

# Protokoll 27.04.2025

Erstellt von: Carolin Scheffler

## Inhaltsverzeichnis

<b>1</b>	<b>Besprochenes</b>	<b>1</b>
1.1	Libraries . . . . .	1
1.2	HTTP . . . . .	2
1.3	neue Anforderungen . . . . .	2
1.4	Build . . . . .	2
1.5	JavaVersion auf dem Server . . . . .	2
1.6	Testmethoden . . . . .	3
1.7	Aussehen Spiel und weitere Spielmechanik . . . . .	3
1.8	Feature . . . . .	3
<b>2</b>	<b>Übergeordnetes Ziel</b>	<b>4</b>
<b>3</b>	<b>Nächstes Ziel: Sonntag</b>	<b>4</b>
3.1	Koordinator . . . . .	4
3.2	Architektur . . . . .	4
3.3	GitHub . . . . .	4
3.4	Doku+Tests . . . . .	4
<b>4</b>	<b>Nächstes Ziel: Mittwoch</b>	<b>5</b>
4.1	Vorbereitung . . . . .	5
4.2	Am Mittwoch zu erledigen: . . . . .	5
<b>5</b>	<b>Zeitplan</b>	<b>5</b>
<b>6</b>	<b>Igler Fragen:</b>	<b>5</b>

## 1 Besprochenes

### 1.1 Libraries

- libGDX-Overkill, eher für Echtzeitspiele
- JavaFX- nur 2D

- LWJGL- no

**Entscheidung:** FXGL-klingt gut, basiert auf JavaFX, JavaFX nutzt OpenGL. Man kann isometrie umsetzen? Müssen wir selbst umsetzen.

## 1.2 HTTP

- HTTP-Request verfolgt Request-Response: Problem mit Verzögerungen falls wir ständige updates nutzen, und kein Broadcast
- da nur 2 Spieler: Clients können periodisch Request-update senden, (Problem: mehrere Request könnten zu Verzögerungen führen, hier aber durch 2 Spieler irrelevant)

## 1.3 neue Anforderungen

mehrere Spiele gleichzeitig?

**Entscheidung:** Frage an Iglar stellen. Punkt 6



2.PNG

## 1.4 Build

**Entscheidung:** Als Build Tool wird gradle genutzt.

## 1.5 JavaVersion auf dem Server

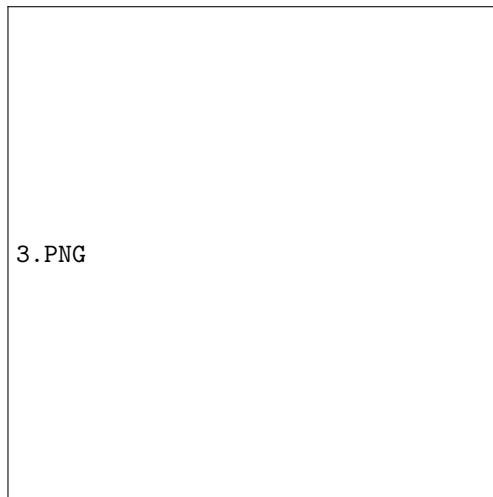
**Entscheidung:** Java 17 auf dem Server schon erledigt.

## 1.6 Testmethoden

JUnit und CI/CD

## 1.7 Aussehen Spiel und weitere Spielmechanik

- Spielbrettgröße: 16x16
- Felder: werden geflagt
- Aufbau erstes Spielbrett wird festgelegt sein
- Aktionspoints einführen:  
Welche Aktionen gibt es? → Bewegen, Schießen
- Aktionen Bewegung: Bewegungsrate: 1 Feld( mit Powerups 2 Felder) Man kann nicht auf das Feld zurück, wenn im letzten Zug dort gewesen!
- 2 Spielzugvarianten:



## 1.8 Feature

- Zufallsmap erstellen können
- mehrere Fahrzeuge möglich:  
Panzer: schießen: 3 Felder, Bewegungsrate: 2 Felder  
Artillerie schießen: 5 Felder, Bewegungsrate: 1 Feld
- Irgendwann 4 Spieler hin fürs Balancing
- Powerups

## 2 Übergeordnetes Ziel

2 Panzer auf dem Brett bewegen, dann Features!

## 3 Nächstes Ziel: Sonntag

- ServerProjekt: Springboot drauf und lokal ausprobieren
- ClientProjekt: 2D Array erstellen als Koordinaten
- Objekt für erste Bewegungen (Objekt hat move-funktion)
- Spielmap entwerfen
- Klassenstruktur überlegen

### 3.1 Koordinator

- Einmal OrdnerKonzept für Gitlab entwerfen
- SpritesKonzept anschauen
- Serverstruktur pushen

### 3.2 Architektur

- HTTP-Broadcast Alternativen
- Springboot genauer einlesen falls Zeit, kompatibilität gewährleistet? Welche Grenzen Springboot hat?

### 3.3 GitHub

- Wenn alle Springboot haben und Zeit ist, Server jederzeit verfügbar (keine Prio)
- Branchmanagement
- Gitlab als Planungstool/Projektmanagementtool? Welche Möglichkeiten haben wir?

### 3.4 Doku+Tests

- CI/CD
- JUnits vorbereiten, Testfälle überlegen?

## **4 Nächstes Ziel: Mittwoch**

### **4.1 Vorbereitung**

- Git Verwaltung über IntelliJ
- FXGL Tutorial anschauen
- Springboot Tutorial anschauen

### **4.2 Am Mittwoch zu erledigen:**

- ClientProjekt
- Spielmap entwerfen
- Klassenstruktur überlegen

## **5 Zeitplan**

12 Wochen

Jede Woche Aufgabenformulierung für kommende Woche.

## **6 Iglar Fragen:**

- Sessionmanagement: Ist HTTP ausreichend oder lowlevel?