

PFLICHTENHEFT

---

Go <insert cooler name here>

---

Matthias "Kettensäge"Telöken  
Tobias Faßbender

19. Februar 2018

# Inhaltsverzeichnis

<b>1</b>	<b>Projekt: Go</b>	<b>4</b>
1.1	Zielbestimmung . . . . .	4
1.1.1	Musskriterien . . . . .	4
1.1.2	Wunschkriterien . . . . .	4
1.1.3	Abgrenzungskriterien . . . . .	4
<b>2</b>	<b>Produkteinsatz</b>	<b>5</b>
2.1	Zielgruppen . . . . .	5
2.2	Betriebsbedingungen . . . . .	5
<b>3</b>	<b>Produktumgebung</b>	<b>6</b>
3.1	Software . . . . .	6
3.2	Hardware . . . . .	6
3.3	Orgware . . . . .	7
<b>4</b>	<b>Produktfunktionen</b>	<b>8</b>
4.1	Benutzerfunktionen . . . . .	8
4.1.1	Benutzer-Kennung . . . . .	8
4.1.2	Persönliches Profil . . . . .	8
4.2	Spielfunktionen . . . . .	9
4.2.1	Initialisierung . . . . .	9
4.2.2	Spielverlauf . . . . .	9
4.2.3	Speichern und Laden . . . . .	10
4.3	Analysetool . . . . .	11
4.4	Statistiken . . . . .	12
<b>5</b>	<b>Produktdaten</b>	<b>13</b>
<b>6</b>	<b>Produktleistungen</b>	<b>14</b>
<b>7</b>	<b>Benutzeroberfläche</b>	<b>15</b>
<b>8</b>	<b>Qualitätszielbestimmungen</b>	<b>16</b>
<b>9</b>	<b>Globale Testszenarien und Testfälle</b>	<b>17</b>
<b>10</b>	<b>Entwicklungsumgebung</b>	<b>18</b>
10.1	Software . . . . .	18
10.2	Hardware . . . . .	18



# 1 Projekt: Go

## 1.1 Zielbestimmung

Es soll eine Oberfläche mit verschiedenen Funktionen für ein Go Spiel erstellt werden. Dabei sollen Spiele zu 2 Spielern über Internet möglich sein, sowie das einladen einer KI (z.B. GNU-Go) gegen die gespielt werden kann.

### 1.1.1 Musskriterien

- Das Spiel muss mit 2 Spielern übers Internet Spielbar sein
- Es muss über eine Grafische Oberfläche verfügen, auf der die Spieler mit der Maus die Eingaben machen können
- Das Spiel muss über verschiedene Brett-größen verfügen (9x9, 13x13, 19x19)
- Es muss ein Analysetool vorhanden sein, mit dem beendete Spiele analysiert werden können
- Die Spiele (sowohl fertige als auch aktive) müssen automatisch auf dem Server gespeichert werden und müssen zusätzlich auch lokal gespeichert werden können
- Das Spiel muss von auf mehreren Servern lauffähig sein

### 1.1.2 Wunschkriterien

- Um auch allein Spielen zu können muss ein Interface vorhanden sein mit dem gegen KI's (wie z.B. GNU-Go) gespielt werden kann
- Eine Statistik für Spieler muss erstellt werden und von den Spielern einsehbar sein

### 1.1.3 Abgrenzungskriterien

- Im Rahmen dieses Projekts wird keine Künstliche Intelligenz als gegenspieler programmiert. (Eine solche KI kann in einem weiteren Projekt implementiert und durch das enthaltene KI-Interface eingebunden werden)

## 2 Produkteinsatz

Das Programm dient dazu über das Internet das Brettspiel Go mit einer Grafischen Benutzeroberfläche spielen zu können. Außerdem können mit dem Programm gespielte Spiele analysiert werden.

### 2.1 Zielgruppen

Die Zielgruppen sind Leute, die gerne Go spielen, und häufig nicht die Zeit haben ein ganzes Spiel auf einem 19x19 Feld zu spielen. Dafür werden die Spiele automatisch gespeichert und können jederzeit fortgesetzt werden. Außerdem kann das vorhandene Analysetool von Spielern angewendet werden, die Ihre gespielten Züge noch einmal detaillierter analysieren wollen.

### 2.2 Betriebsbedingungen

Das Programm muss auf verschiedenen Servern lauffähig sein. Die Spieler können beim Start des (Client-) Programms einen Server auswählen (oder auch neue Server hinzufügen).

Das Spiel funktioniert nur, solange sich die Spieler in einen Server einloggen können. Der Server muss also für das Spiel laufen und erreichbar sein (eine Internetverbindung wird also während des gesamten Spiels benötigt).

Sollte während des Spiels die Verbindung unterbrochen werden (durch eine gestörte Internet-Verbindung, einen Absturz eines Endgeräts, o.ä.), kann das Spiel nach einem erneuten einloggen weitergeführt werden, da die Spiele automatisch auf dem Server gespeichert werden.

Um das Spiel starten zu können wird eine Java Laufzeitumgebung (JRE) in der Version 1.8 oder höher benötigt (damit ist das Programm vom Betriebssystem unabhängig)

## 3 Produktumgebung

Das Programm ist durch die Implementierung in Java vom Betriebssystem unabhängig. Für die Ausführung des Programms müssen aber einige Bedingungen erfüllt sein.

### 3.1 Software

- Client
  - Beim Client muss für die Ausführung eine Java Laufzeitumgebung (JRE) in der Version 1.8 oder höher installiert sein
- Server
  - Auf dem Server muss ebenfalls eine Java Laufzeitumgebung in der Version 1.8 oder höher installiert sein
  - Der Server muss über eine MySQL-Datenbank verfügen, auf die das Programm zugreifen kann
  - Auf der MySQL-Datenbank des Servers muss das Programm eine eigene Datenbank haben, auf der die Rechte zum Tabellen erstellen, Daten Einfügen, Updaten und Daten und Tabellen löschen vorhanden sind (Auf anderen Datenbanken müssen keine Rechte vorhanden sein).
  - Zum installieren des Servers und zum erstellen der benötigten Datenbanken und Tabellen muss eine Oberfläche enthalten sein mit der diese angelegt werden können (z.B. phpMyAdmin)

### 3.2 Hardware

- Client
  - Ein internetfähiger Rechner wird benötigt
- Server
  - Ein internetfähiger Server wird benötigt
  - Der Server muss vom Internet aus erreichbar sein (also nicht nur aus einem LAN-Netzwerk)
  - Eine recht geringe Rechenleistung bzw. Festplattenkapazität ist für das Programm ausreichend

### **3.3 Orgware**

- Eine permanente Internetverbindung wird für das Spielen benötigt
- Administratoren müssen auf den Server zugreifen können um die Betriebsparameter zu konfigurieren und das Server-Programm zu starten

## 4 Produktfunktionen

### 4.1 Benutzerfunktionen

#### 4.1.1 Benutzer-Kennung

##### /F0010/

Registrieren: Ein benutzer kann sich mit einem Benutzernamen und einem Passwort auf einem Server registrieren. Die Registrierung ist erforderlich, damit der Benutzer vom System erkannt wird und ein Spiel beginnen kann. (Die Passwörter werden in der Datenbank nicht als Klartext gespeichert)

##### /F0020/

Anmelden: Ein Benutzer der sich registriert hat, kann sich im System anmelden. Dazu benötigt er seine Benutzerkennung und sein Passwort.

##### /F0030/

Abmelden: Der Benutzer kann sich im System abmelden, indem das Anwendungsfenster geschlossen und damit das Programm beendet wird. Dies geschieht automatisch. Meldet ein Benutzer sich nicht ab (z.B. weil die Anwendung nicht ordnungsgemäß beendet wurde (z.B. durch einen Systemabsturz oder einen kill -9 Befehl)) wird das vom System nach kurzer Zeit erkannt und der Benutzer wird vom System abgemeldet (Das selbe geschieht auch wenn die Internetverbindung des Benutzers zu lange gestört ist).

#### 4.1.2 Persönliches Profil

Jeder Benutzer verfügt über ein persönliches Profil, dass nur vom Spieler selbst eingesehen werden kann und in dem z.B. Spielstatistiken gespeichert werden.

##### /F0110/

Anzeige des Profils: Jeder Spieler kann nur sein eigenes persönliches Profil einsehen. Darin kann ein Spieler seine Statistiken einsehen.

##### /F0120/

Anzeige anderer Profile: Ein Spieler kann die Profile anderer Spieler zwar nicht einsehen, aber er kann auf die Spiele zugreifen, die zwischen dem Spieler selbst und einem Beliebigen anderen Spieler ausgetragen wurden. Dabei kann ein Spieler auch seine Statistik (Siege/Niederlagen) gegen jeden anderen Spieler sehen.



## 4.2 Spielfunktionen

### 4.2.1 Initialisierung

#### /F0210/

Eröffnung eines Spiels: Ein angemeldeter Benutzer kann ein Spiel eröffnen, indem er entweder eine der vorhandenen Künstlichen Intelligenzen als Gegner auswählt, oder einen anderen Spieler, der gerade online ist zu einem Spiel einlädt. Bei dieser Einladung kann ausgewählt werden auf welchem Brett (9x9, 13x13 oder 19x19) gespielt werden soll und welcher Spieler welche Farbe erhält (auch eine zufällige Auswahl ist möglich). Außerdem können die Vorgabesteine und das Komi ausgewählt werden.

#### /F0220/

Spiele wieder aufnehmen: Soll ein Spiel weitergeführt werden, das schon begonnen wurde, muss der zweite Spieler, der an diesem Spiel teilgenommen hat online sein. Dieser Spieler erhält dann (wie auch beim Erstellen eines Spiels) eine Einladung, auf der aber vermerkt ist, dass es sich um ein Spiel handelt, das schon begonnen wurde. Handelt es sich bei dem Spiel, das vortgesetzt werden soll um ein Spiel gegen eine KI, wird vom System nachgesehen, ob diese KI vorhanden ist und geladen werden kann. Ist dies der Fall, wird das Spiel fortgesetzt. Ansonsten wird eine Fehlermeldung ausgegeben, die den Benutzer über das Problem informiert.

#### /F0230/

Einladungen annehmen/Ablehnen: Wird ein Benutzer von einem anderen Benutzer zu einem Spiel eingeladen, kann dieser die Einladung entweder annehmen oder ablehnen. Wird die Einladung angenommen, wird für beide Benutzer das Spiel gestartet. Wird die Einladung abgelehnt, wird der Benutzer, der die Einladung verschickt hat, darüber informiert und das Spiel wird nicht gestartet.

### 4.2.2 Spielverlauf

#### /F0310/

Zugmöglichkeiten: Jeder Spieler kann sobald er am Zug ist einen Stein seiner Wahl gemäß den Spielregeln auf das Go-Brett legen. Dazu wählt er mit der Maus die Position des Steins auf dem Brett aus und legt den Stein durch einen Mausklick auf das Brett.

#### /F0320/

Passen: Will ein Spieler Passen, kann er dies über eine Schaltfläche auf der Benutzeroberfläche tun. Um ein ungewolltes Passen zu verhindern, wird der Spieler bevor der Zug ausgeführt wird, noch einmal durch einen Dialog gefragt, ob er wirklich passen möchte.

#### /F0330/

Aufgabe: Will ein Spieler das Spiel aufgeben, kann er das, wie auch das Passen, über

eine Schaltfläche tun. Dabei wird der Spieler wieder in einem Dialog aufgefordert diesen Zug zu bestätigen.

### **/F0340/**

Ende des Spiels: Sobald beide Spieler in ihren aufeinander folgenden Zügen gepasst haben, ist das Spiel beendet. Anders als bei einer Aufgabe müssen die Spieler aber nun noch die Punkte zählen, was in **/F0350/** und **/F0360/** beschrieben ist.

### **/F0350/**

Das Zählen der Punkte: Nachdem das Spiel beendet ist müssen die Spieler ihre Punkte Zählen. Um das zu vereinfachen und Fehler durch Verzählen zu vermeiden müssen die Spieler dafür die Felder markieren, die von jedem Spieler eingeschlossen werden. Jeder der beiden Spieler kann dabei alle Felder des Brettes in der Farbe des Spielers markieren, dem das Gebiet zugeordnet wird. Nicht markierte Felder (wozu auch nicht geschlagene Steine zählen dürfen) werden als neutrale Felder angesehen. Sobald beide Spieler die Felder markiert haben, die ihrer Meinung nach einem Spieler gehören, können sie diese Auswahl abschicken.

### **/F0360/**

Einigung über die Auswahl: Nachdem beide Spieler ihre Auswahl der Felder abgeschickt haben, werden diese vom System miteinander verglichen. Neutrale Felder (wozu auch die Felder gehören, die Steine enthalten, die nicht geschlagen oder gefangen wurden) werden bei diesem Vergleich nicht beachtet. Die Auswahl der anderen Felder wird verglichen und muss bei beiden Spielern übereinstimmen. Ist dies der Fall wird das Spiel vom System ausgezählt, die Spieler werden informiert und das Spiel beendet. Unterscheiden sich die Auswahlen der Spieler werden die Auswahlen an die Spieler zurück geschickt und es wird markiert an welchen Punkten uneinigkeit herrscht. Diese Punkte müssen von beiden Spielern erneut ausgewählt und anschließend abgeschickt werden. Dannach werden sie wieder vom System überprüft. Entsteht keine Einigkeit können sich die Spieler über die Chat-Funktion des Spiels austauschen um zu einer Einigung zu kommen.

## **4.2.3 Speichern und Laden**

### **/F0410/**

Automatisches Speichern: Nach jedem Zug, den einer der Spieler macht wird das Spiel automatisch nach dem der Zug bestätigt wurde in der Datenbank des Servers gespeichert. Dadurch wird der Datenverlust durch Abstürzen des Systems verhindert.

### **/F0420/**

Benutzerdefiniertes Speichern: Das Spiel kann auch vom Benutzer zu jeder Zeit gespeichert werden. Diese gespeicherten Spiele werden aber nur auf dem Filesystem des Benutzers gespeichert (wobei der Benutzer den Pfad des gespeicherten Spiels auswählen kann). Diese Spiele können, da sie nicht auf dem Server vorhanden sind,

nur zur Analyse mit dem Analysetool verwendet werden. Die Spiele werden aber natürlich auch auf dem Server gespeichert, und können daher später fortgesetzt werden.

### **/F0430/**

Laden von Spielen: Jedes auf der Datenbank des Servers gespeicherte Spiel kann wie in **/F0220/** beschrieben wieder geladen und fortgeführt werden, falls es noch nicht beendet ist. Ist das Spiel schon beendet kann es nur mit dem Analysetool geladen werden, dass im Folgenden Abschnitt beschrieben wird.

## **4.3 Analysetool**

### **/F0510/**

Start des Analysetools: Ist ein Spiel auf der Datenbank des Servers oder im Filesystem des Benutzers gespeichert, kann dieses Spiel ausgewählt werden um es mit dem Analysetool näher zu analysieren. Das ist auch mit Spielen möglich, die noch nicht beendet wurden.

### **/F0530/**

Speichern und Laden der Analyse: Eine Spielanalyse kann zu jeder Zeit und mit allen analysierten Spielpfaden vom Benutzer auf dem lokalen Filesystem gespeichert und später wieder (aus dem Menü heraus) geladen werden.

### **/F0540/**

Durchgehen des Spiels: Mit dem Analysetool ist es möglich das gesamte Spiel Zug für Zug durchzugehen und nachzuvollziehen. Dabei kann durch Schaltflächen für Vor und Zurück durch die Züge des Spiels navigiert werden.

### **/F0550/**

Ausprobieren anderer Züge: Es ist mit dem Analysetool bei jedem Zug möglich, statt dem im Spiel gesetzten Stein einen anderen Zug zu wählen. Dazu muss nur der Stein auf dem Brett platziert werden, wie es auch im Spiel möglich ist (siehe **/F0310/**). Wird ein Zug des Spiel verändert, wird dadurch ein neuer Pfad erstellt, der nach dem letzten Zug des Spiels abzweigt wird. Es ist für den Benutzer dann möglich diesen Pfad weiter zu Analysieren, oder auch zum eigentlichen Spielpfad zurückzukehren. Der Pfad der erstellt wurde wird dabei gespeichert und bleibt erhalten. Es ist auch möglich von einem solchen Pfad wieder einen neuen Pfad abzuzweigen.

### **/F0560/**

Auswahl des Pfades und des Zugs: Der Pfad in dem sich die Analyse zur Zeit befindet, sowie der Zug und die anderen Pfade und Züge werden vom Analysetool dargestellt. Der Benutzer kann durch auswählen eines Pfades oder Zuges zu einem bestimmten Punkt der Analyse springen.

### **/F0570/**

Löschen von Zügen und Pfaden: Jeder Zug, der mit dem Analysetool gemacht wurde kann auch wieder gelöscht werden. Alle Pfade die auf diesen Zug folgen werden damit ebenfalls gelöscht. Das Löschen der original Spielzüge ist nicht möglich.

### **/F0580/**

Auszählen der alternativen Pfade: Wenn ein alternativer Pfad beendet wird kann das Spiel mit diesem alternativen Ende ausgezählt werden, um zu sehen ob sich das Ergebniss verändert hat. Dazu muss der Benutzer wie in **/F0350/** beschrieben, die Felder auswählen, die zum Bereich eines Spielers gehören. Da es nur eine Analyse ist, ist keine Einigung zweier Spieler für das auszählen nötig.

## **4.4 Statistiken**

### **/F0610/**

Persönliche Statistik: Jeder Spieler kann seine persönliche Statistik ansehen. Diese beinhaltet Informationen über Siege, Niederlagen, eingestellte Komi, Vorgabesteine, Spielfarben und Punkte, bzw. Aufgaben. Die persönlichen Statistiken anderer Spieler kann ein Benutzer sich nicht anzeigen lassen.

### **/F0620/**

Selektierte Statistik: Die Statistiken eines Benutzers können auch selektiert angezeigt werden. Z.B. kann nach Spielen gegen einen Bestimmten Spieler oder eine bestimmte KI selektiert werden. Es ist wieder nur möglich die Statistiken einzusehen die den Benutzer selbst betreffen.

## 5 Produktdaten

### /D0010/

Benutzerdaten: Von jedem Benutzer werden in der Datenbank des Servers, an dem er sich registriert hat, folgende Daten gespeichert.

- Benutzer-ID (eindeutig)
- Benutzerkennung
  - Benutzername (eindeutig)
  - Passwort (verschlüsselt)
- Gespielte Spiele des Benutzers

### /D0020/

Spiele: Begonnene und auch beendete Spiele werden automatisch in der Datenbank des Servers gespeichert und können auch im Filesystem des Benutzers gespeichert werden. Die gespeicherten Daten der Spiele sind:

- Teilnehmende Spieler
- Farben der Spieler
- Vorgabesteine
- Komi
- Liste aller Züge
- Sieger
- Punktzahl

### /D0030/

Analysen: Spielanalysen werden von Benutzern erstellt und werden nur in den Filesystemen der Benutzer gespeichert. Die gespeicherten Daten sind:

- Das Spiel, das analysiert wird (enthält alle Daten die in /D0020/ aufgelistet werden)
- Die vom Benutzer analysierten, alternativen Pfade des Spiels
- Ausgezählte Spielstände, die der Benutzer zu einem bestimmten Pfad oder Zug erstellt hat.

## 6 Produktleistungen

### /L0010/

Schiedsrichter: Das Spiel verfügt über einen Schiedsrichter, der entscheidet ob ein Zug der von einem Spieler gemacht wurde den Regeln entspricht.

### /L0020/

Fehlerhafte Eingaben: Macht der Benutzer eine Eingabe, die Fehlerhaft ist, wird er vom Programm durch einen Dialog drauf hingewiesen. In einem solchen Fall wird ein Zug nicht ausgeführt und der Spieler hat erneut die möglichkeit zu Ziehen.

## **7 Benutzeroberfläche**

## 8 Qualitätszielbestimmungen

	sehr wichtig	wichtig	weniger wichtig	unwichtig
Korrektheit	<b>X</b>			
Benutzerfreundlichkeit	<b>X</b>			
Robustheit		<b>X</b>		
Zuverlässigkeit		<b>X</b>		
Effizienz			<b>X</b>	
Portierbarkeit			<b>X</b>	
Kompatibilität			<b>X</b>	



## **9 Globale Testszenarien und Testfälle**

# 10 Entwicklungsumgebung

Alle verwendeten Entwicklungstools sind als Freeware bzw. Open-Source vorhanden.

## 10.1 Software

- Eclipse
- JDK 1.8
- JUnit
- Ant
- Git
- PhpMyAdmin
- Latex

## 10.2 Hardware

- Internetfähiger Rechner
- Raspberry Pi (Server)
- Fritz!Box

## **11 Ergänzungen**