

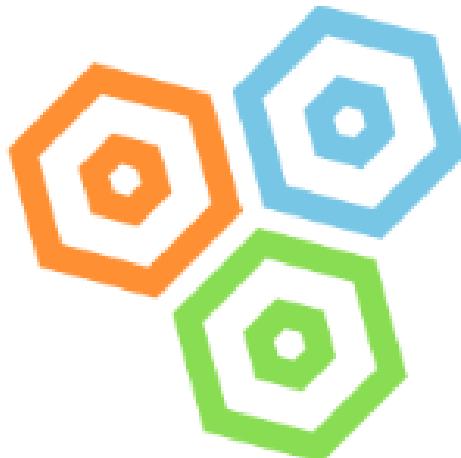
# PROJEKTDOKUMENTATION

## Release III

UNIVERSITÄT KASSEL  
SOFTWARE ENGINEERING I SS 19

**Projekt: Rundenbasiertes Strategiespiel  
RBSG - Enhanced Wars**

GRUPPE G



EnGineerinG  
<Problem/>

SCRUM MASTER: JAN MÜLLER  
PRODUCT OWNER: KEANU STÜCKRAD

8.7.2019 - 4.8.2019

## 1 Vorwort

Das Release III von Gruppe G des Projekts RBSG, ein rundenbasiertes Strategiespiel, soll im Rahmen des Moduls „Software Engineering I“ im Fachbereich „Software Engineering“ im Sommersemester 2019 mit dieser Dokumentation festgehalten werden.

Das Vorgehen in diesem Modul ist so gegliedert, dass es vier Releases mit jeweils zwei Sprints über vier Monate gibt. Das Projekt begann am 13.5.2019 und wird bis zum 1.9.2019 laufen. Ein Client für einen Klon vom Klassiker „Advanced Wars“ soll entwickelt werden. Dabei ist das Team über die vier Releases immer in einen Scrum Master, einen Product Owner und mehrere Developer eingeteilt.

Die Dokumentation wird eine Einleitung in das Release, danach die Zusammenfassung der beiden Sprints und eine abschließende Releaseanalyse enthalten. In der Einleitung sollen die Anforderungen des Kunden, der letzte Standpunkt und die neuen MockUps erläutert werden. In den Sprintdokumentationen wird jeweils kurz das Ziel dargelegt und darauf sollen alle User Stories, Tasks oder Bugs aufgelistet werden. Am Ende wird es immer eine Zeitübersicht mit einer Analyse des Sprints geben. Das letzte Kapitel mit der Releaseanalyse wird aufzeigen, ob das Resultat den Anforderungen entspricht und ob das Team effektiv arbeitete.

# Inhaltsverzeichnis

<b>1 Vorwort</b>	<b>1</b>
<b>2 Release III</b>	<b>5</b>
2.1 Releaseanforderungen . . . . .	5
2.2 Standpunkt des letzten Releases . . . . .	6
2.2.1 Login . . . . .	6
2.2.2 Lobby . . . . .	8
2.2.3 Army Manager . . . . .	9
2.2.4 Waiting Room (Gamelobby) . . . . .	11
2.2.5 Ingame . . . . .	13
2.2.6 Weitere Features . . . . .	15
2.3 MockUps . . . . .	15
2.3.1 Lobby . . . . .	16
2.3.2 Waiting Room (Gamelobby) . . . . .	16
2.3.3 Ingame . . . . .	18
2.3.3.1 Einheit auswählen . . . . .	19
2.3.3.2 Einheit bewegen . . . . .	21
2.3.3.3 Einheit angreifen . . . . .	22
2.3.3.4 Spielende . . . . .	23
2.3.3.5 Weitere Features . . . . .	24
2.4 Domain Stories . . . . .	25
<b>3 Sprint V</b>	<b>26</b>
3.1 Sprintziel . . . . .	26
3.2 User Stories . . . . .	26
3.2.1 Waiting Room - Spielbeitritt anpassen . . . . .	26
3.2.2 Ingame - Spielfeld . . . . .	26
3.2.3 Ingame - Phase beenden . . . . .	27
3.2.4 Ingame - Einheit auswählen . . . . .	27
3.2.5 Ingame - Einheit bewegen . . . . .	27
3.2.6 Ingame - Einheit angreifen . . . . .	27
3.2.7 Ingame - Spielstatus anzeigen . . . . .	28
3.2.8 Ingame - Minikarte anzeigen . . . . .	28
3.2.9 Ingame - Chatintegration . . . . .	28
3.2.10 Ingame - Game Over . . . . .	29
3.2.11 Ingame - Game Won . . . . .	29
3.2.12 Lobby - Spectating . . . . .	29
3.2.13 Ingame - Spectating . . . . .	29
3.2.14 Ingame - Spectator Over . . . . .	30
3.2.15 Ingame - Einheiteninformationen . . . . .	30
3.2.16 Ingame - Musiksteuerung . . . . .	30
3.3 Tasks . . . . .	30
3.3.1 Servernachrichten . . . . .	30
3.3.2 Einheiteninformationen im Datenmodell . . . . .	31

3.4	User Stories im Backlog . . . . .	31
3.4.1	Optional - Lobby - Anzeigen der Einheiten einer Armee . . . . .	31
3.4.2	Optional - Ingame - Auf gegnerischen Aktionen scrollen . . . . .	31
3.4.3	Optional - Waiting Room - Minigame . . . . .	31
3.5	Zeitübersicht . . . . .	32
3.6	Analyse . . . . .	32
3.6.1	Burndown . . . . .	32
3.6.2	Abgeschlossene Stories . . . . .	33
3.6.2.1	Waiting Room - Spielbeitritt anpassen . . . . .	33
3.6.2.2	Ingame - Spielfeld . . . . .	33
3.6.2.3	Ingame - Phase beenden . . . . .	33
3.6.2.4	Ingame - Spielstatus anzeigen . . . . .	34
3.6.3	Angefangene Stories . . . . .	34
3.6.3.1	Ingame - Einheit auswählen . . . . .	34
3.6.3.2	Ingame - Einheit bewegen . . . . .	34
3.6.3.3	Ingame - Minikarte anzeigen . . . . .	34
3.6.4	Nicht abgeschlossene Stories/Tasks . . . . .	34
3.6.5	Entfernte Tasks . . . . .	35
3.6.6	Fazit . . . . .	35
<b>4</b>	<b>Sprint VI</b> . . . . .	<b>36</b>
4.1	Sprintziel . . . . .	36
4.2	Geänderte User Stories vom Sprint V . . . . .	36
4.2.1	Ingame - Minikarte anzeigen . . . . .	36
4.2.2	Ingame - Einheiteninformationen . . . . .	36
4.3	Übernommene User Stories aus Sprint V . . . . .	37
4.4	Geänderte Tasks vom Sprint V . . . . .	37
4.4.1	Einheiteninformationen im Datenmodell . . . . .	37
4.5	Übernommene Tasks aus Sprint V . . . . .	38
4.6	User Stories und Tasks im Backlog . . . . .	38
4.6.1	User Stories aus Sprint V . . . . .	38
4.6.2	Tasks aus Sprint V . . . . .	38
4.6.3	Ingame - Einheit bewegen bestätigen . . . . .	38
4.6.4	Ingame - Einheit angreifen bestätigen . . . . .	38
4.6.5	Ingame - Aktion abbrechen . . . . .	38
4.7	Zeitübersicht . . . . .	40
4.8	Analyse . . . . .	40
4.8.1	Burndown . . . . .	40
4.8.2	Ausreißer . . . . .	40
4.8.3	Abgeschlossen . . . . .	40
4.8.4	Angefangen . . . . .	40
4.8.5	Nicht abgeschlossen . . . . .	40
4.8.6	Entfernt . . . . .	41
4.8.7	Fazit . . . . .	41
<b>5</b>	<b>Abschluss Release III</b> . . . . .	<b>42</b>
5.1	Neue MockUps im Verlauf des Releases III . . . . .	42
5.2	Vergleich MockUps mit aktueller Implementation . . . . .	42

## **6 Quellenangaben**

**43**

## 2 Release III

Das Release III erstreckte sich über die vier Wochen vom 8.7.2019 bis zum 4.8.2019. Die zwei Sprints gingen vom 8.7.2019 bis zum 21.7.2019 und vom 22.7.2019 bis zum 4.8.2019.

Das Team war in diesem Release wie gefolgt aufgeteilt:

- Scrum Master:  Jan Müller
- Product Owner:  Keanu Stückrad
- Developer:
  1.  Georg Siebert
  2.  Juri Lozowoj
  3.  Omar Sood
  4.  Tobias Klipp

### 2.1 Releaseanforderungen

Folgende Mindestanforderungen wurden vereinbart:

1. Client: Waiting Room (Gamelobby)
  - Chatfunktion
  - Armeeauswahl
  - Senden eines Bereit-Signals
2. Client: Ingame
  - Bewegen und Angreifen von Einheiten
  - Minikarte
  - Chatfunktion
  - Anzeige aller Spieler
  - Anzeige der aktuellen Runde und Phase
  - Anzeige des Gewinners
  - Spiele verlassen
  - Beobachtermodus
3. Client: Lobby
  - Spielbeitritt als Beobachter
4. Qualitätssicherung
  - C0 Testabdeckung von 75%

## 2.2 Standpunkt des letzten Releases

Das Release II erstreckte sich vom 10.6.2019 bis zum 7.7.2019. Nach dem 7.7. waren folgende Teile des Clients fertig gestellt:

- Login
  - \* Anmelden
  - \* Registrieren
- Lobby
  - \* Spiel erstellen/beitreten
  - \* Anzeige der angemeldeten Spieler
  - \* Anzeige der aktiven Spiele
  - \* Unterstützung eines Chats
  - \* Ausloggen
- Army Manager
  - \* Konfigurieren eigener Armeen
  - \* Speichern von Armeen lokal und auf dem Server
- Waiting Room (Gamelobby)
  - \* Unterstützung eines Chats
  - \* Anzeige der beigetretenen Spieler
  - \* Zurück zur Lobby
- Ingame
  - \* Anzeige des initialen Spielgeschehens
  - \* Zurück zur Lobby

Weiterhin wurde bereits eine C0 Testabdeckung von mindestens 60% erreicht.

### 2.2.1 Login

Im Login kann der Nutzer sich anmelden oder sich sogar registrieren (siehe Abbildung 1).

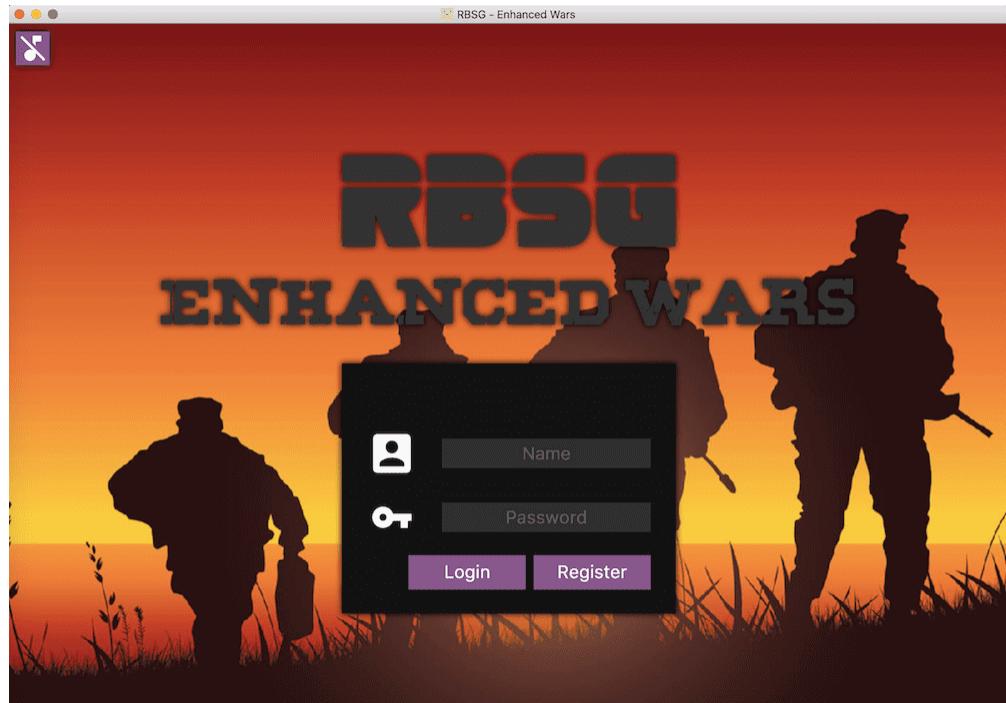


Abbildung 1: Release II: Login Szene

Versucht man sich anzumelden oder zu registrieren, erscheint ein Lade Indicator (siehe Abbildung 2) und die Buttons und Textfelder werden disabled. Macht der Nutzer dabei einen Fehler oder kann keine Verbindung zum Server hergestellt werden, erscheint eine Fehlermeldung (siehe Abbildung 2). Der Login ist internationalisiert und der Untertitel „Enhanced Wars“ ist animiert. Weiterhin kann man die Musik über den Button ein- und ausschalten.

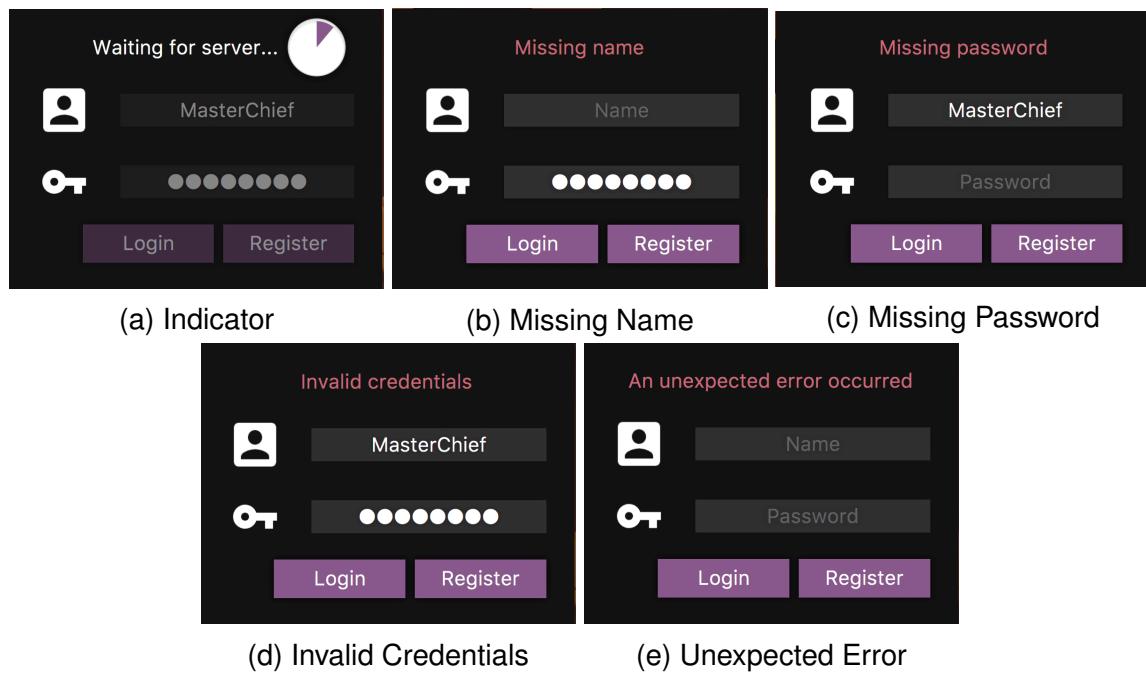


Abbildung 2: Release II: Login Form

## 2.2.2 Lobby

Die Lobby zeigt eine Liste von allen Spielern (die nicht im Spiel sind) und allen Spielen an. Diese aktualisieren sich, wenn der Server die entsprechende Nachricht schickt. Man kann sich über die Buttons oben rechts ausloggen, die Sprache ändern oder die Musik an- und ausschalten. Der Chat wird unten links angezeigt. Rechts sieht man eine Liste mit allen vollständigen Armeen. Hat man keine vollständige Armee ausgewählt, kann man kein Spiel erstellen (siehe Abbildung 3).

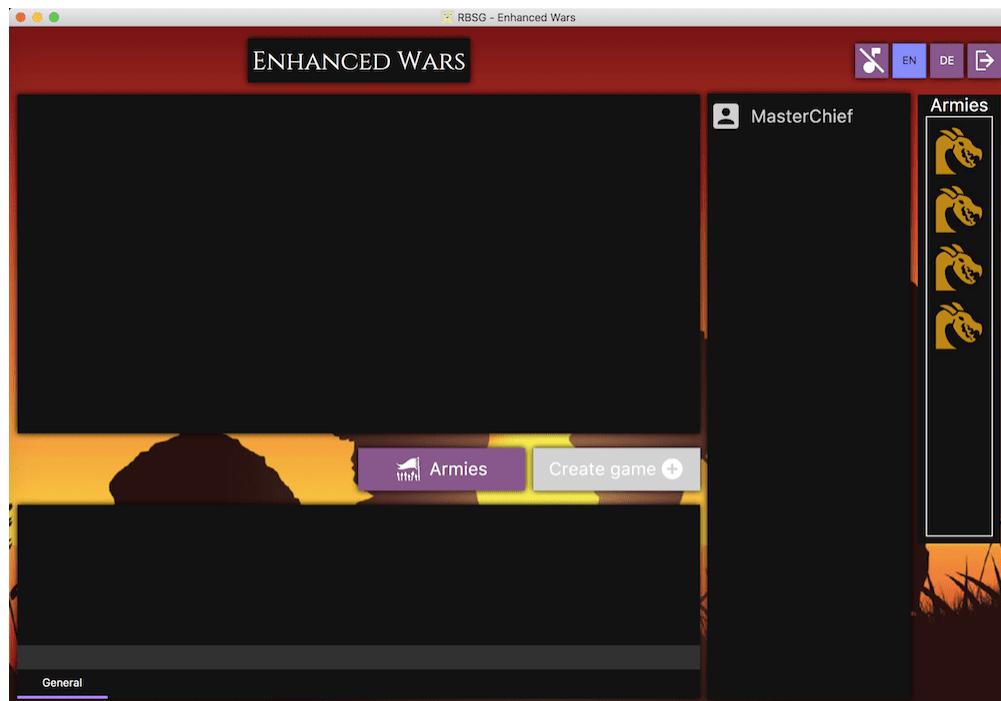


Abbildung 3: Release II: Lobby Szene: Keine Armee ausgewählt

Hat man eine Armee ausgewählt, kann man ein Spiel erstellen oder einem Spiel beitreten (siehe Abbildung 4 oder 6). Spiel erstellen geht über Create Game (siehe Abbildung 5). Der Nutzer hat die Auswahl zwischen einem 2-Spieler-Spiel und einem 4-Spieler-Spiel. Danach wird ein Autojoin getriggert. Die Lobby ist auch internationalisiert (vgl. Abbildung 5 mit 6).

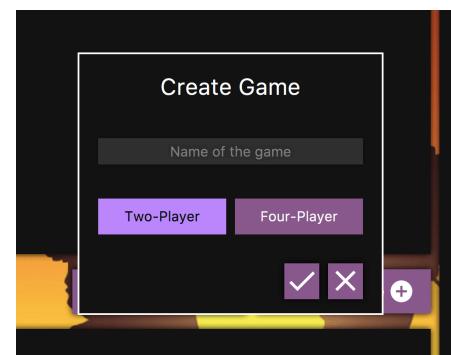


Abbildung 4: Release II: Create Game



Abbildung 5: Release II: Lobby Szene: Armee ausgewählt



Abbildung 6: Release II: Lobby Szene: Sprache DE

### 2.2.3 Army Manager

Im Army Manager sieht der Nutzer eine Liste der Einheiten, die man wählen kann, und eine Liste der Einheiten, die in der gerade rechts ausgewählten Armee sind. In der Mitte steht der Name der ausgewählten Armee, ein + und ein - Button, um Einheiten der Armee hinzuzufügen oder um sie zu löschen und ein Zähler, der anzeigt, wie voll die Armee ist.

Neben der Einheitenliste ist oben eine Vorschau der gerade ausgewählten Einheit. Neben diesem Feld werden die Properties der Einheit angezeigt und welche Einheiten die ausgewählte Einheit angreifen kann. Hier stehen bei keiner Auswahl Fragezeichen (siehe Abbildung 7), ansonsten

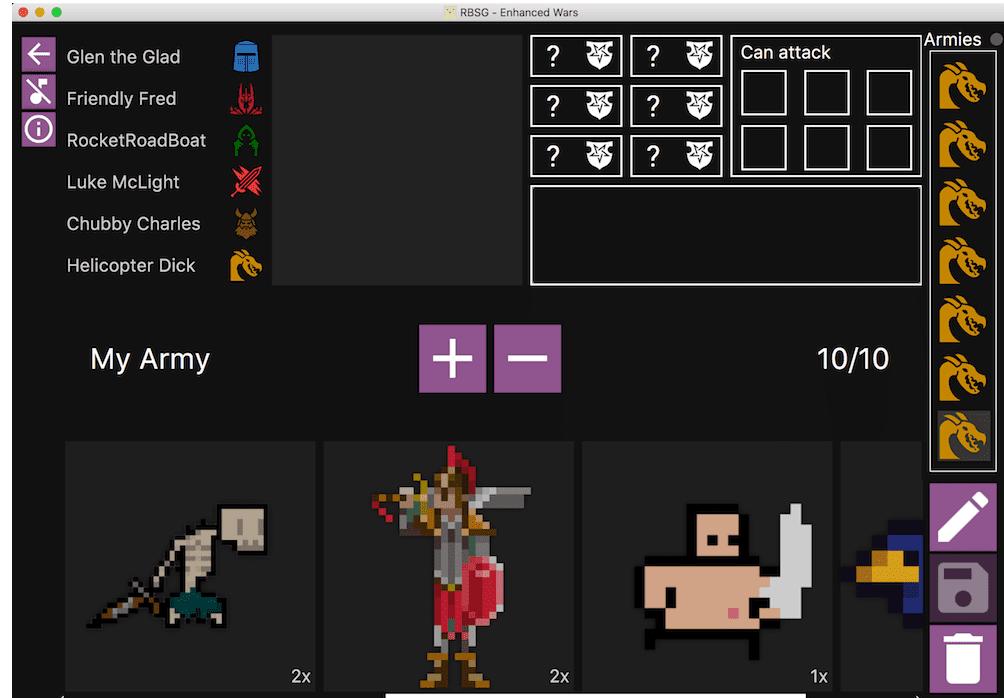


Abbildung 7: Release II: Army Manager Szene: Keine Einheit ausgewählt

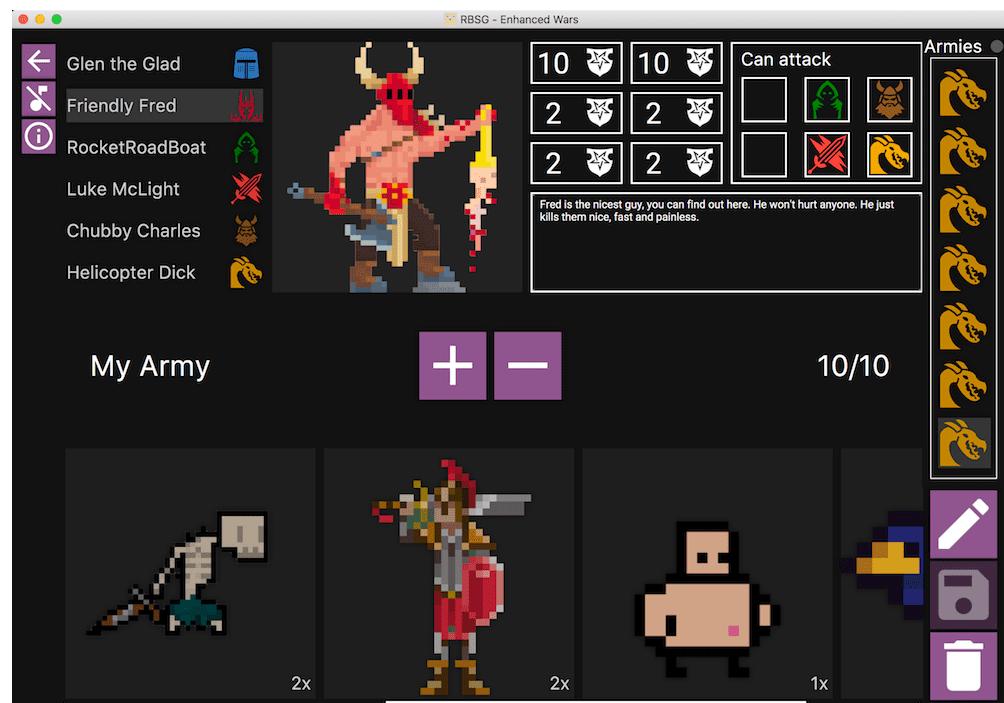


Abbildung 8: Release II: Army Manager Szene: Einheit ausgewählt

Werte (siehe Abbildung 8). Darunter steht der Lore Text der ausgewählten Einheit. Oben rechts sind Buttons, um zurück zur Lobby zu gehen, um die Musik ein- oder auszuschalten und ein Button der die Properties in einem neuen Menu erklärt (siehe Abbildung 9). Unten rechts sind Buttons, um eine Armee komplett

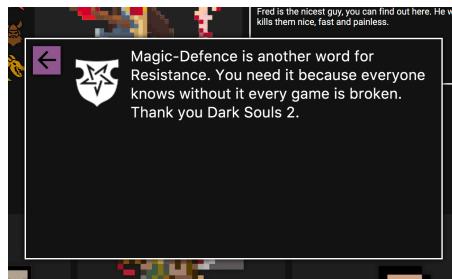


Abbildung 9: Release II: Property Info

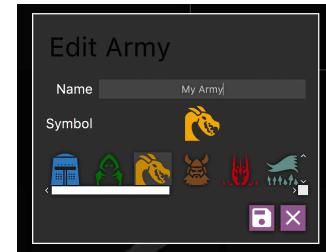


Abbildung 10: Release II: Armee bearbeiten

zu löschen, um sie zu speichern (wird disabled, wenn sich nichts geändert hat (siehe Abbildung 7), wenn der Nutzer etwas ändert wieder enabled (siehe Abbildung 11)) und ein Button, um den Armeenamen zu ändern oder das Icon zu wechseln. Dazu öffnet sich ein weiteres Menu (siehe Abbildung 10).

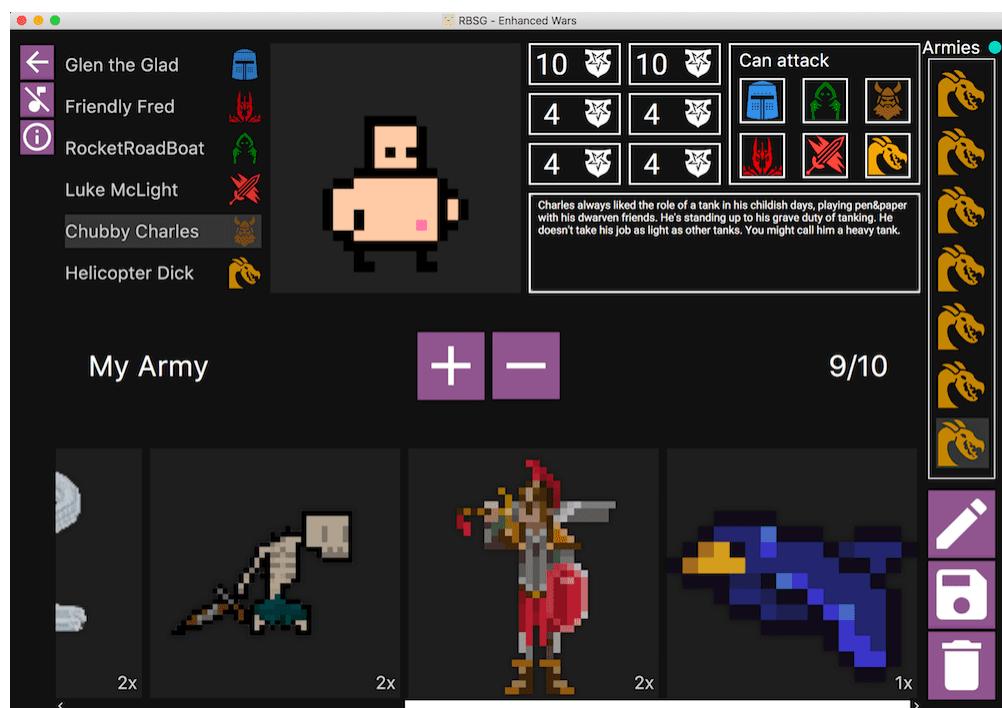


Abbildung 11: Release II: Army Manager Szene: Speicherbar

## 2.2.4 Waiting Room (Gamelobby)

Tritt der Nutzer dem Warteraum bei, dann sieht er oben rechts Buttons. Einmal einen Button zum Verlassen des Spiels in die Lobby, ein Button, um die Musik ein- und auszuschalten, und ein Button, um Informationen anzuzeigen. Da dieser Button noch keine

spezielle Funktion hat, wechselt man mit jenem Testweise auf das Spielfeld (Ingame). Oben in der Mitte wird eine Kartenvorschau angezeigt. Links und rechts daneben sind die Playercards. Je nachdem ob man ein 2- Spieler-Spiel oder ein 4-Spieler-Spiel gestartet hat, sind dort zwei oder vier Spielerkarten (siehe Abbildung 12 und 13).

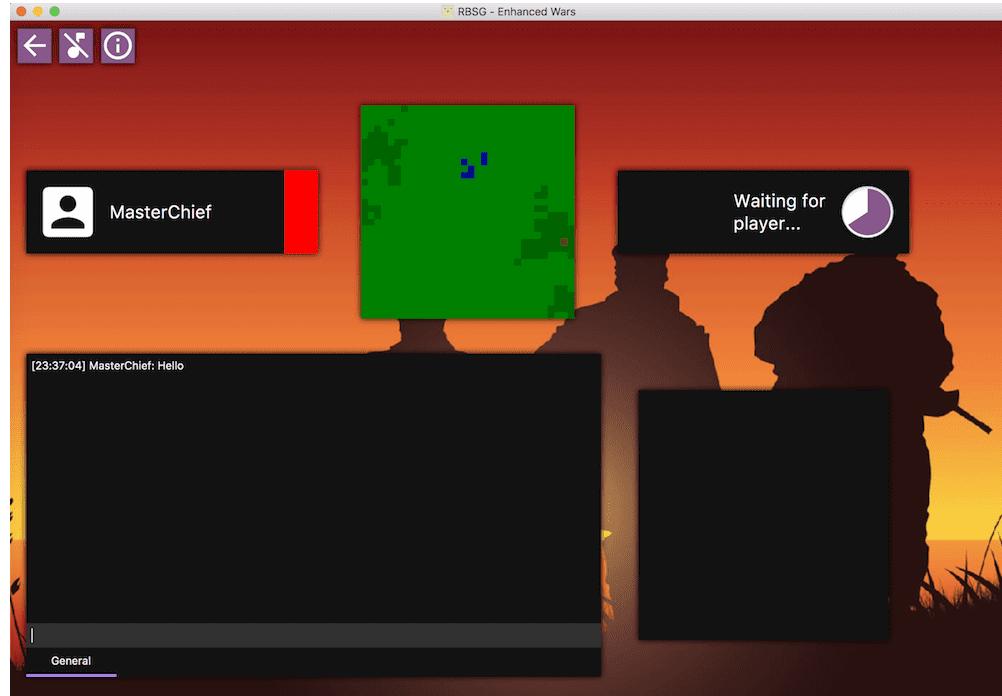


Abbildung 12: Release II: Waiting Room Szene: 2 Spieler

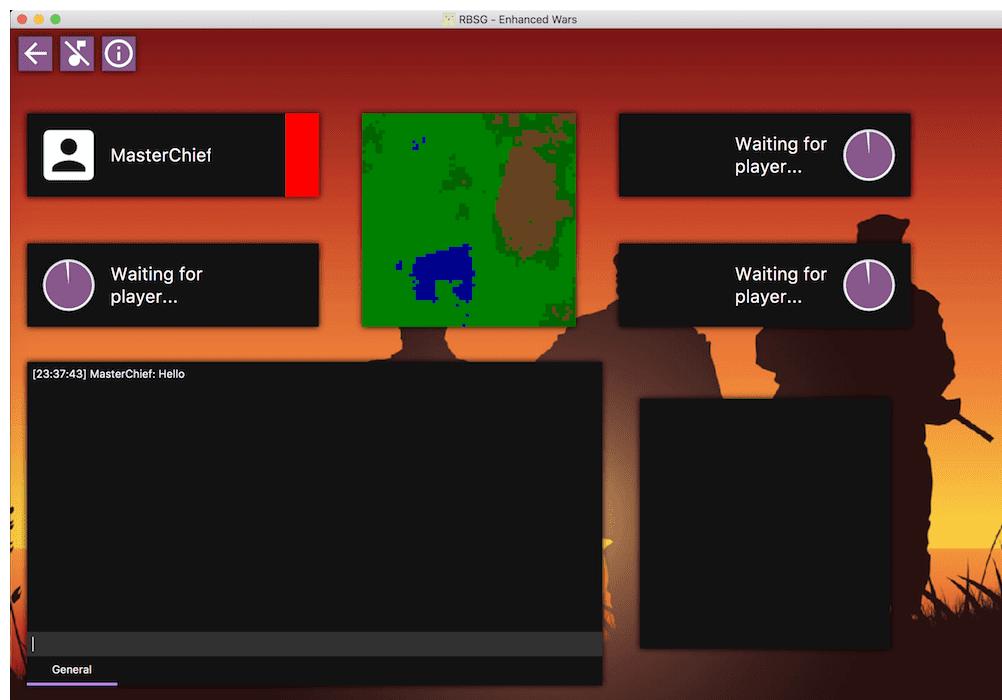


Abbildung 13: Release II: Waiting Room Szene: 4 Spieler

Auf den Spielerkarten stehen der Name des jeweiligen Spielers und der Nutzer sieht die Farbe. Ist noch kein Spieler da, wird ein Indikator angezeigt. Unten ist der Chat und daneben eine freie Pane für ein Minigame, dass noch implementiert werden soll.

## 2.2.5 Ingame

Die Ingame Szene zeigt bisher nur die Karte und das initiale Spielgeschehen. Die initialen Einheiten werden auch angezeigt. Weiterhin gibt es einen Button, der zurück in die Lobby geht. Zwei weitere Buttons kontrollieren das Zoomen (siehe Abbildung 14, 15 und 16).



Abbildung 14: Release II: Ingame Szene



Abbildung 15: Release II: Ingame Szene: Zoom In

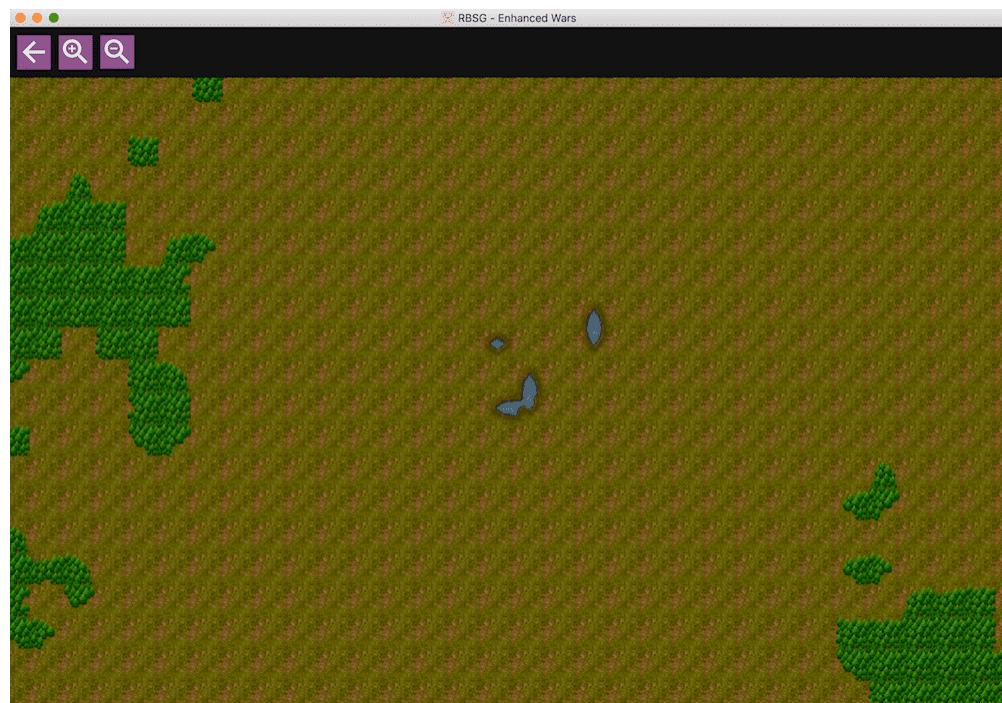


Abbildung 16: Release II: Ingame Szene: Zoom Out

## 2.2.6 Weitere Features

Weiterhin wurde seit den ersten beiden Releases ein Dark Theme Styling implementiert, wie man auf allen vorangegangenen Abbildungen sehen kann. Musik und Internationalisierung war auch völlig optional und funktioniert nun in jeder Szene. Der Nutzer kann auch nach Belieben Chuck Norris Zitate über den Chat-Befehl „\chuckMe“ im Chat versenden und weiterhin wird nun angezeigt, wann jede Nachricht geschickt wurde. Wenn man eine private Nachricht bekommt, wird dies auch auf dem jeweiligen Tab mit der secondary Farbe signalisiert. Die Kartenvorschau im Warteraum war auch optional, genauso wie das Zoomen Ingame. Den Armeenamen, sowie das Icon dazu ändern zu können, war auch optional. Der Autojoin nach dem Create Game war auch nicht verlangt, funktioniert nun aber bestens. Ein weiteres sehr wichtiges Feature ist, dass der Client den Spieler nun Fehlermeldungen (siehe Abbildung 17 und 18) schickt, wenn der Server einen WebSocket unerwartet schließt. Man wird je nach Fehler in den Login oder die Lobby zurückgeschickt. Außerdem gibt es ein Fenster, in dem der Nutzer noch einmal aussuchen kann, ob er sich wirklich ausloggen möchte oder ob er wirklich das Spiel verlassen möchte (siehe Abbildung 19 und 20).



Abbildung 17: Release II: Login Fehler



Abbildung 18: Release II: Spielfehler

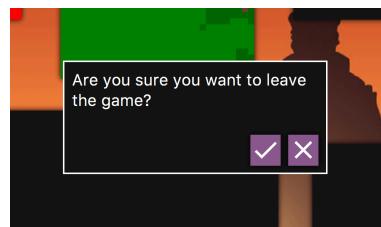


Abbildung 19: Release II: Spiel verlassen

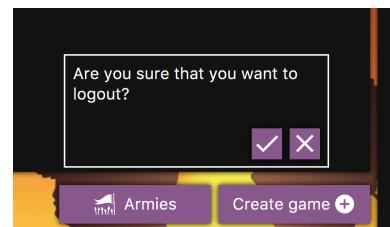


Abbildung 20: Release II: Logout

Die C0 Testabdeckung betrug zum Ende des Releases II 77%.

97% classes, 77% lines covered in package 'de.uniks.se19.team_g'			
Element	Class, %	Method, %	Line, %
project_rbsg	97% (380/389)	80% (1586/1967)	77% (7915/10223)

Abbildung 21: Release II: C0 Testabdeckung (Siehe Line, %)

## 2.3 MockUps

Aufgrund der Releaseanforderungen wurden für das Kundentreffen am 1.7.2019 MockUps zu der neuen Ingame Szene erstellt. Weiterhin gab es auch noch MockUps zur Lobby und zum Waiting Room. Diese wurden nach dem Erscheinen der Serverdokumentation noch

erweitert. Im sechsten Sprint wurden die MockUps auch noch einmal angepasst, da sich grundlegende Änderungen ergeben haben (Ein Kontextmenü wurde durch eine Sidebar ersetzt. Die MockUps waren im Stil des Dark Theme und sollten an die UI aus den vorherigen Releases anknüpfen).

### 2.3.1 Lobby

Es wurde gefordert, in der Lobby ein Button hinzuzufügen, der es dem Nutzer erlaubt einem Beobachtungsmodus beizutreten. Des Weiteren sollte die Armeeauswahl aus der Lobby entfernt werden (und in den Waiting Room verlegt werden).

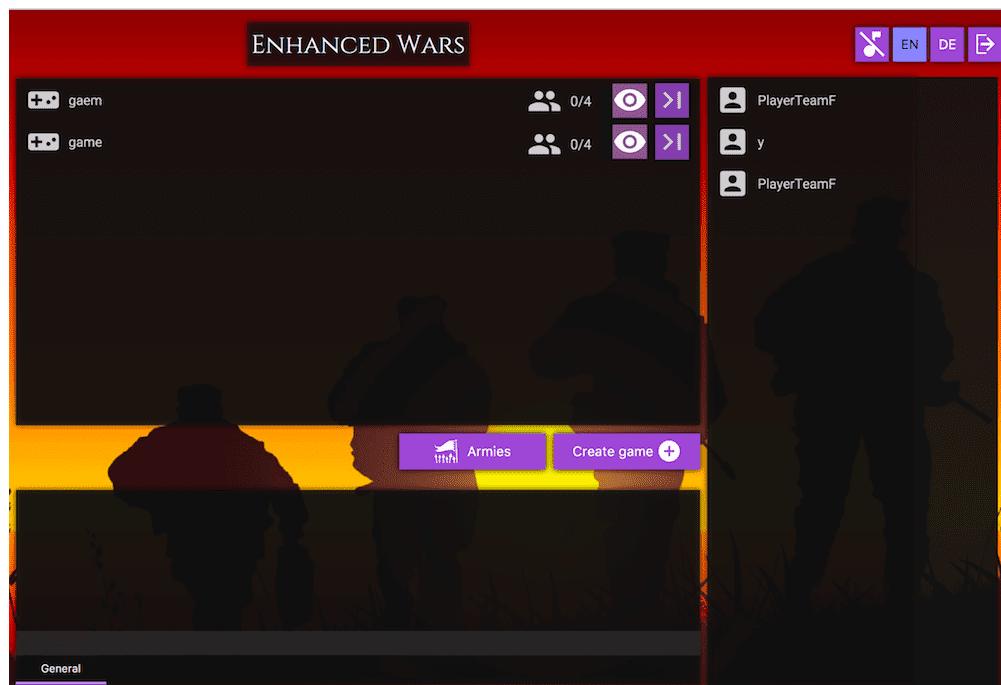


Abbildung 22: MockUp: Beobachtungsmodus Lobby

### 2.3.2 Waiting Room (Gamelobby)

Da der Waiting Room im Release II (siehe Kapitel 2.2.4) schon größtenteils vorhanden war, musste nur noch das Ready geben hinzugefügt werden. Dies sollte bei uns passieren, wenn man eine Armee auswählt (siehe Abbildung 24). Die Armeeauswahl musste deswegen von der Lobby in den Waiting Room umgezogen werden. Spieler, die bereits bereit waren, sollten in ihrer Playercard in der secondary Farbe hinterlegt werden und die Icons sollten sich schwarz färben. Wenn alle Spieler das Ready gaben, wurde ein Overlay angezeigt, in dem ein Indicator lud, bis zur Ingame Szene gewechselt wurde (siehe Abbildung 25). Auf den MockUps waren auch zwei optionale Features zu sehen. Dies war einmal das TicTacToe Spiel unten rechts, das gegebenenfalls implementiert werden sollte. Und außerdem sollte oben in der Mitte der Name des Spiels angezeigt werden (siehe auf den MockUps mit Beispielnamen „Enhanced Wars“).

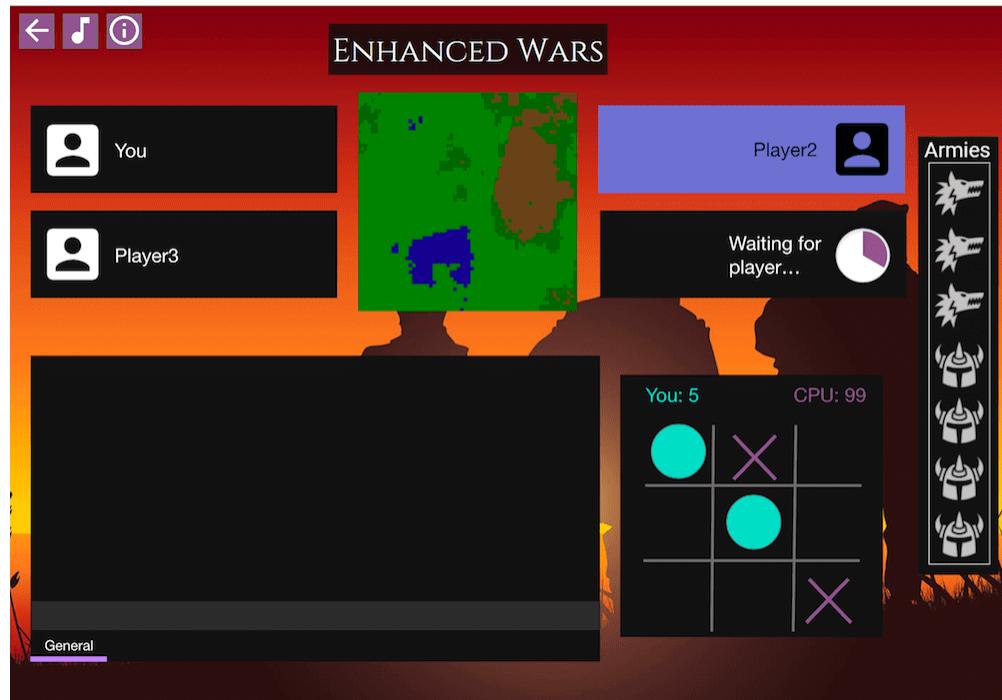


Abbildung 23: MockUp: Nutzer nicht bereit

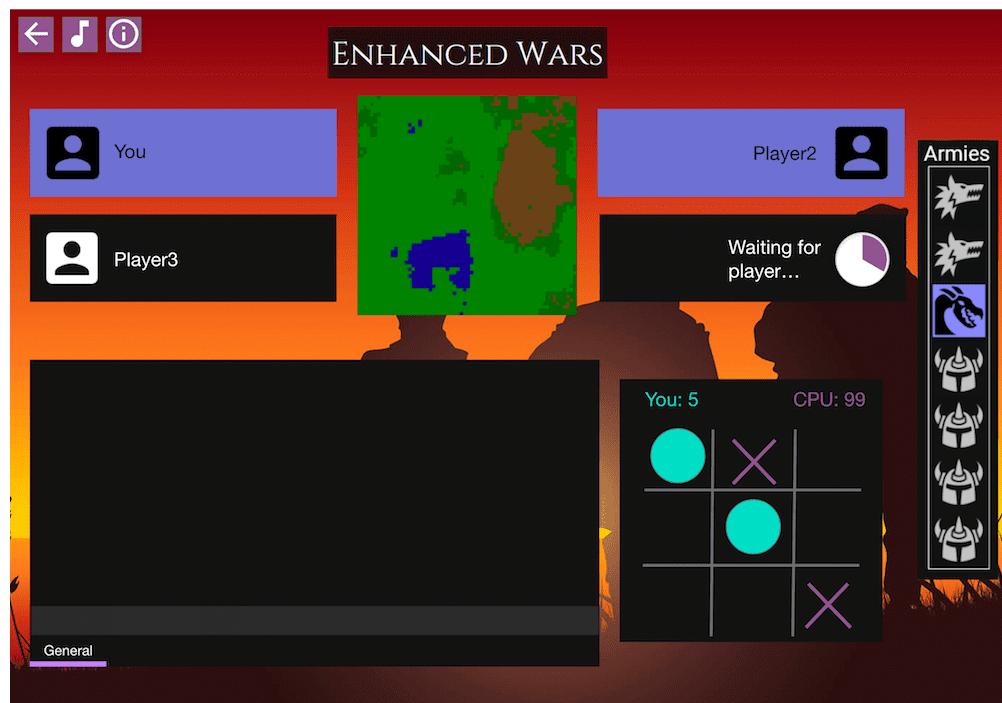


Abbildung 24: MockUp: Nutzer bereit

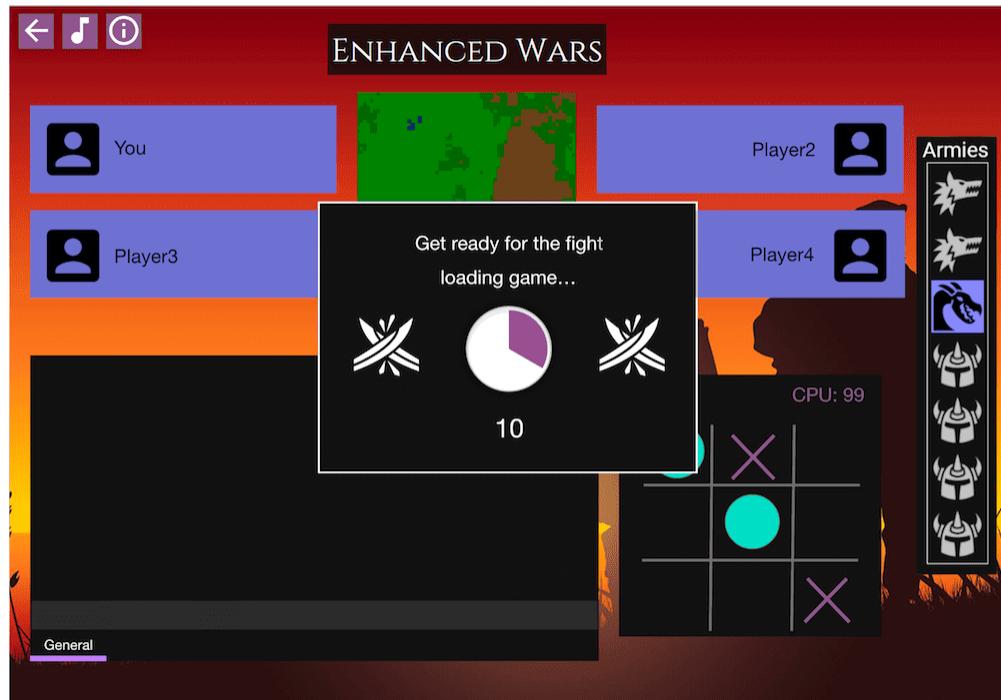


Abbildung 25: MockUp: Spielstart

### 2.3.3 Ingame

Bisher zeigte die Ingame Szene nur das initiale Spielgeschehen. In diesem Release sollte diese Szene hauptsächlich erweitert werden. Chat, die Minikarte und die Spieler mussten angezeigt werden. Der Chat und die Spielerkarten sollten auch über die jeweiligen Buttons oben in der Leiste eingeklappt werden können und Overlays über dem Spielfeld sein. Die Minikarte sollte unten rechts unter der Sidebar angezeigt werden. Die Sidebar war eine neue feste Box rechts neben dem Spielfeld. Diese sollte die Einheiteninformationen und die Bestätigen/Abbrechen Buttons enthalten. Die aktuelle Runde und die aktuelle Phase sollten oben über der Sidebar angezeigt werden. Der aktuelle Spieler sollte in seiner Playercard mit der secondary Farbe hinterlegt werden. In der oberen Leiste kam außerdem der Musik Button zum ein- und ausschalten der Musik hinzu.

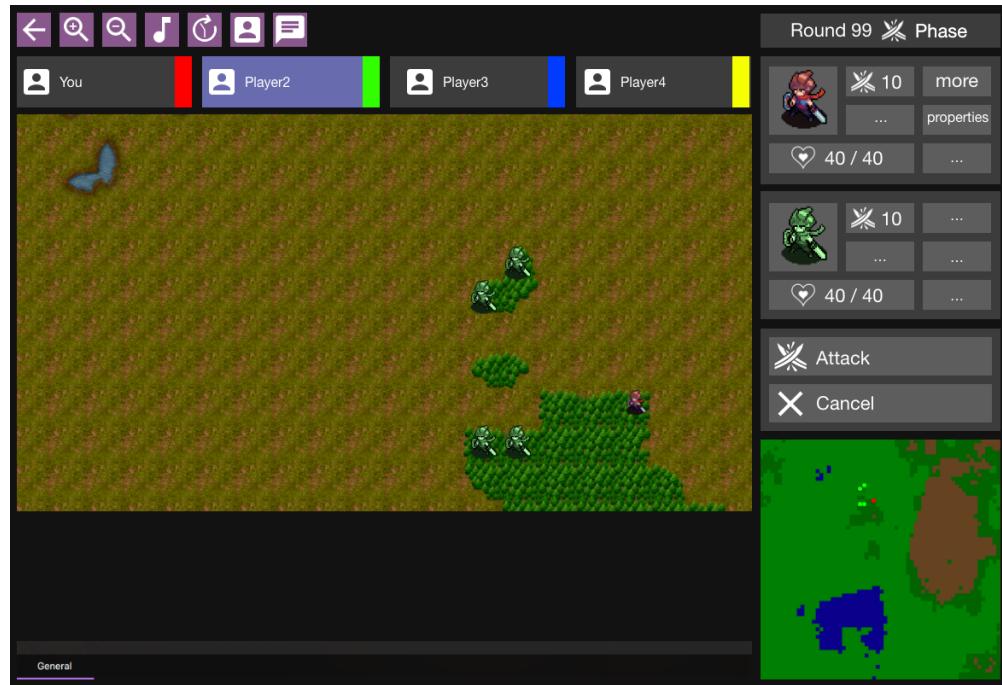


Abbildung 26: MockUp: Ingame

### 2.3.3.1 Einheit auswählen

Der Server stellte drei Phasen zur Verfügung, wenn ein Spieler an der Reihe war. Eine Bewegungsphase, in der mindestens eine Einheit bewegt werden musste, eine Angriffsphase, die übersprungen werden konnte und eine restliche Bewegungspunktephase, die ebenfalls übersprungen werden konnte.



Abbildung 27: MockUp: Phase 1: Auswählen

Nachdem alle drei Phasen vom aktiven Spieler beendet wurden, war die Runde des Spielers vorbei. Wenn der Nutzer selbst an der Reihe war und in einer seiner drei Phasen auf eine Einheit klickte, sollte der Bewegungsradius oder Angriffsradius angezeigt werden. Klickte der Nutzer wieder auf die bereits ausgewählte Einheit, auf eine andere Einheit oder auf ein freies Feld, dann sollte die Einheit nicht mehr ausgewählt sein.



Abbildung 28: MockUp: Phase 2: Auswählen



Abbildung 29: MockUp: Phase 3: Auswählen

In einer gegnerischen Phase sollte die Einheit, die vom Gegner in seinem Client ausgewählt wurde, auch beim Nutzer, währenddessen der Nutzer dem Gegner zuschaute, auf die selbe Art und Weise ausgewählt werden. In Phase 1 sollten zusätzlich zum Bewegungsradius auch alle möglichen Angriffsziele markiert werden (siehe Abbildung 27). In Phase 2 sollten nur alle möglichen Angriffsziele angezeigt werden (siehe Abbildung 28). In Phase 3 sollte ein verkürzter Bewegungsradius ohne Angriffsziele angezeigt werden (siehe Abbildung 29), je nachdem wie weit sich eine Einheit noch bewegen konnte.

### 2.3.3.2 Einheit bewegen

Nachdem der Nutzer in Phase 1 oder 3 eine Einheit ausgewählte und auf ein Feld klickte, sollte die Einheit auf das Feld bewegt werden, sobald der kürzeste Pfad berechnet und zum Server geschickt wurde. Falls nach der Bewegung noch Bewegungspunkte übrig waren, sollte die Einheit weiterhin ausgewählt bleiben und die Bewegungsreichweite dementsprechend verkleinert werden. War ein Gegner an der Reihe und der Server sendete eine bewegte Einheit, dann sollte auch zuerst die Bewegungsreichweite angezeigt werden und danach (kurz verzögert) die Einheit bewegt werden. Später sollte der Nutzer über eine Box in der Sidebar die Möglichkeit haben, seinen Zug doch noch zu widerrufen. Dabei sollte auf dem geklicktem Feld ein Icon angezeigt werden, bis der Nutzer die Aktion bestätigte oder abbruch (siehe Abbildung 30 oder 31). War ein Gegner an der Reihe, wurde der Text und das Icon in der Box nicht angezeigt.



Abbildung 30: MockUp: Bewegen bestätigen (Phase 1)



Abbildung 31: MockUp: Bewegen bestätigen (Phase 3)

### 2.3.3.3 Einheit angreifen

Nachdem der Nutzer in Phase 2 eine Einheit ausgewählte und alle möglichen Angriffsziele angezeigt wurden, konnte er eine gegnerische Einheit neben ihm auswählen (insofern eine vorhanden war) und sie angreifen.



Abbildung 32: MockUp: Angreifen bestätigen

War ein Gegner an der Reihe und der Server sendete eine angegriffene Einheit, dann sollten auch zuerst die möglichen Angriffsziele angezeigt werden und danach (kurz verzögert) sollte die Einheit angreifen. Später sollte dies auch noch über ein Box in der Sidebar bestätigt werden, um dem Nutzer die Möglichkeit zu geben, seinen Angriff doch nicht durchzuführen. Dabei sollte auf dem geklicktem Feld ein Icon angezeigt werden, bis der Nutzer die Aktion bestätigte oder abbruch (siehe Abbildung 32). War ein Gegner an der Reihe, wurde der Text und das Icon in der Box nicht angezeigt.

### 2.3.3.4 Spielende

Wenn ein Spieler verlor, da er keine Einheiten mehr hatte, sollte sich das Icon und der Name in seiner Playercard schwarz färben. Bei diesem Nutzer sollte ein Overlay angezeigt werden, in dem stand, dass er verloren hatte. Außerdem konnte er über das Overlay entscheiden, ob er sich zurück in die Lobby ging oder dem Spiel als Spectator beitrat (siehe Abbildung 33). Wenn nur noch ein Spieler übrig war, gewann er das Spiel. Diesem Nutzer sollte auch ein Overlay angezeigt werden, durch das er zurück zur Lobby wechselte (siehe Abbildung 34). Bei anderen Nutzern, die sich im Spectatormodus befanden, sollte auch ein ähnliches Overlay mit der gleichen Funktionalität angezeigt werden (siehe Abbildung 35).

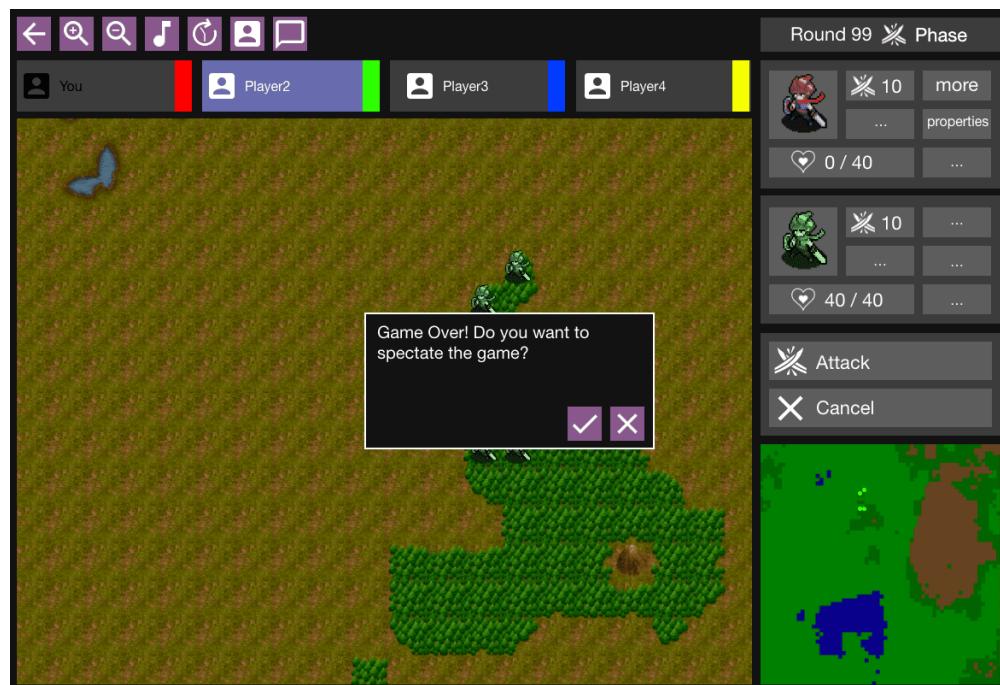


Abbildung 33: MockUp: Spiel verloren



Abbildung 34: MockUp: Spiel gewonnen



Abbildung 35: MockUp: Beobachtungsmodus: Spiel vorbei

### 2.3.3.5 Weitere Features

Es sollte für den Nutzer möglich sein, über einen Button die Phase beenden zu können und in die nächste zu wechseln. Nach Beenden aller drei Phasen, musste auch die Runde beendet werden. Wenn ein Gegner an der Reihe war, musste der Phase beenden Button disabled sein. Beim Beenden der Phasen sollte immer ein Overlay angezeigt werden

(siehe Abbildung 36).

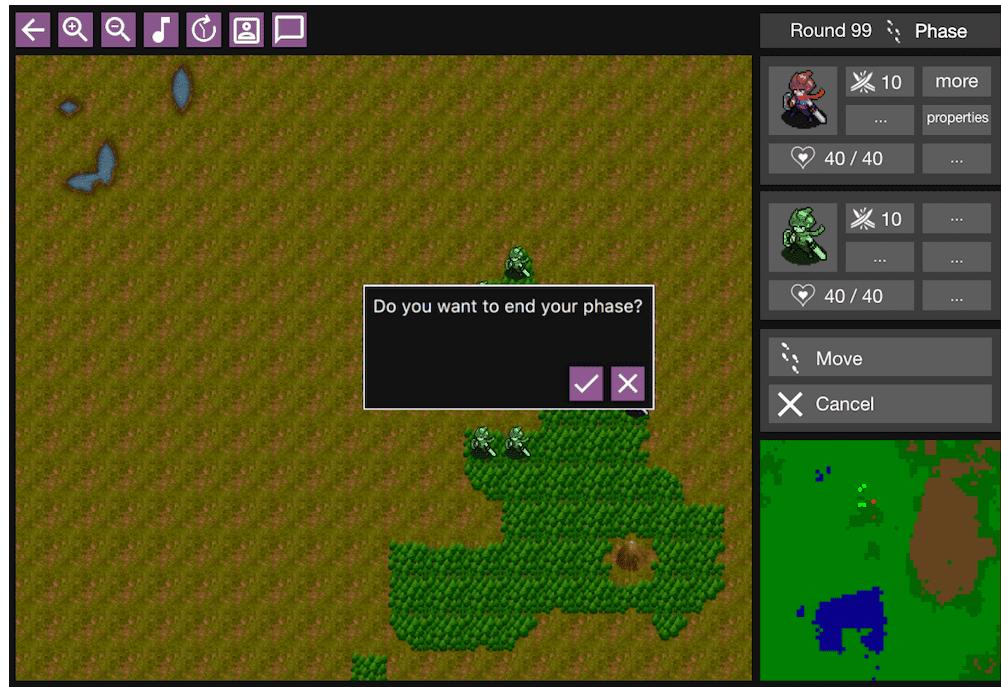


Abbildung 36: MockUp: Phase beenden

Es gab ein weiteres Feature, bei dem der Nutzer über eine Einheit hovern konnte, und je nachdem über welche Einheit der Nutzer hoerte, aktualisierten sich alle Properties und das Bild der gehoerten Einheit in der Sidebar. Die Einheiten des Spielers, der an der Reihe war, sollten in der oberen Box der Sidebar angezeigt werden. Die Einheiten der restlichen Spieler, die nicht an der Reihe waren, sollten in der unteren Box angezeigt werden. Zu Beginn jeder neuen Runde und sollten die beiden Boxen immer wieder leer sein und sich erst wieder füllen, wenn über eine Einheit gehoert bzw. wenn eine Aktion mit einer Einheit ausgeführt wurde.

## 2.4 Domain Stories

Domain Stories wurden einmal zum Ready geben für den Waiting Room und einmal für das Bewegen von Einheiten Ingame angefertigt.

TODO: Bilder einpflegen und beschreiben?

Die Domain Stories wurden vor dem Release der Serverdokumentation für das Kundentreffen erstellt. Deswegen zeigen sie leichte Abweichungen vom tatsächlichen Verhalten des Servers. Zum Beispiel sendet der Server Spiel starten erst, wenn irgendein Spieler den Befehl zum Spiel starten an den Server schickt, nachdem alle eine Ready gegeben haben. Wir nahmen an, dass der Server automatisch das Spiel startet, nachdem alle ready sind. Unsere Move Story trifft teilweise den Server. Der Unterschied zu unserer Story ist, dass wir vorher nicht davon ausgingen, dass eine Runde des Spielers in die drei Phasen unterteilt ist. Deswegen zeigt unsere Story auf dem MockUp oben rechts schon die bewegte Einheit, mit der jetzt eigentlich angegriffen werden sollte. Da dies vom Server her nicht funktioniert, müssen wir dies auf die drei Phasen unterteilen.

## 3 Sprint V

Der fünfte Sprint erstreckte sich über den Zeitraum vom 8.7.2019 bis zum 21.7.2019. Es wurden 116 Storypoints für alle User Stories geschätzt.

### 3.1 Sprintziel

Es sollte nach dem fünften Sprint möglich sein:

- Einem Spiel beizutreten
- Die Einheiten auf dem Spielfeld und der Minikarte zu sehen
- Einheiten auszuwählen, zu bewegen und anzugreifen
- Den Spielstatus zu sehen und Phasen zu beenden

Falls die Entwicklung schneller als erwartet voranschreiten sollte, würden der Ingamechat und das Spielende noch implementiert werden. Sollten alle Ziele abgeschlossen werden, würden über 80 Storypoints abgearbeitet werden, was bei vier Entwicklern als ein guter Fortschritt gewertet werden könnte.

### 3.2 User Stories

Die User Stories, die vom Product Owner erstellt wurden, sind wie folgt aufgebaut: Dem Titel kann entnommen werden zu welcher Szene die Story gehört. Anschließend werden die Story Points und der zugeteilte Entwickler genannt. Danach folgt das Ziel und die Story selbst. Das Ziel fasst die Subtasks, die der Scrum Master erstellt hat, zusammen.

#### 3.2.1 Waiting Room - Spielbeitritt anpassen

**Zuteilung** Diese User Story wurde auf 5 Storypoints geschätzt und an Omar Sood zugewiesen.

**Ziel** Der Spielbeitritt sollte an die Serveränderungen angepasst werden. Das Auswählen einer Armee sollte dem Server ein Bereit-Signal senden und das Spiel sollte starten, sobald alle Spieler bereit sind.

**Story** Karli ist einem Spiel beigetreten und befindet sich im Wartrerraum. Alle anderen Spieler sind bereit. Karli wählt eine Armee aus. Da nun alle Spieler bereit sind, startet das Spiel (und es findet ein Szenenwechsel statt). TODO: Verbindung zur Domain Story

#### 3.2.2 Ingame - Spielfeld

**Zuteilung** Diese User Story wurde auf 13 Storypoints geschätzt und an Georg Siebert zugewiesen.

**Ziel** Das Spielfeld sollte die Einheiten anzeigen. Jedes Feld musste anklickbar sein.

**Story** Karli ist bereits einem Spiel beigetreten und sieht das Spielfeld. Karli sieht die Einheiten auf dem Spielfeld. Karli möchte nun auf ein Feld (einer Einheit) klicken.

### 3.2.3 Ingame - Phase beenden

**Zuteilung** Diese User Story wurde auf 3 Storypoints geschätzt und an Juri Lozowoj zugeteilt.

**Ziel** Es sollte einen Button geben, welcher bei Betätigung die aktuelle Phase beendete. Dies sollte nur möglich sein, wenn der Nutzer auch seine Phase beenden konnte.

**Story** Karli befindet sich bereits in einem Spiel. Karli hat gerade einen Teil ihrer Armee bewegt (oder mit einem Teil angegriffen) und möchte nun die jeweilige Phase beenden. Dafür klickt Karli auf den entsprechenden Button (siehe Mockup). Die Phase wird beendet und die nächste tritt ein (bzw. der nächste Spieler ist an der Reihe).

### 3.2.4 Ingame - Einheit auswählen

**Zuteilung** Diese User Story wurde auf 13 Storypoints geschätzt und an Juri Lozowoj zugeteilt.

**Ziel** Durch Anklicken einer Einheit sollte diese Ausgewählt werden. Danach musste die Bewegungs- und Angriffsreichweite sichtbar sein, je nachdem in welcher Phase der Nutzer sich befand.

**Story** Karli befindet sich bereits in einem Spiel. Karli wählt eine Einheit durch Klicken auf ihre Graphik aus. Gültige Bewegungsziele werden blau und gültige Angriffsziele rot markiert (siehe Mockups).

### 3.2.5 Ingame - Einheit bewegen

**Zuteilung** Diese User Story wurde auf 13 Storypoints geschätzt und an Omar Sood zugeteilt.

**Ziel** Für die Bewegung einer Einheit sollte der kürzeste Pfad zum Zielfeld berechnet werden. Anschließend musste eine Nachricht an den Server gesendet und dessen Antwort korrekt verarbeitet werden.

**Story** Karli befindet sich in der Bewegungsphase oder der restlichen Bewegungspunktephase und hat bereits auf eine Einheit geklickt und sieht alle Felder, auf welche die Einheit bewegt werden könnte (je nachdem ob Karli in der 1. oder 3. Phase ist, mit oder ohne rote Felder). Karli klickt auf ein Feld in Bewegungsreichweite und die Einheit wird bewegt. TODO: Verbindung zur Domain Story

### 3.2.6 Ingame - Einheit angreifen

**Zuteilung** Diese User Story wurde auf 5 Storypoints geschätzt und an Omar Sood zugeteilt.

**Ziel** Es sollte möglich sein, gegnerische Einheiten anzugreifen. Entsprechende Nachrichten sollten an den Server gesendet und dessen Antworten korrekt verarbeitet werden.

**Story** Karli ist in der Angriffsphase und hat bereits eine Einheit ausgewählt. Karli sieht alle Felder in Rot, die angegriffen werden können. Karli wählt eine gegnerische Einheit auf diesen Feldern aus und greift sie an. Die eigene Einheit ist nicht mehr auswählbar und ist nicht mehr ausgewählt.

### 3.2.7 Ingame - Spielstatus anzeigen

**Zuteilung** Diese User Story wurde auf 8 Storypoints geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Es sollte einen Button zum Ein- und Ausblenden der Spielerinformationen implementiert werden. Der aktive Spieler sollte hervorgehoben werden.

**Story** Karli befindet sich bereits in einem Spiel. In der Menüleiste sieht Karli einen Button zum Anzeigen der Spieler und die aktuelle Rundennummer. Karli klickt auf diesen Button. Eine Anzeige der Spieler wird eingeblendet (siehe Mockup). Der aktive Spieler ist hervorgehoben (in der selected-Color siehe Stylesheet bzw. MockUp).

### 3.2.8 Ingame - Minikarte anzeigen

**Zuteilung** Diese User Story wurde auf 8 Storypoints geschätzt und an Georg Siebert zugeteilt.

**Ziel** Es sollte eine Minikarte implementiert werden, die über einen Button ein- und ausblendbar werden konnte.

**Story** Karli befindet sich bereits in einem Spiel. In der rechten oberen Ecke sieht Karli die Minikarte, welche neben dem Terrain auch alle Einheiten anzeigt. Karli drückt auf den Button zum Ausblenden der Minikarte. Die Minikarte wird ausgeblendet.

### 3.2.9 Ingame - Chatintegration

**Zuteilung** Diese User Story wurde auf 13 Storypoints geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Es sollte einen Button zum Ein- und Ausblenden des Chats implementiert werden.

**Story** Karli befindet sich bereits in einem Spiel. Karli blendet den Chat durch Betätigen des zugehörigen Buttons ein (siehe Mockup). Karli schreibt eine Nachricht an die anderen Spieler. Karli kann die Antworten der anderen Spieler lesen.

### 3.2.10 Ingame - Game Over

**Zuteilung** Diese User Story wurde auf 2 Storypoints geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Wenn der Nutzer verloren hatte, sollte ein Overlay angezeigt werden. Auf diesem konnte er in die Lobby wechseln oder dem Spiel zuschauen.

**Story** Karli hat keine Einheiten mehr und verliert das Spiel. Karli sieht, wie sich Karlis Playercard schwarz färbt (Name und Icon). Karli kann auswählen, ob in die Lobby gewechselt oder ob dem Spiel als Spectator beigetreten wird.

### 3.2.11 Ingame - Game Won

**Zuteilung** Diese User Story wurde auf 2 Storypoints geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Wenn der Nutzer gewann, sollte ein Overlay angezeigt werden, das ihn in die Lobby wechseln ließt.

**Story** Karli hat als einziger noch Einheiten und gewinnt das Spiel. Karli wechselt in die Lobby.

### 3.2.12 Lobby - Spectating

**Zuteilung** Diese User Story wurde auf 5 Storypoints geschätzt und an Juri Lozowoj zugeteilt.

**Ziel** Es sollte ein Button hinzugefügt werden, der einen Spielbeitritt als Beobachter ermöglichte.

**Story** Karli hat sich bereits eingeloggt und sieht die Lobby. Karli sieht die Spielliste und möchte dieses mal nicht selbst spielen. Karli drückt auf den Beobachter-Button und wechselt nun zum Spielfeld.

### 3.2.13 Ingame - Spectating

**Zuteilung** Diese User Story wurde auf 13 Storypoints geschätzt und an Juri Lozowoj zugeteilt.

**Ziel** Das Spiel sollte so angepasst werden, dass ein Zuschauer weder mit dem Spielfeld interagieren, noch Chatnachrichten schreiben konnte.

**Story** Karli sieht bereits das Spielfeld. Karli kann weder Aktionen auf dem Spielfeld ausführen, noch im Chat mitschreiben. Karli merkt aber, dass alles andere funktioniert, holt Popcorn und genießt das Spiel.

### 3.2.14 Ingame - Spectator Over

**Zuteilung** Diese User Story wurde auf 2 Storypoints geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Wenn das Spiel vorbei war, sollte dem Nutzer ein Overlay angezeigt werden, durch das er in die Lobby wechselte.

**Story** Karli sieht, dass Bob als einziger noch Einheiten hat und das Spiel endet. Karli wechselt in die Lobby.

### 3.2.15 Ingame - Einheiteninformationen

**Zuteilung** Diese User Story wurde auf 8 Storypoints geschätzt und an Georg Siebert zugeteilt.

**Ziel** Bewegte der Nutzer die Maus über eine Einheit, sollten alle verfügbaren Informationen dieser Einheit in der Sidebar angezeigt werden.

**Story** Karli befindet sich bereits in einem Spiel. Karli fährt mit seinem Mouse-Courser über eine Einheit. Es werden Karli alle wichtigen Informationen über die Einheit in der Sidebar angezeigt (siehe MockUps).

### 3.2.16 Ingame - Musiksteuerung

**Zuteilung** Diese User Story wurde auf 3 Storypoints geschätzt und an Omar Sood zugeteilt.

**Ziel** Es sollte ein Musik Button hinzugefügt werden, mit dem sich die Musik genauso bedienen ließ wie in den restlichen Szenen.

**Story** Karli befindet sich bereits in einem Spiel. Karli möchte die Musik ausschalten. Karli klickt auf den dafür vorgesehenen Button (siehe Mockup). Die Musik wird ausgeschaltet.

## 3.3 Tasks

Die Tasks wurden vom Scrum Master erstellt und sind nicht aus Nutzersicht beschreibbar, da sie nur technische Aufgaben waren. Aus diesem Grund unterschieden sich Tasks nur in einer Sache von User Stories. Sie hatten keine Story.

### 3.3.1 Servernachrichten

**Zuteilung** Diese Task wurde auf 8 Zeitstunden geschätzt und an Omar Sood zugeteilt.

**Ziel** Der Entwickler sollte alle restlichen Servernachrichten abfangen, die in keiner User Story berücksichtig wurden. Dazu gehörten unter anderem Änderungen des Datenmodells, welche durch Aktionen der Gegenspieler anfielen.

### 3.3.2 Einheiteninformationen im Datenmodell

**Zuteilung** Diese Task wurde auf 5 Zeitstunden geschätzt und an Tobias Klipp zugeteilt.

**Ziel** Der Entwickler sollte die Enums UnitType und UnitTypeInfo zu einer Enum zusammenfügen und musste darauf achten, dass alle Funktionalitäten erhalten blieben.

## 3.4 User Stories im Backlog

Weitere Stories (nur optionale Aufgaben) wurden im Backlog abgelegt. Da die Ziele für diesen Sprint schon ziemlich umfangreich waren, blieben alle diese Stories den kompletten Sprint im Backlog und waren niemandem zugeteilt und wurden auch nicht geschätzt.

### 3.4.1 Optional - Lobby - Anzeigen der Einheiten einer Armee

**Zuteilung** Diese User Story wurde bereits im Release II auf 5 Storypoints geschätzt und war in diesem Sprint niemandem zugeteilt.

**Ziel** Der Entwickler sollte ein Overlay implementieren, dass die Einheiten einer nicht ausgewählten Armee anzeigen, wenn man über das Icon hoverte.

**Story** Albert befindet sich im ArmyBuilder. Es sind Armeen vorhanden. Albert hovert über eine Armee. Es erscheint ein Overlay, in welchem die Einheiten der Armeen sehen kann.

### 3.4.2 Optional - Ingame - Auf gegnerischen Aktionen scrollen

**Ziel** Der Entwickler sollte ein automatisches Springen auf der Karte implementieren, wenn ein Gegner eine Aktion durchführt.

**Story** Karli ist nicht an der Reihe und schaut den Gegnern zu. Immer wenn ein Gegner eine Aktion durchführt (Bewegen bzw. Angreifen einer Einheit), sieht Karli, wie sich der Bildschirm auf die Aktion zentriert.

### 3.4.3 Optional - Waiting Room - Minigame

**Ziel** Der Entwickler sollte ein Minispiel implementieren.

**Story** Karli befindet sich einem Spiel beigetreten und befindet sich im Waiting Room. Karli hat Langeweile, da noch nicht alle ready sind. Karli spielt so lange Tic Tac Toe, bis das Spiel startet.

### 3.5 Zeitübersicht

Diese Übersicht zeigt alle Stories, welche im Rahmen der Ziele bearbeitet werden sollten. Wie zu erkennen ist wurden nicht alle Stories abgeschlossen. Es wurden jedoch schon umfangreichere Grundlagen für Features des sechsten Sprints gelegt.

User Story	Soll Zeit	Ist Zeit	Noch Zeit	Entwickler
Waiting Room - Spielbeitritt anpassen	5 h	13 h 9 min	-	Omar Sood
Ingame - Spielfeld	13 h	8 h 22 min	-	Georg Siebert
Ingame - Phase beenden	3 h	15 h 20 min	-	Juri Lozowoj
Ingame - Einheit auswählen	13 h	8 h 46 min	4h 14 min	Juri Lozowoj
Ingame - Einheit bewegen	13 h	11 h 19 min	35 min	Omar Sood
Ingame - Einheit angreifen	5 h	-	5 h	Omar Sood
Ingame - Spielstatus anzeigen	8 h	18 h 45 min	-	Tobias Klipp
Ingame - Minikarte anzeigen	8 h	8 h 25 min	4 h	Georg Siebert
Ingame - Chatintegration	13 h	-	13 h	Tobias Klipp
Ingame - Game Over	2 h	-	2 h	Tobias Klipp
Ingame - Game Won	2 h	-	2 h	Tobias Klipp

Tabelle 1: Zeitübersicht fünfter Sprint

### 3.6 Analyse

Der fünfte Sprint endete am 21.7.2019. Das Team schaffte 29 der 116 geschätzten Storypoints.

#### 3.6.1 Burndown

Wie im Burndown-Diagramm zu erkennen ist, wurden diesen Sprint nur vier Stories mit einem Umfang von 29 Storypoints abgeschlossen (siehe Kapitel 3.6.2). Wie Tabelle 1 zeigt, beträgt die Arbeitszeit trotzdem 84 Stunden. Die Hauptursache ist die Unterschätzung der Aufgabenumfänge, wodurch mehrere begonnene Stories in diesem Sprint nicht mehr abgeschlossen werden konnten und andere deutlich mehr Zeit als erwartet in Anspruch nahmen. Zudem führte die in diesem Release höhere Anzahl an blockierenden und relativen Tasks zu einer insgesamt langsameren Entwicklung, da sich die Entwickler häufig miteinander absprechen mussten.

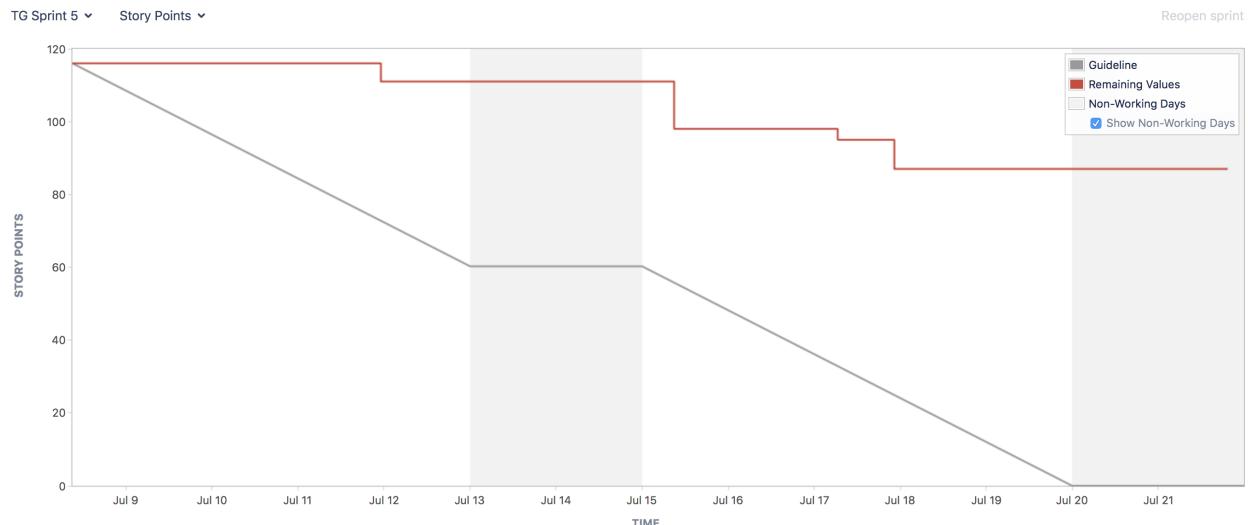


Abbildung 37: Sprint V: Burndown Diagramm

### 3.6.2 Abgeschlossene Stories

Es wurden bisher vier Stories abgeschlossen (siehe Abbildung 38). Dies ist weniger als erwartet, aber aufgrund komplexer Aufgaben nicht zu vermeiden. Im Nachhinein wird es nun eine genauere Beschreibung der abgeschlossenen Stories geben.

P	T	Key	Summary	Assignee	Status
✗	⌚	TG-255	Waiting Room - Spielbeitritt anpassen	Omar <goatfryed> ...	DONE
✗	⌚	TG-270	Ingame - Spielfeld	Georg Siebert	DONE
✗	⌚	TG-273	Ingame - Phase beenden	Juri Lozowoj	DONE
✗	⌚	TG-256	Ingame - Spielstatus anzeigen	Tobias Klipp	DONE

Abbildung 38: Sprint V: Abgeschlossene Stories

#### 3.6.2.1 Waiting Room - Spielbeitritt anpassen

**Verlauf** TODO: Schreiben

**Ergebnis** TODO: Schreiben

#### 3.6.2.2 Ingame - Spielfeld

**Verlauf** TODO: Schreiben

**Ergebnis** TODO: Schreiben

#### 3.6.2.3 Ingame - Phase beenden

**Verlauf** TODO: Schreiben

**Ergebnis** TODO: Schreiben

### 3.6.2.4 Ingame - Spielstatus anzeigen

**Verlauf** TODO: Schreiben

**Ergebnis** TODO: Schreiben

## 3.6.3 Angefangene Stories

Wie bereits in der Analyse des Burndowns erwähnt, wurden drei bereits begonnene Stories zum Ende des Sprints noch nicht abgeschlossen. Im kommenden Abschnitt wird es einen Überblick geben, wie weit diese bearbeitet wurden.

### 3.6.3.1 Ingame - Einheit auswählen

**Verlauf** TODO: Schreiben

**Stand** TODO: Schreiben

### 3.6.3.2 Ingame - Einheit bewegen

**Verlauf** TODO: Schreiben

**Stand** TODO: Schreiben

### 3.6.3.3 Ingame - Minikarte anzeigen

**Verlauf** TODO: Schreiben

**Stand** TODO: Schreiben

## 3.6.4 Nicht abgeschlossene Stories/Tasks

Des Weiteren wurden neben den drei bereits angefangenen Stories noch weitere zwölf Stories beziehungsweise Tasks aufgrund von Zeitgründen nicht angefangen und abgeschlossen (siehe Abbildung 39). Diese Aufgaben sind nicht optional und sollten von unserem Team im nächsten Sprint bewältigt werden.

P	T	Key	Summary	Assignee	Status
~	BUG	TG-276	Ingame - Einheit auswählen	Juri Lozowoj	TO DO
~	BUG	TG-281	Ingame - Einheit bewegen	Omar <goatfryed> ...	TO DO
~	BUG	TG-282	Ingame - Einheit angreifen	Omar <goatfryed> ...	TO DO
~	BUG	TG-311	Einheiteninformationen im Datenmodell	Tobias Klipp	TO DO
~	BUG	TG-263	Ingame - Minimap anzeigen	Georg Siebert	TO DO
~	BUG	TG-267	Ingame - Chatintegration	Tobias Klipp	TO DO
~	BUG	TG-274	Ingame - Kontextmenü-Framework	Georg Siebert	TO DO
~	BUG	TG-283	Ingame - Game Over	Tobias Klipp	TO DO
~	BUG	TG-285	Ingame - Game Won	Tobias Klipp	TO DO
~	BUG	TG-286	Ingame - Spectator Over	Tobias Klipp	TO DO
~	BUG	TG-287	Lobby - Spectating	Juri Lozowoj	TO DO
~	BUG	TG-288	Ingame - Spectating	Juri Lozowoj	TO DO
~	BUG	TG-289	Servernachrichten	Omar <goatfryed> ...	TO DO
~	BUG	TG-266	Ingame - Musiksteuerung	Omar <goatfryed> ...	TO DO
~	BUG	TG-275	Ingame - Einheiteninformationen	Georg Siebert	TO DO

Abbildung 39: Sprint V: Nicht Abgeschlossene Stories/Tasks

### 3.6.5 Entfernte Tasks

Am Ende des fünften Sprints wurden der Task „Einheiteninformationen im Datenmodell“ aus dem Sprint in das Backlog verschoben, da er nicht Teil der Mindestanforderung ist.

### 3.6.6 Fazit

Die Ziele des Sprints wurden verfehlt. Um die Mindestanforderungen zu erreichen musste das Entwicklerteam seine Produktivität im sechsten Sprint deutlich steigern. Da die C0 Abdeckung (siehe Abbildung 40) zum Ende des fünften Sprints bereits 79% betrug, gab es bei der Qualitätssicherung keine Defizite.

96% classes, 79% lines covered in package 'de.uniks.se19.team_g'			
Element	Class, %	Method, %	Line, %
project_rbsg	96% (410/423)	81% (1820/2235)	79% (8878/11197)

Abbildung 40: Sprint V: C0 Testabdeckung (Siehe Line, %)

## 4 Sprint VI

Der sechste Sprint erstreckte sich über dem Zeitraum vom 22.7.2019 bis zum 4.8.2019. Er umfasst 87 Storypoints.

### 4.1 Sprintziel

Es sollte nach dem sechsten Sprint möglich sein:

- Alle Features aus dem Sprint V (siehe Kapitel 3.1) zu verwenden
- Einen Ingame Chat zu benutzen
- Verschiedene Overlays beim Spielende zu sehen
- Einem Spiel als Beobachter beizutreten
- Einheiteninformationen in einer Sidebar einzusehen
- Die Musik auch im Spiel ein- und auszuschalten

Für den Fall, dass die Mindestanforderungen vor Ende des Sprints abgeschlossen werden sollten, soll eine Sidebar zum Spielfeld hinzugefügt werden. Diese soll es ermöglichen Aktionen, wie Bewegen oder Angreifen, zu bestätigen und abzubrechen. Weitere optionale Features liegen im Backlog vor (siehe Kapitel 4.6). Falls das Team die Umsetzung der oben genannten Features schafft sind die Mindestanforderungen jedoch bereits erfüllt.

### 4.2 Geänderte User Stories vom Sprint V

Die folgenden Stories des fünften Sprints wurden aktualisiert, da sich ihre Mockups geändert haben.

#### 4.2.1 Ingame - Minikarte anzeigen

**Zuteilung** Diese User Story wurde auf 8 Storypoints geschätzt und an Georg Siebert zugewiesen.

**Geändertes Ziel** Es soll eine Minikarte existieren. Über die Minikarte soll es möglich sein den Sichtbereich zu wechseln.

**Geänderte Story** Karli befindet sich bereits in einem Spiel. In der rechten unteren Ecke sieht Karli die Minikarte, welche neben dem Terrain auch alle Einheiten anzeigt. Karli drückt auf eine Stelle in der Minikarte. Der Sichtbereich ändert sich zu dieser Stelle.

#### 4.2.2 Ingame - Einheiteninformationen

**Zuteilung** Diese User Story wurde auf 8 Storypoints geschätzt und an Georg Siebert zugewiesen.

**Geändertes Ziel** Bewegt der Nutzer die Maus über eine Einheit sollen alle verfügbaren Informationen dieser Einheit angezeigt werden.

**Geänderte Story** Karli befindet sich bereits in einem Spiel. Karli fährt mit seinem Mouse-Courser über eine Einheit. Es werden alle wichtigen Informationen über die Einheit in der Sidebar angezeigt (siehe MockUps).

## 4.3 Übernommene User Stories aus Sprint V

Weiterhin gab es auch User Stories aus dem fünften Sprint, welche noch nicht abgeschlossen wurden. Diese wurden nun in den sechsten Sprint übernommen. Es folgt eine Liste dieser Stories:

- Ingame - Einheit auswählen (angefangen in Sprint V)
- Ingame - Einheit bewegen (angefangen in Sprint V)
- Ingame - Einheit angreifen
- Ingame - Spielstatus anzeigen (angefangen in Sprint V)
- Ingame - Minikarte anzeigen (angefangen in Sprint V)
- Ingame - Chatintegration
- Ingame - Game Over
- Ingame - Game Won
- Lobby - Spectating
- Ingame - Spectating
- Ingame - Spectator Over
- Ingame - Einheiteninformationen
- Ingame - Musiksteuerung

Eine genauere Beschreibung zu jeder Story befindet sich im Kapitel 3.2.

## 4.4 Geänderte Tasks vom Sprint V

Die folgenden Tasks des fünften Sprints wurden aktualisiert, da sich ihre Zuteilung änderte.

### 4.4.1 Einheiteninformationen im Datenmodell

**Geänderte Zuteilung** Diese Task wurde auf 5 Zeitstunden geschätzt und an Omar Sood zugewiesen.

**Ziel** Der Entwickler soll die Enums `UnitType` und `UnitTypeInfo` zu einer `Enum` zusammenfügen und darauf achten, dass alle Funktionalitäten erhalten bleiben.

## 4.5 Übernommene Tasks aus Sprint V

Weiterhin gab es auch Tasks aus dem fünften Sprint, welche noch nicht abgeschlossen wurden. Diese wurden nun in den sechsten Sprint übernommen. Es folgt eine Liste dieser Tasks:

- Servernachrichten

Eine genauere Beschreibung zu jeder Task befindet sich im Kapitel 3.3.

## 4.6 User Stories und Tasks im Backlog

**Ziel** TODO: Intro schreiben

### 4.6.1 User Stories aus Sprint V

- Optional - Lobby - Anzeigen der Einheiten einer Armee
- Optional - Ingame - Phase automatisch beenden
- Optional - Ingame - Auf gegnerischen Aktionen scrollen
- Optional - Ingame - Einheiten bewegen - Hover
- Optional - Waiting Room - Minigame

### 4.6.2 Tasks aus Sprint V

- Einheiteninformationen im Datenmodell

### 4.6.3 Ingame - Einheit bewegen bestätigen

**Ziel** TODO: Ziel beschreiben. Hier stand etwas ganz anderes

**Story** Karli hat bereits eine Einheit ausgewählt und auf das Feld geklickt, auf das Karli die Einheit bewegen möchte. Ein Icon erscheint auf diesem Feld (siehe MockUps). In der Sidebar kann Karli aussuchen, ob die Bewegung ausgeführt werden soll. Karli klickt „Move“. Die Einheit wird bewegt.

### 4.6.4 Ingame - Einheit angreifen bestätigen

**Ziel** TODO: Ziel beschreiben. Hier stand etwas ganz anderes

**Story** Karli hat bereits eine Einheit ausgewählt und auf die gegnerische Einheit geklickt, welche Karli angreifen möchte. Ein Icon erscheint auf diesem Feld (siehe MockUps). In der Sidebar kann Karli aussuchen, ob der Angriff ausgeführt werden soll. Karli klickt „Attack“. Die gegnerische Einheit wird angegriffen.

### 4.6.5 Ingame - Aktion abbrechen

**Ziel** TODO: Ziel beschreiben. Hier stand etwas ganz anderes

**Story** Karli hat bereits eine Einheit ausgewählt und das Feld geklickt, auf das Karli eine Einheit bewegen/angreifen möchte. Ein Icon erscheint auf diesem Feld (siehe MockUps). In der Sidebar kann Karli aussuchen, ob die Aktion ausgeführt werden soll. Karli klickt „Cancel“. Die Aktion wird abgebrochen und das Icon auf dem Feld ausgeblendet.

## 4.7 Zeitübersicht

TODO: Intro schreiben

User Story/Task	Soll Zeit	Ist Zeit	Noch Zeit	Entwickler
Ingame - Einheit auswählen	13 h	8 h 46 min	4h 14 min	Juri Lozowoj
Ingame - Einheit bewegen	13 h	11 h 19 min	35 min	Omar Sood
Ingame - Einheit angreifen	5 h	-	5 h	Omar Sood
Ingame - Spielstatus anzeigen	8 h	24 h	-	Tobias Klipp
Ingame - Minikarte anzeigen	8 h	8 h 25 min	4 h	Georg Siebert
Ingame - Chatintegration	13 h	-	13 h	Tobias Klipp
Ingame - Game Over	2 h	-	2 h	Tobias Klipp
Ingame - Game Won	2 h	-	2 h	Tobias Klipp
Lobby - Spectating	5 h	-	5 h	Juri Lozowoj
Ingame - Spectating	13 h	-	13 h	Juri Lozowoj
Ingame - Spectator Over	2 h	-	2 h	Tobias Klipp
Ingame - Einheiteninformationen	8 h	-	8 h	Georg Siebert
Ingame - Musiksteuerung	3 h	-	3 h	Omar Sood
Servernachrichten	8 h	-	8 h	Omar Sood

Tabelle 2: Zeitübersicht sechster Sprint

## 4.8 Analyse

TODO: Intro schreiben

### 4.8.1 Burndown

TODO: Schreiben

### 4.8.2 Ausreißer

TODO: Schreiben

### 4.8.3 Abgeschlossen

TODO: Schreiben

### 4.8.4 Angefangen

TODO: Schreiben

### 4.8.5 Nicht abgeschlossen

TODO: Schreiben

#### **4.8.6 Entfernt**

TODO: Schreiben

#### **4.8.7 Fazit**

TODO: Schreiben

## 5 Abschluss Release III

TODO: Intro schreiben

### 5.1 Neue MockUps im Verlauf des Releases III

TODO: Schreiben

### 5.2 Vergleich MockUps mit aktueller Implementation

TODO: Schreiben

## 6 Quellenangaben

### Abbildungsverzeichnis

1	Release II: Login Szene . . . . .	7
2	Release II: Login Form . . . . .	7
3	Release II: Lobby Szene: Keine Armee ausgewählt . . . . .	8
4	Release II: Create Game . . . . .	8
5	Release II: Lobby Szene: Armee ausgewählt . . . . .	9
6	Release II: Lobby Szene: Sprache DE . . . . .	9
7	Release II: Army Manager Szene: Keine Einheit ausgewählt . . . . .	10
8	Release II: Army Manager Szene: Einheit ausgewählt . . . . .	10
9	Release II: Property Info . . . . .	11
10	Release II: Armee bearbeiten . . . . .	11
11	Release II: Army Manager Szene: Speicherbar . . . . .	11
12	Release II: Waiting Room Szene: 2 Spieler . . . . .	12
13	Release II: Waiting Room Szene: 4 Spieler . . . . .	12
14	Release II: Ingame Szene . . . . .	13
15	Release II: Ingame Szene: Zoom In . . . . .	14
16	Release II: Ingame Szene: Zoom Out . . . . .	14
17	Release II: Login Fehler . . . . .	15
18	Release II: Spielfehler . . . . .	15
19	Release II: Spiel verlassen . . . . .	15
20	Release II: Logout . . . . .	15
21	Release II: C0 Testabdeckung (Siehe Line, %) . . . . .	15
22	MockUp: Beobachtungsmodus Lobby . . . . .	16
23	MockUp: Nutzer nicht bereit . . . . .	17
24	MockUp: Nutzer bereit . . . . .	17
25	MockUp: Spielstart . . . . .	18
26	MockUp: Ingame . . . . .	19
27	MockUp: Phase 1: Auswählen . . . . .	19
28	MockUp: Phase 2: Auswählen . . . . .	20
29	MockUp: Phase 3: Auswählen . . . . .	20
30	MockUp: Bewegen bestätigen (Phase 1) . . . . .	21
31	MockUp: Bewegen bestätigen (Phase 3) . . . . .	22
32	MockUp: Angreifen bestätigen . . . . .	22
33	MockUp: Spiel verloren . . . . .	23
34	MockUp: Spiel gewonnen . . . . .	24
35	MockUp: Beobachtungsmodus: Spiel vorbei . . . . .	24
36	MockUp: Phase beenden . . . . .	25
37	Sprint V: Burndown Diagramm . . . . .	33
38	Sprint V: Abgeschlossene Stories . . . . .	33
39	Sprint V: Nicht Abgeschlossene Stories/Tasks . . . . .	35
40	Sprint V: C0 Testabdeckung (Siehe Line, %) . . . . .	35

### Tabellenverzeichnis

1	Zeitübersicht fünfter Sprint . . . . .	32
2	Zeitübersicht sechster Sprint . . . . .	40