

# Software Engineering 1

## Abgabedokument

### Teilaufgabe 1

#### (Anforderungsanalyse und Planungsphase)

<b>Nachname, Vorname:</b>	Lymarenko Oleksandr
<b>Matrikelnummer:</b>	12028261
<b>E-Mail-Adresse:</b>	a12028261@univie.ac.at
<b>Datum:</b>	17.03.2024

## Aufgabe 1: Anforderungsanalyse

Analysieren der Spielidee und des Netzwerkprotokolls um 8 Anforderungen (bestehend zumindest aus 3 funktionalen, 3 nichtfunktionalen und einer Designbedingung) nach den folgenden Kriterien zu dokumentieren. Achten Sie darauf den im Skriptum und der Vorlesung behandelten Qualitätsaspekten Genüge zu tun.

### Typ der Anforderung: funktional

#### Anforderung 1

- **Beschreibung:** Betreten der Wasserfelder: Die KI darf kein Wasserfeld betreten.
- **Bezugsquelle:** Spielidee, *„Außerdem darf Wasser unter keinen Umständen betreten werden (Spielfiguren können nicht schwimmen).“*

#### Anforderung 2

- **Beschreibung:** Spiel anfordern: Ein Spieler kann das Spiel anfordern.
- **Bezugsquelle:** Spielidee, *„Initial gilt es am Server ein neues Spiel anzufordern. Dieser erste Schritt wird noch von einem Menschen durchgeführt...“*

#### Anforderung 3

- **Beschreibung:** Sieg im Normalfall: Eine KI kann gewinnen, indem sie mit einem Schatz die gegnerische Burg betritt.
- **Bezugsquelle:** Spielidee, *„Sobald diese gefunden wurde, muss die KI sich zu dieser bewegen, die Bewacher der Burg mit dem gerade gefundenen Schatz bestechen und so die Burg "übernehmen". Hat die KI dies geschafft, gewinnt diese und damit auch der menschliche Spieler, der diese KI erstellt hat.“*

### Typ der Anforderung: nicht funktional

#### Anforderung 4

- **Beschreibung:** Dauer der Spielrunde: Eine Spielrunde muss maximal 5 Sekunden dauern.
- **Bezugsquelle:** Spielidee, *„Für jede dieser rundenbasierten Spielaktion hat die KI maximal 5 Sekunden Bedenkzeit.“*

#### Anforderung 5

- **Beschreibung:** Dauer des Spiels: Ein Spiel muss nicht länger als 320 Runden dauern.
- **Bezugsquelle:** Spielidee, *„Um die Spiele für die Zuschauer spannend zu gestalten, wurde festgelegt, dass ein Spiel insgesamt nicht länger als 320 Spielaktionen (und damit 320 Runden) dauern darf.“*

#### Anforderung 6

- **Beschreibung:** Kartengeneration: Die Kartenhälften müssen zufällig generiert werden.
- **Bezugsquelle:** Spielidee, *„Kartenhälften müssen zufällig mit Algorithmen generiert und nicht statisch vorgegeben werden.“*

## Typ der Anforderung: Designbedingung

### Anforderung 7

- **Beschreibung:** Format für den Nachrichten- und Datenaustausch: Für den Nachrichten- und Datenaustausch muss XML verwendet werden.
- **Bezugsquelle:** Netzwerkprotokoll, *„Die ausgetauschten Daten bzw. Nachrichten werden im XML-Format definiert bzw. erwartet.“*

### Anforderung 8

- **Beschreibung:** CLI-Visualisierung: Das Spiel muss mittels command-line interface visualisiert werden.
- **Bezugsquelle:** Spielidee, *„Während des Spiels müssen die Karte und deren bekannten Eigenschaften und wichtige Spielzustände von den Clients mittels command-line interface (CLI) für Anwender nachvollziehbar visualisiert werden.“*

### Anforderung 9

- **Beschreibung:** Die Architektur des Spiels: Die Architektur des Spieles ist Client/Server Architektur.
- **Bezugsquelle:** Spielidee, *„Die Grobarchitektur ist damit als klassische Client/Server Architektur vorgegeben.“*

## Aufgabe 2: Anforderungsdokumentation

Dokumentation einer zum relevanten Bereich passenden Anforderung nach dem vorgegebenen Schema. Ziehen Sie eine Anforderung heran, für die alle Bestandteile der Vorlage mit relevantem Inhalt befüllt werden können. Wir empfehlen hierzu eine **funktionale** Anforderung auszuwählen.

### Dokumentation Anforderung

- **Name:** Richtung der Bewegung
- **Beschreibung und Priorität:** Eine KI kann sich nur horizontal und vertikal zu direkten Nachbardsfeldern bewegen. Priorität: Hoch
- **Relevante Anforderungen:**
  - Sieg im Normalfall: Falls die Burg auf einem diagonal benachbarten Feld liegt, kann diese nicht betreten werden.
  - Betreten der Wasserfelder: Diese Anforderung stellt eine Ausnahme dar. Auch wenn ein Wasserfeld horizontal oder vertikal benachbart liegt, kann es trotzdem nicht besucht werden.
- **Relevante Business Rules:**
  - Die Karte soll den Spielregeln entsprechen. Es soll beispielsweise keine nicht erreichbaren Felder geben. Das würde die Bewegungen direkt beeinträchtigen.
  - Der Server muss korrekt implementiert werden, um die Aktionen (inklusive Bewegungen) der KI kontrollieren zu können und bei Bedarf bestrafen.
- **Impuls/Ergebnis - Typisches Szenario:**
  - Vorbedingungen:**
    - Das Spiel ist bereits gestartet.
    - Die beiden Spieler sind bereits registriert.
    - Die Karte wurde aus zwei Kartenhälften erfolgreich erstellt.
    - Die Spielfigur befindet sich auf einer Wiese oder auf einem Berg.
    - Es gibt mindestens ein betretbares Feld in direkter, horizontaler oder vertikaler Nachbarschaft von Spielfigur.
  - Hauptsächlicher Ablauf:**
    - Impuls: Der Client ist an der Reihe.
    - Ergebnis: Die zugehörige KI entscheidet (aufgrund von Suchalgorithmen) die Spielfigur vertikal oder horizontal zu bewegen.
    - Impuls: Der Client überträgt eine Bewegung an Server.
    - Ergebnis: Der Server bestätigt die Bewegung.
  - Nachbedingungen:**
    - Die Spielfigur hat sich in die gewünschte Richtung bewegt.
- **Impuls/Ergebnis - Alternativszenario**
  - Vorbedingungen:**
    - Das Spiel ist bereits gestartet.
    - Die Karte wurde aus zwei Kartenhälften erfolgreich erstellt.
    - Die Spielfigur befindet sich auf einer Wiese oder auf einem Berg.
    - Die Spielfigur hat einen Schatz.
    - Vertikal oder horizontal auf einem benachbarten Feld befindet sich eine Wiese mit der Burg
  - Hauptsächlicher Ablauf:**
    - Impuls: Der Client/die KI ist an der Reihe.

Ergebnis: Die KI entscheidet (aufgrund von Suchalgorithmen) die Spielfigur vertikal oder horizontal zu bewegen.

Impuls: Der Client überträgt eine Bewegung an Server.

Ergebnis: Der Server bestätigt die Bewegung

Impuls: Die Anzahl der Bewegungsaktionen, die erforderlich sind, um das Feld mit der Burg zu betreten, wurde erreicht.

Ergebnis: Die Spielfigur befindet sich nun auf dem Feld mit der Burg.

Impuls: Der Server erkennt, dass die Spielfigur die Burg übernimmt.

Ergebnis. Der zugehörige Client gewinnt das Spiel.

**Nachbedingungen:** Der Client hat das Spiel gewonnen, das Spiel ist zu Ende.

• **Impuls/Ergebnis - Fehlerfall:**

**Vorbedingungen:**

- Das Spiel ist bereits gestartet.
- Die Karte wurde aus zwei Kartenhälften erfolgreich erstellt.
- Die Spielfigur befindet sich auf einer Wiese oder auf einem Berg.
- Vertikal oder horizontal auf einem benachbarten Feld befindet sich ein Wasserfeld.

**Hauptsächlicher Ablauf:**

Impuls: Der Client/die KI ist an der Reihe.

Ergebnis: Die KI entscheidet, die Spielfigur Richtung Wasserfeld zu bewegen.

Impuls: Der Client überträgt die Bewegung an Server.

Ergebnis: Der Server lehnt die Bewegung ab und bestraft die KI, indem sie verliert.

**Nachbedingungen:**

Der zugehörige Client hat das Spiel verloren.

• **Benutzergeschichten:**

- Als KI möchte ich meine Spielfigur gemäß den Regeln bewegen, um meine Chancen für den Sieg zu maximieren.
- Als Server werde ich kein Überspringen der Felder erlauben, um ein faires Spiel zu gewährleisten. Ich werde nur diejenigen Bewegungen akzeptieren, die sich horizontal oder vertikal richten.

• **Benutzerschnittstelle:**

BWWW

W~W Spielfigur(S) bewegt sich nach rechts.

SWWW

→

BWWW

W~W

WSWW

• **Externe Schnittstellen:**

Schnittstelle Client/Server: Es wird eine Verbindung zwischen dem Client und dem Server mittels HTTP-Protocol aufgebaut, die es ermöglicht, die Bewegungen und andere Aktionen durchzuführen.

Schnittstelle XML: Die Nachrichten und Anfragen, die die Bewegungsaktionen beschreiben, werden in XML definiert.

### **Aufgabe 3: Architektur entwerfen, modellieren und validieren**

Modellieren Sie händisch alle notwendigen Packages, Klassen und deren Methoden (samt Beziehungen) als zwei UML Klassendiagramme. Achten Sie darauf, dass die Modelle sinnvoll benannte Packages, **Klassen**, **Methoden** (inkl. Parameter und Rückgaben) und **Felder** beinhalten und die Vorgaben der Spielidee bzw. des Netzwerkprotokolls vollständig in sinnvoller Granularität abgedeckt werden.

**Basierend auf dem Klassendiagramm:** Erstellen Sie zwei Sequenzdiagramme zu den beiden in der Übungsangabe vorgegebenen Aspekten. Alle erstellten Diagramme sollten semantisch und syntaktisch korrekt sowie untereinander konsistent sein.



