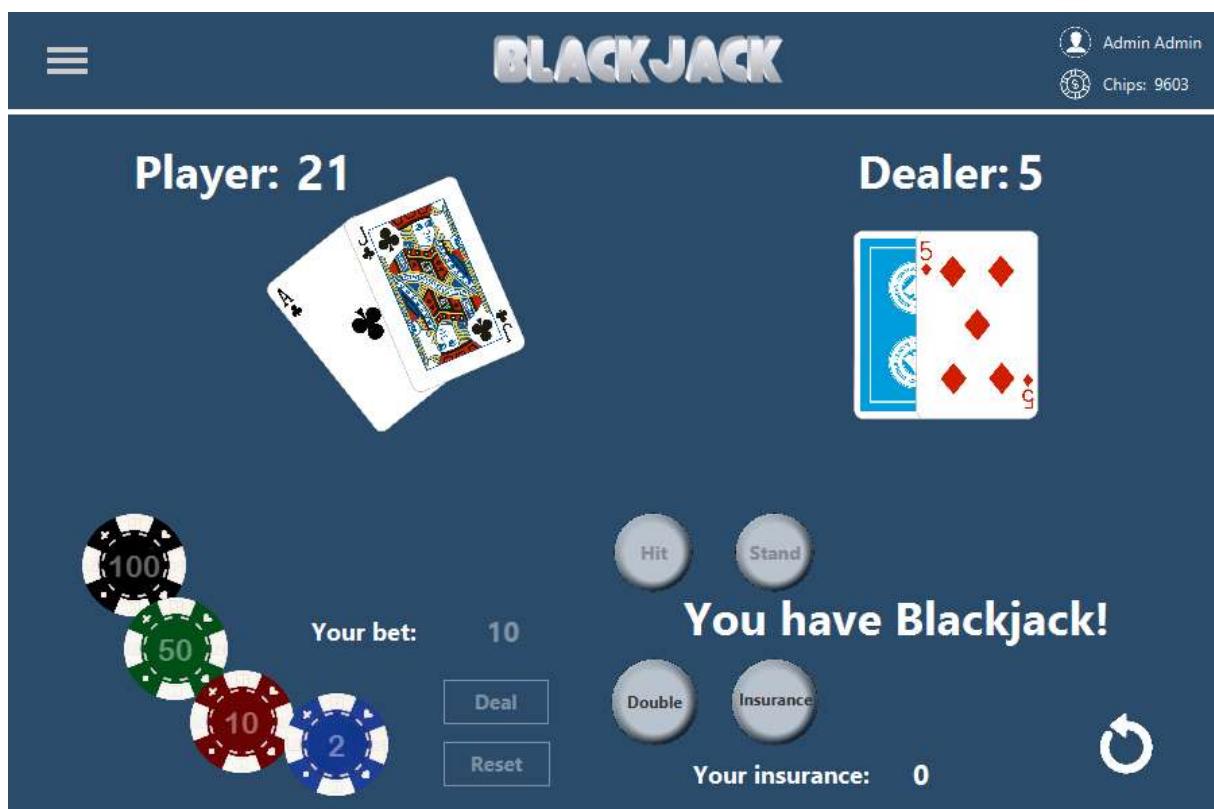


Abbildung 65: in game Blackjack

Wenn der Benutzer direkt mit dem Deal Button ein Blackjack zieht, bleibt die zweite Karte vom Dealer gedeckt. Somit hat der Spieler mit einem Blackjack direkt gewonnen.

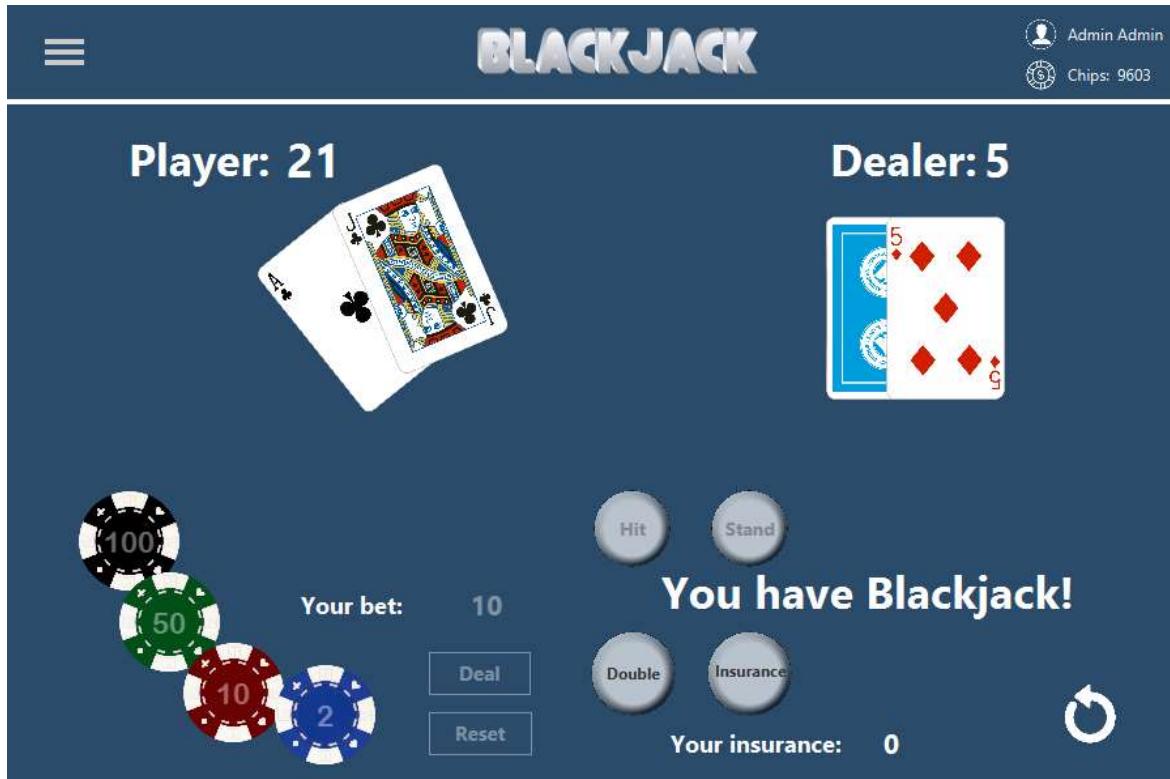


Abbildung 66: in game Blackjack

Hier habe ich noch den Code wo das Deck der Karten kreiert wird:

```
/*Hier wird eine endliche Kartendeck kreiert.*/
public void createDeck() {
    int value;
    for (int i = 0; i < cardTypes.length; i++) {
        for (int j = 0; j < cardNames.length; j++) {
            if (j < 10) {
                /*Alles was unter 10 ist wird ein Wert gegeben. Das Wert wird mit der Position von der ArrayList + 1 gerechnet.*/
                value = j + 1;
            } else {
                /*Sonst wird den Karten das Wert 10 gegeben.*/
                value = 10;
            }
            /*Die Karten werden im ArrayList cards abgespeichert*/
            cards.add(new Card(value, cardNames[j], cardTypes[i]));
        }
    }
}
```

Abbildung 67: Code Blackjack

Array erstellt um die Typen der Karte und die Zahl der Karten zu speichern:

```
private String[] cardTypes = {"H", "D", "S", "C"};
private String[] cardNames = {"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "J", "Q", "K"};
```

Abbildung 68: Code Blackjack

Stolpersteine

Ich hatte am Anfang nur mit kleinen Problemen zu kämpfen, die ich mit ein bisschen herumprobieren lösen konnte. Zum Beispiel wusste ich nicht, wie ich die Karten auf der Spieloberfläche aufzeigen konnte. Schlussendlich konnte ich dies dann lösen.

Doch gegen den Schluss tauchte mir das Problem mit dem 3:2 Gewinn auf. Da unser ganzes Programm mit Integer (Ganze Zahlen) gelöst wurde, konnte ich das Multiplizieren mit 1,5 bei den ungeraden Zahlen nicht umsetzen. Denn es hat das Zahl nach dem Komma herausgeschnitten.

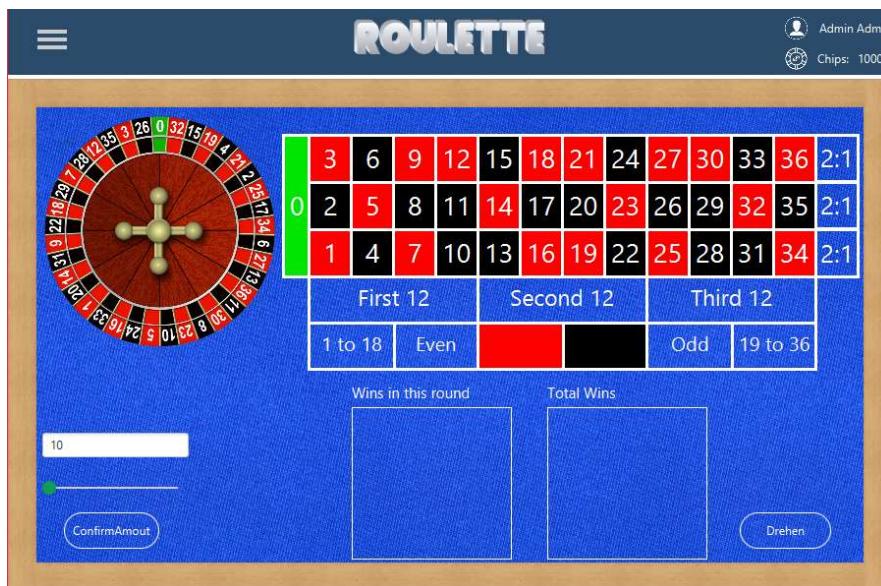
Als ich mit meinem Experten, Herrn Fiechter, darüber gesprochen habe, fiel mir ein Lösungsvorschlag ein. So kam ich auf die Idee nur die 10-er Reihen Zahlen zu verwenden. Doch nach langen Überlegungen fiel mir ein, dass sich auch die 2-er Reihen Zahlen mit 1,5 multiplizieren lassen. So habe ich das umgesetzt. Das Zahl beim Einsatz muss sich durch zwei dividieren können, sonst funktioniert es nicht. Herr Fiechter war mit diesem Lösungsvorschlag auch einverstanden.

2.5.2.7 Implementation Roulette



Abbildung 69: Start Screen Roulette

Hier sieht man die Übersichtsseite zum Spiel Roulette. Mit dem Titel und dem Kontostand, wie in den Anforderungen gewünscht. Der Button «Back» geht man zur Spielübersichtsseite und mit dem Button «Spielen» kommt man zum Spiel. «Game rules» ist für genaue Informationen zum Spiel Roulette.



Hier die eigentliche Spiel Maske. Es ist hier alles vorhanden wie im richtigen Casino, es gibt ein Roulette Rad und einen Tisch auf welchem der Benutzer sogar Chips setzen kann. Benutzername und Kontostand werden auch angezeigt.

Abbildung 70: Roulette

Freitag, 4. Mai 2018

Zuerst aber wählt der Spieler im Textfeld unten links einen Einsatz aus. Dazu kann er auch, den sich darunter befinden, Slider benutztten. Mit «Confirm Amount» kann man sein Geldeinsatz bestätigen.

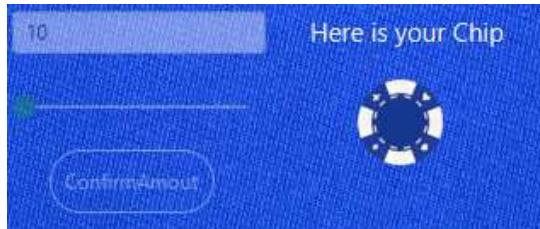


Abbildung 71: Chip Roulette

Nach den obigen Schritten erscheint ein Chip. Diesen kann man mit der Maus packen.



Abbildung 72: Roulette

Wenn man den Chip aufgenommen hat kann man ihn auf ein gewünschtes Feld ziehen, wie in diesem Bild angezeigt. Der Spieler kann so viele Einsätze tätigen wie er möchte so lange er möchte. Mit dem Button «Drehen» beendet er aber seinen Zug. Dann dreht sich das Rad.

Freitag, 4. Mai 2018

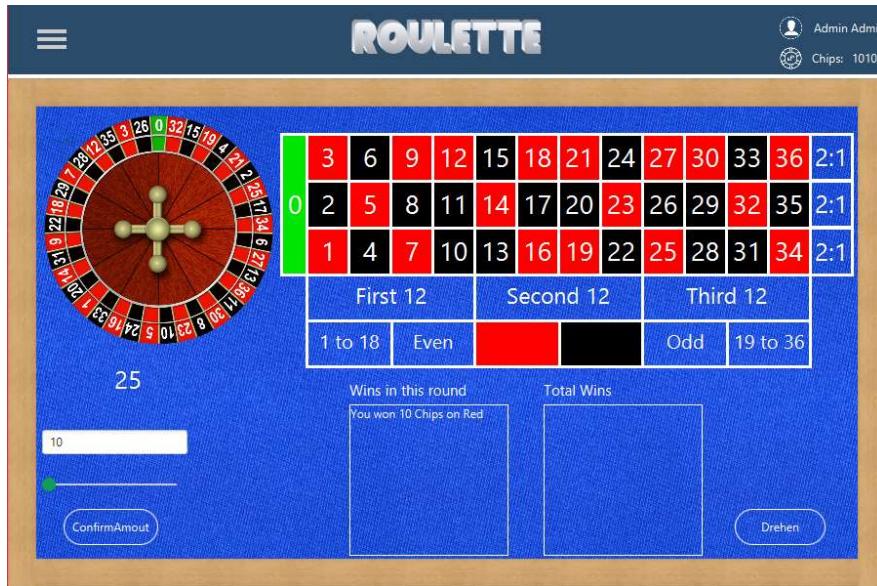


Abbildung 73: Roulette

Nach dem Drehen wird eine zufällige Glückszahl unter dem Rad angezeigt. Im Test habe ich auf Schwarz gesetzt und es kam 25 heraus. 25 ist eine schwarze Zahl, darum hat der Spieler gewonnen es wird ihm im Feld «Wins in this rounds» angezeigt.



Abbildung 74: Roulette

Wenn man nochmals dreht werden die Gewinne der vorherigen Runde in die Tabelle «Total Wins» gespeichert werden.

```
public void auswertung(int magicnumber, ArrayList<RoulettePlacedChips> placedChips) throws SQLException {
    System.out.println("\n" + magicnumber + "\n");
    gewinn = 0;
    placementWon.clear();

    for (RoulettePlacedChips rc : placedChips) {
        switch (rc.getChipName()) {

            case "Straight Up Bets":

                case "0":
                    if (magicnumber == 0) {
                        gewinn += rc.getAmount() * 35;
                        placementWon.add(new RoulettePlacedChips(rc.getChipName(), rc.getAmount() * 35));
                        winCounter++;
                        updateWin();
                        winUpdate = actions.getWinRoulette() + gewinn;
                        updateGewinn();
                        winUpdate = 0;
                    }
                    break;
                case "11":
```

Abbildung 75: Code Roulette

Ich habe für die Logik eine extra Klasse «Roulette» erstellt. Hier wird in der Methode «auswerten» mit einem switch Case geschaut, ob der Spieler auf die zufällige Zahl gesetzt hat. Wenn das Zutrifft wird das Feld in eine Liste übergeben. Diese wird nachher auf dem GUI wieder ausgegeben.

Stolpersteine:

Ich hatte manch ein Problem in meinem Programm, welche ich aber immer lösen konnte. Am Anfang war die Schwierigkeit, das Drag and Drop zu implementieren. Ich habe mich aber damit auseinandergesetzt. Dazu habe ich ein Tutorial von Genuine Coder angeschaut. Danach konnte ich es umsetzen. Danach musste ich für jede mögliche Einsatzvariante ein Image View auf dem Roulette Tisch erstellen. Was mit sehr viel Zeitaufwand verbunden war.

Als das alles gemacht war musste ich die Abfrage erstellen und mit der Datenbank verknüpfen. Was auch wieder mit viel Zeitaufwand verbunden war. Dafür von der Logik her nicht sehr schwierig.

Schlussendlich hatte ich noch ein Problem die richtigen Informationen am richtigen Ort zu haben. Also das hin und her schieben der Daten hat nicht immer tadellos funktioniert.

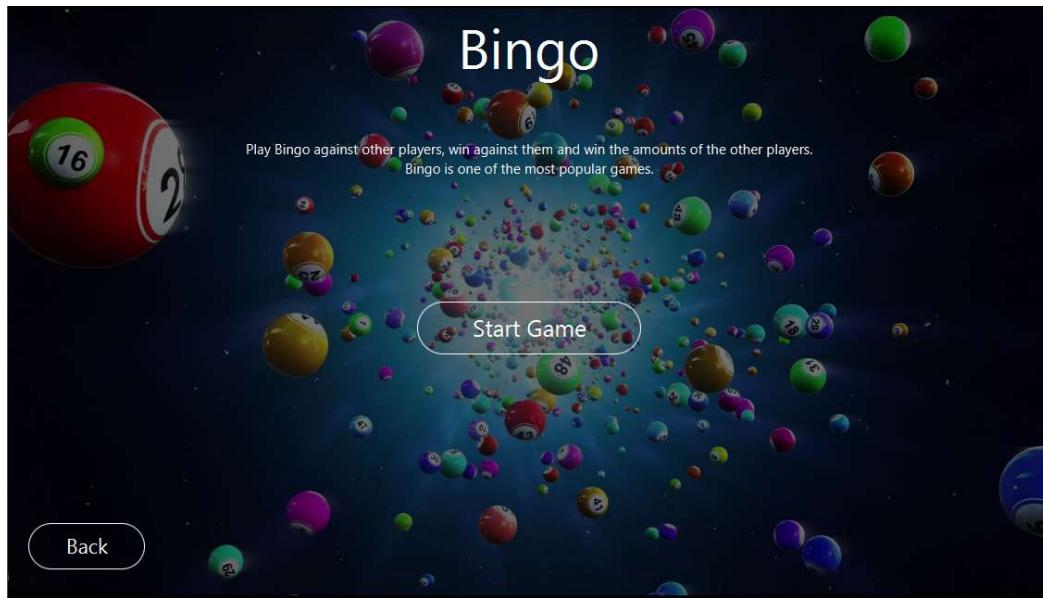


Abbildung 76: Start Screen Bingo

Hier sieht man die erste Übersichtsseite vom Spiel Bingo. Mit dem Button Start Game kommt man zur nächsten Übersichtsseite. Wenn man zurück zur Spielauswahl möchte kann man auf Back drücken.



Abbildung 77: Buy Card Bingo

Bei dieser Übersichtsseite sieht man die Karte, welche man erhält, wenn man auf Buy Card drückt. Weiter erfährt man, wie teuer eine Karte ist, dass die Zahlen von 1 bis 75 ist und die Zahlen zufällig generiert wurden. Ebenfalls gibt es denn Button Back, bei dem man zur ersten Übersichtsseite von Bingo kommt.



Abbildung 78: Bingo

Dies ist das Fenster des Spiels Bingo. Links sieht man die gekaufte Karte mit den gezogenen Zahlen. Über dem Sack sieht man eine Zahl, diese wird alle 30 Sekunden neu gezogen und dann unter Pulled Numbers in das Kästchen geschrieben. Unten rechts sieht man wie viele Gegenspieler man hat und wie diese heißen. Wenn man auf den Button Show Bots drückt werden die Karten der Gegenspieler angezeigt und man sieht wie viele Zahlen sie bereits angekreuzt haben.

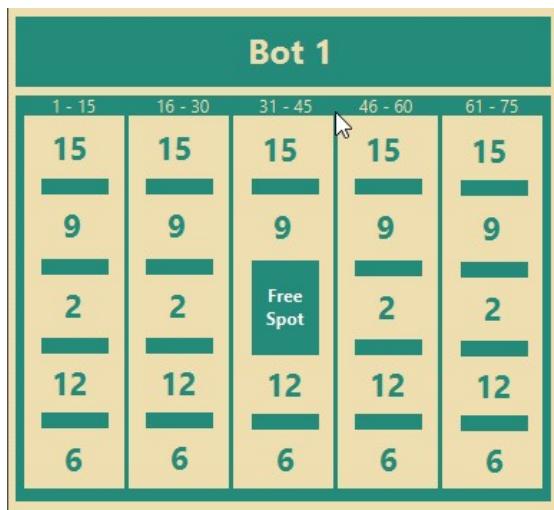


Abbildung 79: Bingo

Dies eines der 5 Fenster von den Bots. Sie zeigen nur 1 Karte und wenn der Bot eine Zahl auf der Karte hat, welche bereits gezogen wurde, erscheint ein rotes Kreuz über der Zahl.

Freitag, 4. Mai 2018

Stolpersteine:

Das einzige Problem beim Spiel Bingo ist, dass meine Bots nicht zuerst Bingo rufen können und der Benutzer nachher noch 15 Sekunden Zeit hat. Bei mir gewinnen die Bots automatisch, nachdem sie 5 Zahlen in einer Reihe haben. Ansonsten gab es keine Probleme.

2.5.2.9 Implementation Statistik

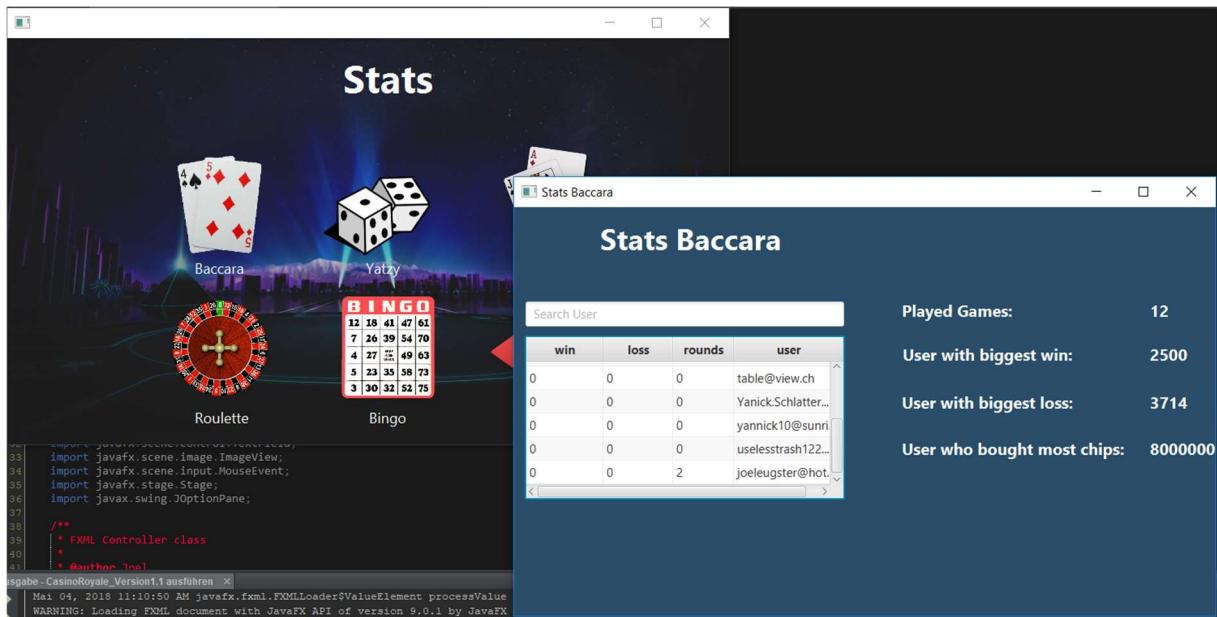


Abbildung 80: Statistik

Zur Statistik haben wir es so gelöst, dass sich der Admin mit dem Benutzernamen „Admin“ und Passwort „Admin“ einloggen kann. Dann ist er direkt auf der Spielauswahl und kann ein Spiel auswählen, klickt auf ein Spiel und dann öffnet sich ein Fenster, bei dem der Admin die Statistiken ansehen kann. In der Datenbank haben wir verschiedene Attribute erstellt, zum Beispiel ein Counter für die Anzahl gespielte Spiele. Diese Variable wird immer um 1 erhöht, wenn ein Spiel gespielt wird. Auch für den grössten Gewinn und den grössten Verlust werden die Werte angezeigt.

2.6 Kontrollieren

2.6.1 Testprotokoll

2.6.1.1 Testprotokoll Lernende

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
1.1	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
1.2	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
1.3	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
1.4	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
1.5	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>

Testbericht Lernende

Ich habe das Programm auf alle Anforderungen getestet und es hat all die gewünschten Anforderungen erfüllt. Es gibt eine Help Funktion und man kann Chips nach Belieben einkaufen. Das war für mich der wichtigste Punkt.

2.6.1.2 Testprotokoll Schule

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
2.1	30.04.2018	funktioniert	Wurde erfüllt	<i>Y. Schlatter</i>
2.2	30.04.2018	funktioniert	Wurde erfüllt	<i>Y. Schlatter</i>
2.3	30.04.2018	funktioniert	Wurde erfüllt	<i>Y. Schlatter</i>
2.4	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
2.5	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>

Testbericht Schule

Die Anforderungen der Schule wurden ebenfalls alle erfüllt. Das Programm besitzt eine Anmeldefunktion sowie auch eine Übersichtsmaske für den Admin.

2.6.1.3 Testprotokoll Programm

Testprotokoll Benutzer an das Programm:

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
3.1	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
3.2	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
3.3	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
3.4	30.04.2018	funktioniert	nicht von jedem Punkt aus kann er sich direkt abmelden.	<i>Y. Schlatter</i>
3.5	30.04.2018	funktioniert	nicht in Ordnung	<i>Y. Schlatter</i>

Testbericht Benutzer

Die von mir getesteten Fälle funktionieren alle, ich konnte alles mühelos machen. Das einzige kleine Problem, das ich hatte, ich konnte mich nicht von überall ausloggen.

Testprotokoll Administrator an das Programm:

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
4.1	30.04.2018	funktioniert	In Ordnung	<i>Y. Schlatter</i>
4.2	30.04.2018	funktioniert	In Ordnung	<i>Y. Schlatter</i>
4.3	30.04.2018	funktioniert	In Ordnung	<i>Y. Schlatter</i>
4.4	30.04.2018	funktioniert	In Ordnung	<i>Y. Schlatter</i>
4.5	30.04.2018	funktioniert	In Ordnung	<i>Y. Schlatter</i>

Testbericht Administrator

Die von mir getesteten Fälle funktionieren alle, ich konnte alles mühelos ausführen. Es gab keinerlei Problem und somit funktioniert das Administrator-Tool einwandfrei.

2.6.1.4 Testprotokoll Casino Royale

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
5.1	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.2	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.3	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.4	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.5	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.6	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.7	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.8	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.9	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.10	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.11	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.12	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.13	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.14	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.15	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.16	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.17	02.05.2018	funktioniert	in Ordnung	J. Eugster
5.18	02.05.2018	funktioniert	in Ordnung	J. Eugster

Testbericht Casino Royale

Es funktioniert alles einwandfrei. Es kann am Kunden weitergegeben werden.

2.6.1.5 Testprotokoll Baccara

Testfall-nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
6.1	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.2	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.3	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.4	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.5	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.6	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.7	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.8	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.9	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.10	02.05.2018	funktioniert	Nicht ganz (nur ganze Zahlen und mindestens 10)	<i>J. Eugster</i>
6.11	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.12	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.13	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.14	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.15	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.16	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.17	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.18	02.05.2018	funktioniert nicht	Gemischt ja, endlich nein	<i>J. Eugster</i>
6.19	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.20	02.05.2018	funktioniert nicht	Betrag wird zu * 2 ausbezahlt	<i>J. Eugster</i>
6.21	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.22	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>

6.23	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.24	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.25	02.05.2018	funktioniert	Nur Button klicken nicht Meldung bestätigen	<i>J. Eugster</i>
6.26	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.27	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>
6.28	02.05.2018	funktioniert	in Ordnung	<i>J. Eugster</i>

Testbericht Baccara

Alles in allem funktioniert das Spiel Baccara. Baccara kann ohne Probleme gespielt werden. Die folgenden Anforderungen wurden nicht oder nicht ganz erfüllt:

1. A.6.10: Man kann den Chips Betrag nur bedingt selber wählen. Mindestens 10 Chips und nur ganze Chips also nicht 1,5 Chips.
2. A.6.18: Es gibt verschiedene Karten, die aber nicht endlich sind, denn sie werden Random erstellt.
3. A.6.20: Der Betrag wird zu * 2 ausbezahlt und nicht 2:3.
4. A.6.25: Man muss nur auf den Button klicken und nicht noch eine Meldung betätigen.

Das Spiel läuft auch ohne diese Anforderungen. Es sind kleine Fehler, die noch behoben werden könnten. Wir haben allerdings nicht genug Zeit.

2.6.1.6 Testprotokoll Yatzy

Testfall- nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
7.1	30.04.2018	funktioniert	in Ordnung	R. Meier
7.2	30.04.2018	funktioniert	in Ordnung	R. Meier
7.3	30.04.2018	funktioniert	in Ordnung	R. Meier
7.4	30.04.2018	funktioniert	in Ordnung	R. Meier
7.5	30.04.2018	funktioniert	in Ordnung	R. Meier
7.6	02.05.2018	funktioniert	in Ordnung	R. Meier
7.7	02.05.2018	funktioniert	in Ordnung	R. Meier
7.8	02.05.2018	funktioniert	in Ordnung	R. Meier
7.9	02.05.2018	funktioniert	in Ordnung	R. Meier
7.10	02.05.2018	funktioniert	in Ordnung	R. Meier
7.11	02.05.2018	funktioniert	in Ordnung	R. Meier
7.12	02.05.2018	funktioniert	in Ordnung	R. Meier
7.13	02.05.2018	funktioniert	in Ordnung	R. Meier
7.14	02.05.2018	funktioniert	in Ordnung	R. Meier
7.15	02.05.2018	funktioniert	in Ordnung	R. Meier
7.16	02.05.2018	funktioniert	in Ordnung	R. Meier
7.17	02.05.2018	funktioniert	in Ordnung	R. Meier
7.18	02.05.2018	funktioniert	in Ordnung	R. Meier
7.19	02.05.2018	funktioniert	in Ordnung	R. Meier
7.20	02.05.2018	funktioniert	in Ordnung	R. Meier
7.21	02.05.2018	funktioniert	in Ordnung	R. Meier
7.22	02.05.2018	Funktioniert nicht	Ich konnte die Funktion nicht einfügen	R. Meier

7.23	02.05.2018	Funktioniert nicht	Ich konnte die Funktion nicht einfügen	<i>R. Meier</i>
7.24	02.05.2018	Funktioniert nicht	Ich konnte die Funktion nicht einfügen	<i>R. Meier</i>
7.25	02.05.2018	funktioniert	in Ordnung	<i>R. Meier</i>
7.26	02.05.2018	funktioniert	in Ordnung	<i>R. Meier</i>
7.27	02.05.2018	funktioniert	in Ordnung	<i>R. Meier</i>

Testbericht Yatzy

Alles funktioniert einwandfrei bis auf die Testfälle 7.22, 7.23, 7.24. Diese konnte ich nicht richtig implementieren. Es kann nicht direkt neu gestartet werden. Dazu wird der Betrag nicht zurückgegeben und bleibt gesetzt. Ansonsten ist das Programm abgabebereit.

2.6.1.7 Testprotokoll Blackjack

Testfall- nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
8.1	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.2	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.3	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.4	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.5	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.6	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.7	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.8	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.9	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.10	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.11	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.12	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.13	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.14	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.15	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.16	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.17	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.18	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.19	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.20	30.04.2018	funktioniert teilweise	mangelnd	A. Sharushan
8.21	30.04.2018	funktioniert	in Ordnung	A. Sharushan
8.22	30.04.2018	funktioniert	in Ordnung	A. Sharushan

8.23	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.24	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.25	30.04.2018	funktioniert teilweise	mangelnd	<i>A. Sharushan</i>
8.26	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.27	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.28	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.29	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>
8.30	30.04.2018	funktioniert	in Ordnung	<i>A. Sharushan</i>

Testbericht Blackjack

Alles in allem funktioniert dieses Spiel. Blackjack kann ohne Probleme gespielt werden.

Einige Anforderungen wurden teilweise nicht erfüllt:

- **Testfall-Nr.: 8.20**
Wenn es ein Unentschieden gibt, erscheint eine Meldung. Aber man kann diese Meldung nicht bestätigen, sondern muss selber neustarten.
- **Testfall-Nr.: 8.25**
Der Spieler kann einen Betrag setzen, aber der Spieler kann nicht mit einem Klick auf den Versicherungsbutton den Betrag setzen.

Diese Probleme sind keine Behinderungen für das Spiel. Das Spiel kann aber ohne Probleme gespielt werden. Die Anforderungen, die nicht erfüllt sind, sind nur kleine Erweiterungen, um das Spiel umgänglicher zu machen.

2.6.1.8 Testprotokoll Roulette

Testfall- nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
9.1	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.2	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.3	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.4	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.5	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.6	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.7	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.8	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.9	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.10	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.11	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.12	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.13	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.14	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.15	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.16	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.17	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.18	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.19	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.20	30.04.2018	funktioniert	in Ordnung	<i>Y. Schlatter</i>
9.21	30.04.2018	funktioniert nicht	Wenn ich etwas Gewinne wird mir der Betrag nicht angezeigt.	<i>Y. Schlatter</i>

Testbericht Roulette

Freitag, 4. Mai 2018

Alles in allem funktioniert dieses Spiel. Roulette kann gespielt werden. Das einzige Problem ist, dass im Moment der Gewinn, nicht richtig mit der Datenbank kommuniziert wird.

2.6.2.9 Testprotokoll Bingo

Testfall- nummer	Datum	Resultat	Bemerkung	Tester/ Unterschrift
10.1	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.2	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.3	02.05.2018	funktioniert	Es gibt 2 Übersichtsseiten, auf der ersten sieht man den Betrag nicht, auf der zweiten schon	<i>Y. Meyner</i>
10.4	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.5	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.6	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.7	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.8	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.9	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.10	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.11	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.12	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.13	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.14	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.15	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.16	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.17	02.05.2018	funktioniert nicht	der Gegenspieler kann kein Bingo anmelden	<i>Y. Meyner</i>
10.18	02.05.2018	funktioniert nicht	sobald der Gegenspieler 5 in einer Reihe hat, hat er gewonnen, er muss nicht zuerst Bingo rufen	<i>Y. Meyner</i>

10.19	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.20	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.21	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>
10.22	02.05.2018	funktioniert	in Ordnung	<i>Y. Meyner</i>

Testbericht Bingo

Das Programm hat keine Fehler, keine Bugs und ist im Ganzen gut spielbar. Einzig was beim Programm fehlt ist, dass der Gegenspieler nicht «Bingo» rufen kann, sondern automatisch gewinnt, wenn er 5 Zahlen in einer Reihe hat.

2.6.3 Unitest / Testszenario

Aus zeitlichen Gründen konnten wir keine Unit-Test und Testszenarien machen. Wir haben sie im Zeitplan eingeteilt, aber haben in anderen Bereichen zu viel Zeit investiert unter anderem in den Testfallspezifikationen, die wir sehr detailliert gemacht haben.

2.7 Auswerten

2.7.1 Kurzbericht IPA

2.7.1.1 Gruppen Reflexion

Als wir die Gruppen zugeteilt bekamen haben wir unsere Ziele vereinbart. Wichtig war es vor allem, dass wir sie immer im Hinterkopf behalten. Es hat ganz gut funktioniert, wir konnten gut nach IPERKA arbeiten, haben alles Step-by-Step gemacht, wenn jemand in der Gruppe Hilfe brauchte bekam er die auch und Probleme konnten wir immer in der Gruppe lösen. Dem Zeitplan kamen wir nicht nach, da wir bei einzelnen Punkten dann doch mehr oder auch weniger Zeit einplanten. Schlimm ist es nicht da es doch Zeitlich aufging und er hat uns zum Orientieren sehr geholfen.

Die Spiele Programmieren hat grundsätzlich gut funktioniert, wir haben nur vereinzelt Fehler. Beim Spiel von Yannick Meyner würde er gerne noch mehr Karten ziehen können, momentan ist es eine. Joel hat auch Probleme mit dem ziehen einer Karte. Beim Ziehen der dritten Karte ist nicht alles Regel Komfort. Bei Yannick Schlatter funktioniert alles wie er es wollte. Er hätte sich nur gerne besser informiert bevor er mit dem Programmieren begonnen hat. Sharushan hat einige Dinge, die nicht funktionieren die er gerne verbessert hätte. Darunter sind die Beträge, die manchmal nicht angezeigt werden. Rafael hatte Probleme mit dem Bot den er schlussendlich nicht fertiggekriegt hat, aber gerne hätte. Dazu können die Spieler, wenn sie nicht am Zug sind beim anderen reinklicken und die Berechnungen werden Falsch.

Rafael war 4 Tage krank was das Arbeiten aufgewühlt hat. Er konnte nicht so viel machen wie er gerne wollte. Ansonsten hat alles gut funktioniert. Die Zusammenarbeit war gut, es wurde immer sinnvoll aufgeteilt was zu tun war und die Absprache ist gut gelungen.

Besser zu machen wäre der Zeitplan, vor allem genauer und logischer. Es wäre vermutlich so noch besser gelaufen. Dazu noch mehr Zeit ins Planen zu investieren hätte sich gelohnt.

Für unsere Gruppe war es sicher eine positive Erfahrung. Wir haben alle viel mitgenommen. Sei es bei der Kommunikation in der Gruppe, eine Dokumentation zu verfassen oder auch ein wenig wie die Arbeitswelt aussehen könnte. Das Team konnte ihre Javakenntnisse erweitern durch das teils intensive Programmieren. Wir alle haben gelernt Aufträge sinnvoll aufzuteilen und somit das Zusammenarbeiten verbessert. Für uns alle sehr positiv ist das Kennenlernen von dem IDAF Projekt, durch dies können wir uns besser vorbereiten und wissen auch was auf uns zukommt.

Um Zukünftige Probleme zu bewältigen würden wir sicherlich uns Ziele aufschreiben damit wir vor Augen haben was zu beachten ist. Dazu diese Dokumentation zur Hand haben um nochmals Nachzuschauen, wenn etwas unklar ist.

2.7.1.2 Fazit

Die Programmierwochen waren super. Wir konnten sehr viel lernen. Auch Open HPI war sehr gut, denn alles zuerst erklärt bekommen und nachher selber testen ist wirklich eine sehr gute Idee mit viel Lernförderlichkeit. Das zusammenarbeiten in der Gruppe hatte uns auch gefallen. Wenn wir Probleme hatten mussten wir das nicht immer mit den Lehrern klären, sondern konnten es in der Gruppe lösen. Das ist ziemlich wichtig für die Zukunft, denn man arbeitet vielleicht später im Team und man muss das können. Auch mit Sharushan als Projektleiter war sehr gut, denn er hat uns immer gesagt, wer was machen soll. Er hat die Aufgaben sehr gut verteilt, alle haben etwa gleich viel gearbeitet. Wir würden daher die Programmierwochen für alle weiterempfehlen.

2.7.1.3 Leistungsbereitschaft

Zu den Aufgaben, die von uns gefordert sind, haben wir auch zusätzliche Dinge gemacht. Anforderungsanalysen wurden ausgebaut, spezifisch führten wir persönliche Interessen ein. Wir erstellten eine Shop-Seite, die nicht gefordert war. Um mehr Übersicht zu schaffen kreierten wir eine Menubar die jeder in seinem Spiel nutzt und zusätzlich noch verschiedene Methoden wie Slider, Normale Textfelder und Chips die man klicken kann um Geld zu setzen. Yannick führte eine Sicherheitseinstellung ein in der Datenbank um Brute-Forces zu verhindern, dazu noch eine E-Mail Verifizierung damit es nicht zu viele Einträge in der Datenbank gibt.

Abbildungsverzeichnis

Abbildung 1: Das Team	8
Abbildung 2: Zeitplan	9
Abbildung 3: IPERKA	29
Abbildung 4: Login Screen Casino Royale	46
Abbildung 5: Sitemap-Navigation, alle Screens.....	48
Abbildung 6: Use-Case Casino Royale	49
Abbildung 7: Use-Case Baccara.....	50
Abbildung 8: Use-Case Yatzy	51
Abbildung 9: Use-Case Blackjack.....	52
Abbildung 10: Use-Case Roulette	53
Abbildung 11: Use-Case Bingo.....	54
Abbildung 12: Use-Case Admin.....	55
Abbildung 13: UML Casino Royale	62
Abbildung 14: UML Baccara	63
Abbildung 15: UML Yatzy	64
Abbildung 16: UML Blackjack	65
Abbildung 17: UML Roulette.....	66
Abbildung 18: UML Bingo	67
Abbildung 19 und 20: Design Login.....	68
Abbildung 21 und 22: Design Home	69
Abbildung 23 und 24: Design Shop	70
Abbildung 25 und 26: Design Spielauswahl.....	71
Abbildung 27 und 28: Design Baccara	72
Abbildung 29 und 30: Design Yatzy.....	73
Abbildung 31 und 32: Design Baccara	74
Abbildung 33 und 34: Design Roulette	75
Abbildung 35 und 36: Design Bingo	76
Abbildung 37 und 38: PAP mit Datenbank.....	79
Abbildung 39: Konzeptionelles Datenbankmodell.....	80
Abbildung 40: Logisches Datenbankmodell.....	81
Abbildung 41: Code Datenbank (Action)	82
Abbildung 42: Datenbank	82
Abbildung 43: Entscheidungsmatrix Branches	173
Abbildung 44: Entscheidungsmatrix Pattern	174
Abbildung 45: Login Casino Royale.....	175
Abbildung 46: Sign Up Casino Royale.....	176
Abbildung 47: Forgot Password.....	177
Abbildung 48: Datenbank	178
Abbildung 49: Code Datenbank.....	178
Abbildung 50: Code Datenbank.....	178
Abbildung 51: Shop	179
Abbildung 52: Code Shop.....	180
Abbildung 53: Start Screen Baccara.....	181
Abbildung 54: in game Baccara.....	182
Abbildung 55: in game Baccara.....	183

Abbildung 56: Start Screen Yatzy	184
Abbildung 57: in game Yatzy	185
Abbildung 58: in game Yatzy	186
Abbildung 59: Start Screen Blackjack.....	187
Abbildung 60: in game Blackjack.....	188
Abbildung 61: in game Blackjack.....	189
Abbildung 62: Code Blackjack	189
Abbildung 63: in game Blackjack.....	190
Abbildung 64: in game Blackjack.....	191
Abbildung 65: in game Blackjack.....	191
Abbildung 66: in game Blackjack.....	192
Abbildung 67: Code Blackjack	192
Abbildung 68: Code Blackjack	192
Abbildung 69: Start Screen Roulette	194
Abbildung 70: Roulette	194
Abbildung 71: Chip Roulette	195
Abbildung 72: Roulette	195
Abbildung 73: Roulette	196
Abbildung 74: Roulette	196
Abbildung 75: Code Roulette	197
Abbildung 76: Start Screen Bingo.....	198
Abbildung 77: Buy Card Bingo.....	198
Abbildung 78: Bingo	199
Abbildung 79: Bingo	199
Abbildung 80: Statistik	201

Quellenverzeichnis

<http://www.cheval-lab.ch/was-ist-usability/usability-grundlagen/normen-und-richtlinien/iso-9241-110/>

<https://www.gamevelvet.com/yatzy-online/rules>

<http://www.rouletteonline.net/how-to-play-roulette/>

<https://www.blackjackinfo.com/blackjack-rules/>

<https://www.casino.com.de/baccara/regeln>

[https://de.wikipedia.org/wiki/Baccara_\(Gl%C3%BCcksspiel\)](https://de.wikipedia.org/wiki/Baccara_(Gl%C3%BCcksspiel))

<http://archive.space.unibe.ch/en/polymechaniker/ausbildung/iperka-6-schritt-methode.html>
dieser Link gehört zu Abbildung 3: IPERKA

Glossar

Da wir während dem Dokumentieren alle Fachbegriffe erklärt haben, haben wir hier kein Glossar geschrieben. Es würde sich überschneiden.