Projektziele und Vorgehen

# 1 Rahmenbedingungen

## 1.1 Ziel des Projektes

Ziel des Projektes ist die Implementierung eines Programms, das in der Lage ist, gegen ein anderes Programm “4 gewinnt” zu spielen. Dabei wird die Kommunikation der Programme über einen zentralen Server realisiert.

## 1.2 Äußere Bedingungen und deren Realisierung

Die verwendete Programmiersprache ist **Java** (J2SE). Entsprechend wird objektorientiert gearbeitet. Der Import der Programmbestandteile in eclipse auf anderen PCs muss möglich sein.

Eine entsprechende Benutzeroberfläche (im folgenden **GUI**) wird darauf aufbauend bereit gestellt. Folgendes wird angezeigt:

* Satzstatus
* Spielstand
* Spielfeld
* Spielzüge des Spiels
* Anwendung: 3 Abfragen über vergangene Spiele

*Realisierung:* Die Realisierung des GUI erfolgt über Java Fx.

Ebenso besteht eine **Datenbankanbindung**, über die auch vorherige Spielabläufe und -stände gespeichert werden können. gespeichert werden die folgenden Daten:

* Gegner
* Startspieler
* Sieger
* Punkte
* Spiele
* Sätze
* Züge

Realisierung der Datenbank soll über HSQLDB erfolgen. Ebenso sollen für eine Anwendung Befehle und Beispieldaten bestehen, die 3 Abfragenüber vergangene Spiele realisieren können.

Der **Spielablauf** wird geregelt durch die Kommunikation über den Server. Ab der Freigabe des Spielzugs hat der Agent i.d.R. **2 Sekunden** Zeit, seinen Zug zu platzieren. Im Zweifel kann die Zeit während des Spiels auf 1 oder 3 Sekunden variiert werden. Punktevergabe erfolgt im Tunier nach dem **Punktemodus**.

Im Rahmen der Entwicklung entstehen ebenfalls eine Entwicklungsdokumentation und eine Benutzerdokumentation. Rahmenbedingungen dafür:

* Es wird im Quellcode dokumentiert, Modelle graphisch dargestellt, sowie eine text-gebundene Dokumentation verfasst
* Jeder ist dafür verantwortlich, den von ihm erstellten Part der Software ausreichend zu dokumentieren. Zusätzlich gibt es eine Kontrolle der Dokumentationstätigkeiten im Rahmen der Projektorganisation.
* Dokumentationssprache ist deutsch.

# 2 Vorgehensweise

## 2.1 Entwicklungsmodell

Bei der Entwicklung wird nach dem Spiral-Modell vorgegangen. Das Modell sieht vor, dass für jeden Entwicklungsschritt ein Zyklus von 4 Phasen durchlaufen wird. Die Phasen gleidern sich folgernmaßen:[[1]](#footnote-1)

1. Ziele: Ziele Festlegen, Rahmen definieren

2. Vorgehen: Alternativen bewerten, Risiken bewerten

3. Produkt: Realisierung des Zwischenprodukts

4. Test: Beurteilung der Phasen 1-3, Planung des nächsten Zyklus

Zyklen in diesem Projekt können sein:

- Entwicklungszyklus Prototyp: Ziel ist ein vorzeigetauglicher Prototyp

- Entwicklungszyklus Beta: Ziel ist die auslieferungsfähige Beta-Software

- Entwicklungszyklus Abschluss: Ziel ist die Fertigstellung der Software, wobei nach Projektplan in diesem Zyklus hauptsächlich getestet werden soll

## 2.2 Verwaltung der Dokumente

Das Projekt wird mit Github als Repository durchgeführt. Eine Einführung für das Team wird dazu von einem Teammitglied vorbereitet. Die URL zu dem Projekt ist:

https://github.com/Frameset91/untitled0815

# 3 Dokumentation der einzelnen Zyklen

## 3.1 Erster Zyklus: Prototyp

1. Ziele (10.09.2012):

Projektorganisation aufstellen und Modell vereinbaren, auf deren Grundlage die einzelnen Kleingruppen weiter arbeiten können; Arbeitsteilung vornehmen; erfolgreichen Prototypen erstellen

2.

2.1Vorgehensweise in der erste Vorlesung:

* Gliederung in Arbeitspakete:
  + Modellierung
    - von allen Teammitgliedern mitentwickelt/Zustimmung erhalten
    - befasst sich mit Themenübergreifender Architektur
    - Grundlage für alle weiteren Modelle und Implementierungen
    - muss bei veränderten Modellen stets auf Korrektheit überprüft werden
    - Sicherstellung der zeitgerechten Beantwortung der Zug-Aufforderungen
  + GUI
    - Technologie: Java Fx
    - die oben genannten Bedingungen (s. 1.2) müssen erfüllt werden
    - Modell und Implementierung
  + Datenmodell
    - ER-Modell erstellen
    - implementieren
  + Logik (z.B. 4-gewinnt-Algorithmus)
    - Modelle und Implementierung
    - Algorithmus für 4 gewinnt entwickeln
    - Logik für weitere Anforderungen entwickeln (s. 1.2)
  + Kommunikation (Client-server, XML)
    - Anwendung der unter der Projektbescheibung (<http://nadviser.de/Pg/Auftrag/Auftrag4gNet510.pdf>) genannten Protokolle
    - Sicherstellung der Kommunikation
  + Projektorganisation/Administration und Dokumentationssicherstellung
    - Sicherstellung der ordnungsgemäßen Dokumentation (z.B. Ablage zur Verfügung stellen)
    - Projektablaufplan erstellen
* zunächst werden Modelle überblicksmäßig von Björn, Sascha, Nora und Johannes bearbeitet
* Projektmanagement und zentrale Ablage werden von Alex und Henny vorbereitet

2.2 Arbeitsaufteilung für folgende Vorlesungen:

* GUI → Nora
* Datenbank und DBConnect → Henny
* Logik → Johannes
* Threads und Events in Java → Sascha
* Spielablauf → Sascha
* Log → Alex
* Kommunikation mit Server → Björn
* Projektorganisation → Henny

3. Produkt: siehe Dokumentation zu Modellen und Projektablaufplan

* Der Ablauf der Implementierung ist gemäß Aufteilung (s.o.), wobei Info-Sessions eingeschoben werden und der Abgleich regelmäßig erfolgt.

4. Test und Rückblick auf den Zyklus:

* Gefundene Modellierungsprobleme:
  + Im Bezug auf Eventhandling
  + Klassendiagramm bedarf einer Überarbeitung
  + GUI muss auf den neusten besprochenen Stand gebracht werden
  + Kurze Session zu Threads war nötig
* Analyse des Codes durch den Dozenten:
  + hochst. 100 Zeilen pro Methode --> zur Zeit bis 284
  + number of static attr per type: max 10 --> zur Zeit darüber (Erklärungsansatz: benötigt für Tests)
  + sämtliche Dokumente sollen als PDF oder im JPEG-Format vorliegen

## 3.2 Zweiter Zyklus: Beta-Version

1. Ziele

* Entwicklung der Beta-Version, die bereits 5 Spielzüge spielen kann und dabei mit GUI und Datenbank verknüpft ist
* Nebenbei Dokumentation und notwendige Dokumente anlegen und möglichst weit fertigstellen

2. Vorgehen

* Wie bisher hat jeder seinen Aufgabenbereich, da sich diese Aufteilung bewährt hat
* Für Absprachen werden die Vorlesungsstunden genutzt

3. Produkt

* Siehe Quellcode und Dokumentation

4. Test

## 3.3 Dritter Zyklus: Projektabschluss

1. Ziele

2. Vorgehen

3. Produkt

4. Test

## 3.4 Fazit zum Entwicklungsmodell

- Zyklen kum unterscheidbar

- gut aber: Dokumentation von Problemen zu bestimmten Zeitständen 🡪 Progress ablesbar

1. Quelle zum Nachlesen:<http://www.infforum.de/themen/anwendungsentwicklung/thema_SE_spiral-modell.htm> [↑](#footnote-ref-1)