**Das 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 gliedern sich folgendermaßen:[[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 sind:

- Entwicklungszyklus Prototyp: Ziel ist ein vorzeigetauglicher Prototyp

- Entwicklungszyklus Beta-Release: Ziel ist die auslieferungsfähige Beta-Software

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

Dokumentation der einzelnen Zyklen:

1. Prototyp

1. Ziele: erfolgreichen Prototypen erstellen; dazu: gemeinsame Überlegungen zur Architektur und daraufhin Arbeit in Kleingruppen an vereinbarten Arbeitspaketen; Erstellung eines Zeitplans

2. Vorgehen: Einigung auf Zuständigkeiten:

* GUI → Nora
* Datenbank → Henny
* Logik/KI → Johannes
* Spielablauf → Sascha
* Log → Alex
* Kommunikation mit Server → Björn
* Projektorganisation und Dokumentation → Henny und Alex
* Modellierung: alle

3. Produkt: Erste Entwürfe zu Klassendiagramm und ERD sind entstanden. Info-Sessions über wichtige Themen werden für alle Teammitglieder vorbereitet, Themen sind: Github-Benutzung, Threads, JavaDoc. Programmierung erfolg gemäß den Zuständigkeiten. Testen des Prototypen erfolgt manuell.

4. Test und Rückblick auf den Zyklus:

Gefundene Modellierungsschwierigkeiten: Eventhandling, Klassendiagramm muss überarbeitet werden, GUI muss angepasst werden

Feedback: sämtliche Dokumente müssen ab sofort als PDF oder JPEG vorliegen; ansonsten soweit in Ordnung

2. Beta-Release

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. Das Programm soll möglichst weit gebracht sein, so dass im letzten Zyklus das Testen im Fokus liegt.

2. Vorgehen: Wie bisher hat jeder seinen Aufgabenbereich, da sich diese Aufteilung bewährt hat. Für Absprachen werden die Vorlesungsstunden genutzt.

3. Produkt: (vgl. Quellcode) Testklassen und –fälle z.T. werden bereits implementiert und entworfen.

4. Test und Rückblick auf den Zyklus:

Beim Spielen auf dem Server treten z.T. einige andere Bedingungen auf als beim Testen auf den eigenen Rechnern; Beim Verpacken in eine ausführbare Jar-Datei sind in Bezug auf die Datenbank einige Punkte zu beachten; Die KI muss noh weiterentwickelt werden; die Stabilität des Programms muss ausgebaut werden.

3. Release:

1. Ziele: Ziel ist die Abgabe des voll funktionsfähigen Programms, sowie der Dokumentation.

2. Vorgehen: Weiterhin gibt es die Verantwortlichen für die einzelnen Komponenten. Ein Verantwortlicher für die Dokumentation wurde bestimmt. Außerdem wird ein Testkonzept entwickelt. Auffälligkeiten werden in einer To-Do-Liste mit Zuständigkeit, Erfassungsdatum, Erfasser und Status hinterlegt.

3. Produkt: Als Produkte können der Agent, die Dokumentation, ebenso wie die Tests angesehen werden.

4. Test und Rückblick auf den Zyklus:

Trotz vorher guter Einhaltung des Zeitplans sind einige unerwartete Überraschungen, insb. Beim Testen mit dem Server, aufgefallen. Da aber ausreichend Zeit eingeplant war für Änderungen, ist es machbar. Automatisierte Tests wären an dieser Stelle in einem größeren Projekt sicher sinnvoll. Da aber die Vorlesungszeit zur Verfügung steht und ausreichend Teammitglieder vorhanden sind, ist die Aufgabe des Testens auch so gut zu bewältigen.

Fazit zum Entwicklungsmodell:

Es war für alle Teammitglieder die erste Erfahrung mit dem Spiralmodell. Positiv ist zu bemerken, dass das Modell recht einfach zu verstehen und umzusetzen ist. Allerdings war es in diesem kleinen Rahmen schwierig, die einzelnen Zyklen voneinander abzugrenzen. Da kaum Vorgaben gemacht werden, wird weiteres Projektmanagement, z.B. ein Zeitplan, benötigt, so dass es beinahe wie nebenläufige Strategien scheint und die Abstimmung beider Mittel zeitaufwendig ist. In einem größeren Projekt wäre dieses Entwicklungsmodell sicher eher zu gebrauchen.

1. Quellen zum Nachlesen z.B.: Skript Wiedemann oder<http://www.infforum.de/themen/anwendungsentwicklung/thema_SE_spiral-modell.htm> [↑](#footnote-ref-1)