

# Meilenstein 1: Treffen 03.12.2019 Protokoll

## Aufgabe 1: Kontextanalyse

- Einleitung
- Motivation
- Vision
- Projektkontext
  - sollte im Fließtext vorliegen
  - Todo: Wer übernimmt diese Aufgabe?
- Marco

## Aufgabe 2: Definition von Fachbegriffen

Möglichst heute, als Basis für Rest des Dokuments

Todo: festlegen der Begriffe, wer ist für das ausformulieren verantwortlich?

- benutzer
- spieler
- lobby
- spielpartien
- Benutzer-client
- server
- ki client
- NPC
- editor
- Zuschauer
- level config (überbegriff zu szenario,charbeschreibung,etc)
- charakter
- Szenario
- Partieconfig
- char-Beschreibung
- item
- gadget
- aktion
- (alles mögliche an spielfelder namen)
- Bewegungsphase
- aktionsphase
- Intelligence Points
- Sichtlinie
- Geheimnis
- Beim FA schreiben hinzufügen von weiteren Fachbegriffen, welche benötigt werden
- Dominik A

### Aufgabe 3: Domänenmodell

Todo: Inhalt, wer ist für das Fertigstellen verantwortlich?

- Jonnas

### Aufgabe 4: Anforderungsdefinition

#### Akteure

Todo: Welche Akteure, wer ist für das Ausformulieren verantwortlich?

- Caro

#### Funktionale Anforderungen

Todo: festlegen der Titel, wer formuliert was aus?

Naming: FA-C15, FA-SXX, FA-KIXX, FA\_-EXX

FA-GXX für general FA

- server, config files für server
- server lädt szenario und beschreibung der charaktere
- server zwei clients über netzwerk
- Server zuschauer, beliebige Anzahl
- server, timeouts für clients
- server, verspätete nachrichten werden verworfen
- server, client kann pause anfordern, KIs nicht
- server, persistente sessions
- server, regelverstöße
- server, leitet aktualisierung des Spielzustandes an alle clients weiter
- server, gewinnüberprüfung
- server, Standardkonformes logging -> replay (optional)
- client, intro
- client, registriert sich als Mitspieler beim server
- client, kann sich als zuschauer beim server registrieren
- client, teilt server mit dass von mensch gesteuert
- client, nutzername
- client, GUI
- client, interaktion mit GUI
- client, hilfefunktion (optional)
- client, hotkeys (optional)
- client, rundenphasen werden animationen dargestellt
- client, wunsch auf pausieren
- client, persistente verbindung TCP und websocket
- client, anzeigen des winners am ende

- client, statistik (optional)
- client, replay (optional)
- ki, ki teilt server mit, dass ki
- ki, dürfen nicht pausieren, spieler darf dann beliebig lange
- ki, ki spielt sinnvoll
- ki, es können verschieden stufen eingestellt werden (optional)
- ki, schnittstelle für client (optional)
- editor, json format für alle dateien, speicherung der Ressourcen
- editor, szenarios und char beschreibungen für GUI
- editor, zufällige szenarien (optional)
- editor, validation von stuff
- gen, 2.1 in FA übersetzen (dominik T)
- gen, 2.2 in FA übersetzen, Appendix A (caro)
- gen, 2.3 in FA übersetzen (caro)
- gen, 2.4 in FA übersetzen (dominik T)
- gen, 2.5 in FA übersetzen (relativ kurz!) (lukas)
- gen, 2.6 in FA übersetzen (lukas)
- gen, 2.7 in FA übersetzen (relativ kurz) (lukas)
- gen, 2.8 partie vorbereitung in FA (marco)
- gen, spielbeginn 2.9 (gehört evtl zu server) (otto)
- gen, 2.10 in FA spielablauf (dominik a)
- gen, 2.11 Siegbedingung in FA (dominik a)
- gen, 2.12 allgemeine bemerkungen (dominik a)

### **zuteilung**

- FA zu Komponenten (Kap3) Otto

### **Nicht-funktionale Anforderungen**

Todo: festlegen der Titel, wer formuliert was aus?

- server cli docker
- ki, cli docker
- client, responsive (frame rate 144fps und input delay 50ms)
- server, antwortzeit 100ms
- alle, bei absturz logs ablegen
- alle, testcoverage 50% (unit und monkey-testing)
- alle, Codeanalyse:Verwendung von Sonarqube zur Qualitätsicherung
- alle, continous integration für code und merge requests
- alle, ease of use (erlernbar innerhalb x-facher benutzung)
- alle, UX -> sinnvolles maß (alle teammitglieder zufrieden)
- alle, Doxygen dokumentation
- verantwortlich Marco