

# Dear Diary

## MILESTONE 1

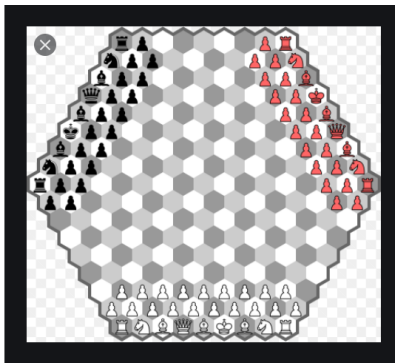
04.03.21

Gruppenfindung

Rundenbasiertes Spiel

08.03.21

Schach : Multiplayer



Auswahl:

- timer: Automatisch oder selber einstellen
- und non-timer
- Farbwahl

Kommunikation:

- Privat, Alle, temporär Allianz - Gründung

Anzahl Spieler:

min. 2

-> zuerst 3 - Spieler (der Rest anpassen)

Regel:

- <https://greenchess.net/rules.php?v=three-hexagonal>
- <https://www.chess.com/forum/view/chess-variants/88-hexagons-chess-new-hexagonal-chess-variant>
- [https://en.wikipedia.org/wiki/Hexagonal\\_chess](https://en.wikipedia.org/wiki/Hexagonal_chess)  
~König?!  
Queen (doch nit broken Edit: teilweise)

Brett:

regelmässiges 6 -Eck

Brettgrösse:  $n * (2n - 1)$  ->  $n = 9$

153 Felder bei 3 Spieler

Bewegung:

- Figur tippen
- Feldauswahl
- Knöpfe?

Code:

- Methode: schlag, move
- GUI
- Feldbezeichnung

Zusatz:

- Preis
- XP
- Teams: nur gerade Anzahl  
gleiche Farbe  
2 moves pro runde

12.03.2021

Chat:

- Soll wichtig sein.

To Do - Liste nötig.

Spielverlauf:

- mögliche Replays

Präsentation:

- infos zu server/client
- mock ups
- usw. steht auf PPP

13.03.2021

Präsentationsvorbereitung (siehe PPP und word doc)

15.03.2021

Für PPP fehlt noch: (besserer Titel ;))

- Project-timeline in der Präsentation

- Beispiel - Protokoll (Server/Client):

  - Client zu Server: Mov A0 -> B1 Player 1

  - Server antwort

- allg. Schachregeln :

  - Rochade

  - en passant

PPP vor Do als PDF

Gant/Excel: Milestone Daten

17.03.2021

“letzter Schliff” der Präsentationsfolie. Sie Folie des Netzwerkprotokolls und Software Requirements.

präsentation einmal "geübt" mit Stoppuhr, usw.

## MILESTONE 2

22.03.2021

Orientierung der aktuellen Situation:

.gitignore war noch nicht installiert

Netzwerkprotokoll beinahe fertig (Raffael)

bereits start an den "switch-cases" für implementierung des Netzwerkprotokolls

Server / Client vor Milestone 1 funktioniert gar nicht. -> aktuell neue Version erstellt.

gemeinsames Programmieren: (Jonas' PC)

Version des Server - Clients basierend aus Vorlesung

- Server erhält Eingabe und kann Client zurückgeben.  
Bsp. Client: hallo -> Server erhält hallo -> Client bekommt hallo zurück
- Rückgabe zum Client funktioniert momentan nur auf einem Client
- Erstellung von .gitignore  
scheint noch nicht zu funktionieren?!
- Login für mehrere Clients funktioniert. (noch nicht "meaningful")  
Ausser Client 1 agieren die anderen noch nicht gleich

ToDo:

einige Milestones....

Username

Login/out

Ping/Pong

Problembehebung (Client to Client)

funktioniert .gitignore nun?

nächstes Treffen: Mi 24.03.2021 um 16:00 Uhr

24.03.2021

Aktuell:

Alle Clients interagieren mit Server.

Aber Rückgabe der Nachrichten vom Server gelangen nur an Client 1.

-> Problembehebung nötig.

Implementierung des Netzwerkprotokolls ~Anwendungsbereit.

Ping/Pong sollte funktionieren (noch nicht getestet)

gemeinsames Programmieren: (Jonas' PC)

- Versuch der Problembehebung (siehe Aktuell) -> noch nicht geschafft

- anfängliche Nutzung des Netzwerkprotokolls im Server / Client  
einige Methoden hinzugefügt, Typos korrigiert -> Netzwerkprotokoll (net.java)  
kompiliert  
Ausgabe von Txt noch nicht optimal  
(Netzwerkprotokoll human-readable ???)
- Start der Codierung der Usernames (in ServerThread)  
Entscheidung zur Verwendung der LinkedList. (Speicherung und Kontrolle)  
Im Server wird angezeigt was geschrieben wird (jedoch noch falsche Usernames)  
Probleme:  
Clients mit ihrem Username speichern resp. Abrufen immer nur die letzte  
eingeloggte Person stimmt.  
Format der Rückgabe noch "unschön": ein Enter zu viel

Todo:

Client to Client Chat immer noch nicht behoben  
scheinbare ClassNotFoundException (manchmal vorhanden und manchmal nicht!!!)  
Probleme der Usernames beheben.  
(Netzwerkprotokoll wirklich human-readable?)  
nach Client - Behebung unbedingt Ping-Pong testen

25.03.2021

Aktuell: siehe 24.03.2021

gemeinsames Programmieren: (Jonas' PC)

- Username wird gleich gespeichert.  
Kann nun zum Client zugeordnet werden.  
Im Server sowie im Client zeigt es an, wer was schreibt.
- Client to Client Chat testen über Hamachi.  
Probleme dieser Woche wurden endlich behoben.  
-> Chat funktioniert nun.
- Username-Vorschlag nach Login existiert nun. (Name auf jeweiligen Computer)
- Änderung des Usernames zu Beginn noch möglich. Kontrolle ob Name bereits existiert funktioniert inkl. Fehlermeldung.
- nach Logout wird der entsprechende username aus der LinkedList entfernt.  
Wiederverwendung des "ausgeloggeden" User möglich.

erledigt

beinahe fertig

naja

kann nicht einschätzen, bitte jemand anderes Farbe ändern

nicht begonnen

Milestones:

gitignore:

scheint zu funktionieren, alte Klassen (vor gitignore) noch löschen, .idea hinzufügen

source code documented:

ja

protocol encode...:

noch nicht alle methoden vollständig

change nickname:

nur zu Beginn

client to client chat:

funktioniert

dear Diary:

heute aktualisiert

Login:

username vorschlag zu beginn, und username speicherung

-> empfängt aber schon nachrichten bevor username-Eingabe

Logout:

funktioniert gut

Ping:

zu testen

Pong:

zu testen

network protocol in source code:

ja

network protocol documented:

human readable?

use of protocol:

wird verwendet

QA concept:

noch nicht begonnen!!!

unique nickname:

nein noch nicht

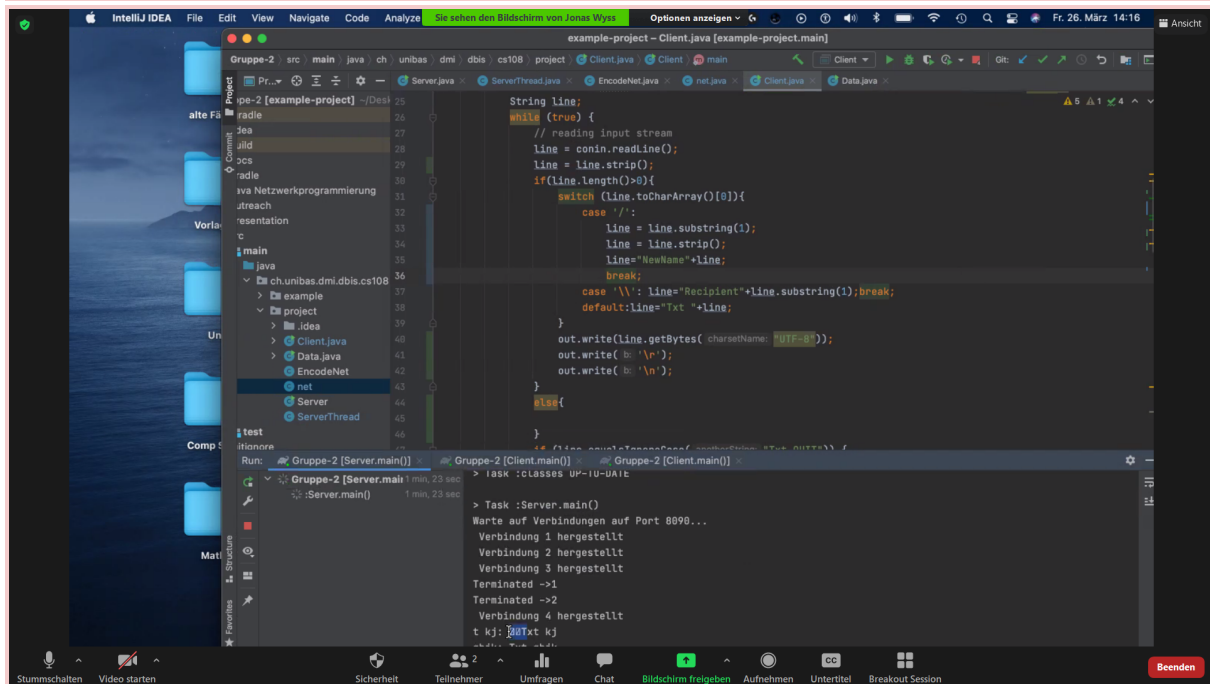
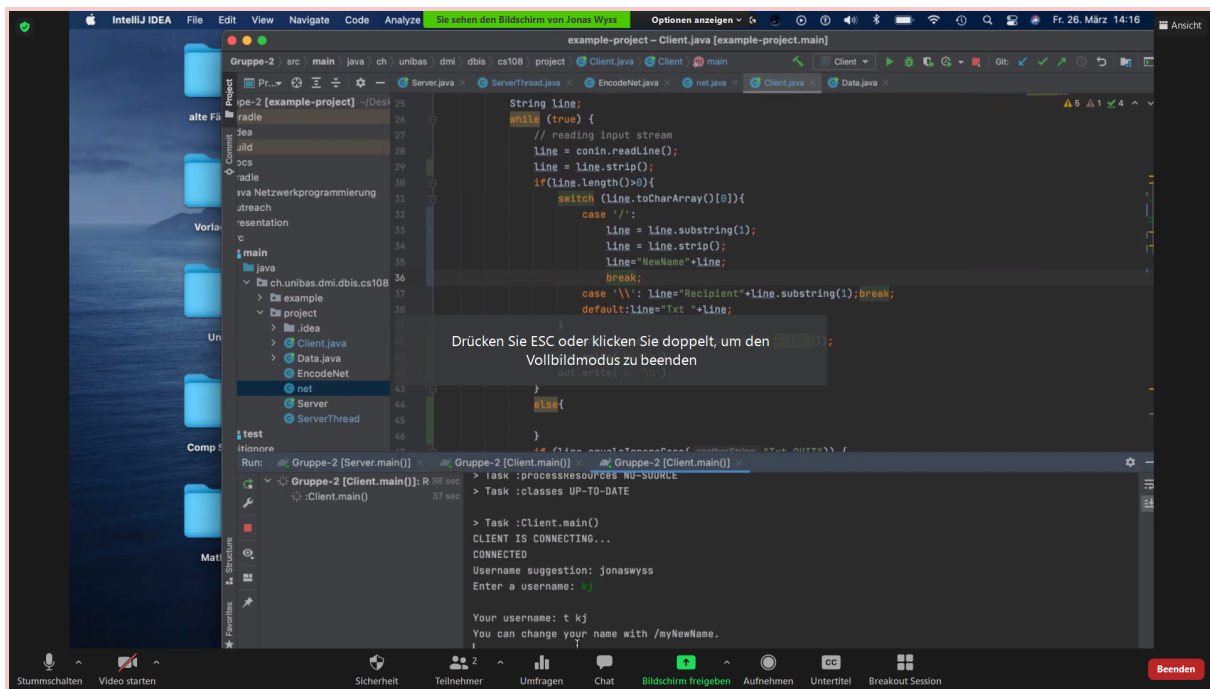
whoami:

funktioniert

(bonus executable jar: wird noch nicht beachtet)

26.03.2021

Problem:



-> Problem gelöst (waren “\r\n”)

gemeinsames Programmieren: (Jonas’ PC)

!!!Netzwerkprotokoll wurde noch nicht korrekt verwendet. Änderungen werden vorgenommen.

QA concept erstellt (noch auf repository stellen)

28.03.2021

ping pong funktioniert einfach nicht (für milestone 2 aufgegeben) (sehr lange versucht gemeinsam zu programmieren)

alles mit username und chat sind funktionstüchtig

.gitignore: alle möglichen files vom repository entfernt

QA concept auf repository gestellt

## MILESTONE 3

29.03.2021

Aktualisierung des Gant-Projekts:

Sascha & (main) Raffael: Fortsetzung Netzwerkprogrammierung

Sascha: erstellung javadoc im Netzwerk so gut wie möglich

Jonas: Game Logic

Edi: Chat GUI

Allg.:

Erstellung von Dokumenten (z.B. QA, Präsentation) erst nach Vollendung der Programme.

08.04.2021

Aktueller Stand der jeweiligen Gruppen erklärt:

“Alle” müssen noch an ihrem Aufgabenbereich weiterarbeiten

- Netzwerk:

programm bleibt im in.read() stecken (siehe net\_thread) -> Andere Lösung?!  
frage in Übungsstunde

Format für jar Aufruf wurde implementiert (Voraussetzung erfüllt):  
z.B. java -jar Jar.java server 8090

- Game Logic:

Springer noch programmieren

dann am Manual arbeiten

- Chat GUI:

implementation in Netzwerk (testen nur möglich, wenn Netzwerk vorhanden ist)

in dieser Zeit, jar file bereitstellen

12.04.2021

Problem:

Netzwerk funktioniert immer noch nicht.

read() Methode ist eigentlich die einzige Methode -> sollte theoretisch auch funktionieren?!

-> Lösungsansatz: ein Thread zum Lesen und ein Thread zum Schreiben sollen im Server sowohl Client initialisiert werden.

Oder doch lieber die Version von ms2 verwenden? (noch unentschieden; Festlegung am Donnerstag)

Manual:

on progress

.jar File:

funktioniert

Chat implementation wegen fehlendem Netzwerk schwierig

Jonas schaut in QA-Theorie rein. Hauptaufgabe vorerst aber Manual.

Edi "lernt" Logger kennen, wenn möglich schon implementieren.

15.04.2021

Netzwerkprotokoll funktioniert immer noch nicht! resp. Idee mit separierten Threads war gut. Aber zu viele unentdeckte Fehler vorhanden.

-> Version von ms2 wird übernommen und an die neue angepasst.



=> gemeinsames Programmieren (Saschas PC): mit IntelliJ Code with me

Sicherstellen, dass alle nötigen Files vorhanden und aktualisiert sind:

gradlew, build, Chat, Data, Call -> jar ruft es auf

Initialisierung des Usernames funktioniert beinahe wieder.  
Doppeltespeicherung muss behoben werden.

Gant-Änderung: (auf morgen Fr 16.04.2021 oder zumindest am Freitag fertig)

Raffael: "fixed" Problem bei Initialisierung von Uname.

Jonas: Implementation Game Logic so dass gespielt werden kann (vorerst ohne Lobby)

Sascha & Edi : Chat implementieren mit Server

Diejenigen, die fertig sind und Zeit haben, können bei den anderen helfen. Verwendung von Code with me nicht schlecht: verhindert merging fehler (diese woche zu oft vorgekommen, dass files "verschwunden" waren)

Nachteil Code with me: pro User muss der Commit stattfinden sprich: commit message 1 ist wie immer, aber die anderen commit messages heissen: Changes by USER

=> wenn changes by user als commit message steht, in history der commit anschauen bei dem dies nicht mehr steht

Nächstes Meeting Samstag 17.04.2021

(Kurzes Feedback in Chat about aktuellen Stand am Freitag abend wäre optimal)

16.04.2021 (kein wirkliches Meeting, kam aber zu Interaktionen unter den Gruppen, teilweises gemeinsames Programmieren mit Code with me)

Überblick Milestone 3:

erledigt

beinahe fertig

ongoing

command missing

hat funktioniert, aber abhängig was jz passiert

nicht begonnen

- Source code documented (Sprich javadoc)

- Broadcast: Client -> Server -> all Clients inkl. command existent
- Build Script: `jar` und `javadoc`
- Cmd: z.B. `server <port>`
- working prototype of game in terminal or gui
- chat on gui
- command to list status of different games
- Game Logic: game is playable, fundamental mechanics
- Game State: on server
- Librarian: not JUnit or JavaFX
- Lounging: multiple Lobbies
- Lounging List: command to list players in lobbies
- Manual: How does the game work?
- Player List: command to list all players connected to Server
- Protocol Code: completely defined and in source code
- Protocol Document: up to date
- QA
- Playing working prototype
- Whisper: whisper chat betw. two clients

sonstiges: Coding Conventions unbedingt anpassen!!!

17.04.2021

komplettes gemeinsames programmieren über CodeWithMe.

Aktuell: Überblick Milestone 3: (grau ist grundsätzlich ein gutes Zeichen)

erledigt      beinahe fertig      ongoing

command missing

funktioniert bis jetzt, kommt darauf an, was weiter geschieht

nicht begonnen

- Source code documented (Sprich javadoc)
- Broadcast: Client -> Server -> all Clients inkl. command existent
- Build Script: `jar` und `javadoc`
- Cmd: z.B. `server <port>`
- working prototype of game in terminal or gui
- chat on gui
- command to list status of different games
- Game Logic: game is playable, fundamental mechanics
- Game State: on server
- Librarian: not JUnit or JavaFX
- Lounging: multiple Lobbies //todo fehlermeldung wenn chose Lobby gar nicht geht

- Lounging List: command to list players in lobbies
- Manual: How does the game work?
- Player List: command to list all players connected to Server
- Protocol Code: completely defined and in source code
- Protocol Document: up to date
- QA -> Jonas schaut hinein auf Sonntag 18.04.2021
- Playing working prototype
- Whisper: whisper chat betw. two clients

sonstiges: Coding Conventions unbedingt anpassen!!!

## MILESTONE 4

22.04.2021

Meeting, Verteilung der verbleibenden Aufgaben

Raffael: GUI Programm für Animation

Sascha: Start Menu GUI

Edi: Lobby GUI

Jonas: Game Logic, Spielfeld GUI

Sascha und Edi: Erstellung von UNIT Tests

26.04.2021

Aktueller Stand der jeweiligen Gruppen erklärt:

“Alle” müssen noch an ihrem Aufgabenbereich weiterarbeiten

Vorstellung der jeweilig programmierten GUIs.

einzelnen GUIs sollen bis Donnerstag in das Programm eingebaut werden.

verschiedenste Bugs besprochen und Behebungsmöglichkeiten diskutiert

29.04.2021

GUIs sind alle zusammengeführt

Figuren müssen ergänzt werden für die Spielfeld GUI

Überlegung für Schach und Schach Matt Implementierung

30.04.2021

Grosser Tag des Testing

sämtliche Funktionen des Spiels werden heute getestet

Schach und Schach Matt funktioniert noch nicht zuverlässig

→ Beschliessung Schach und Schach Matt wird für diesen Meilenstein nicht verwendet

Neues Spielziel wird definiert:

→ Der König kann wie jede normale Figur geschlagen werden und sobald dies erfolgt, hat die Person verloren und alle Figuren verschwinden vom Spielfeld

→ Bei dieser Variante ist Aufmerksamkeit sehr bedeutende. Durch einen unaufmerksamen Zug kann bereits das komplette Spiel verloren werden. Dadurch wird das Spiel sehr kurzlebig und schnell im Vergleich zum normalen, sehr gedanken intensiven und lang andauernden normalen Schach

Spielen mehrere Partien mit diesen Regeln

sämtliche Spielzüge werden durchprobiert

Die neue Gewinnmöglichkeit benötigt eine neue Methode, die das Spiel beendet.

Diese wird geschrieben.

Feststellung: die bereits programmierte Mausssteuerung ist unzuverlässig, es wird häufig eine falsche Figur ausgewählt. Deshalb wird umgestellt auf eine Button Steuerung. Dadurch ist das Spiel viel angenehmer zu spielen, da immer die richtigen Figuren gezogen werden. Die Zugauswahl ist viel zuverlässig damit.

Ein Problem bleibt, die ausgewählte Figur ist nicht sichtbar, deshalb ist nicht klar, ob man bereits eine Figur ausgewählt hat.

Es wird deshalb ein Rahmen erzeugt, der zeigt, welche Figur ausgewählt ist. Dadurch ist das Spiel viel benutzerfreundlicher und einfacher zu steuern.

Weiter wird programmiert, dass die aktuelle Zugfarbe oben angezeigt wird, damit man immer weiss, wer dran ist.

Sascha beginnt externes High Score Dokument zu erstellen und zu implementieren

08.05.2021

Highscore fertig

Javadoc hinzugefügt für alle programmierten Methoden

## MILESTONE 5

10.05.2021

Überblick Meilenstein 5 Planung:

soweit so gut / ongoing; unbedingt spätestens Samstag vor Abgabe kontrollieren

bereits zugeordnete Aufgabe

**deadlines: KEINE AUFSCHIEBUNG MÖGLICH** (wenn nichts steht **SO 23.05**, oder noch nicht bestimmt)

- Source Code

detaillierte Beschreibung einer Klasse mit Javadoc (oracle docs als beispiel)

UnitTests "normale" Kommentare (erklärung von Zügen usw.)

momentan: Klassen Field, Data "normal" kommentieren -> optimal vor "for" und "if" und je nach dem auch "else"

- Architecture: Have documentation outlining the overall program architecture

Was ist das?

- Archiving: Your repository contains an outreach/ folder with the specified content

Was ist das?

- Cmd still works

- Dear Diary: **Raffael**

- GUI

allg. "SCHÖNE UND EINHEITLICHE GUI"

Animierte Figuren: **Raffael** (soll nicht weiter fortgesetzt werden, wenn bis **auf FR 14.05.** nicht erledigt ist)

"It's your turn" nicht mehr im Titel setzen: wenn möglich im Field **Sascha**

"You are black" **Sascha**

evtl. Button zum Spiel verlassen nach schachmatt (anstatt Feld sofort verschwinden lassen)

beim Spiel: aufzeigen, welche Schritte die Figur laufen kann

Button für Manual resp. Anleitung in Field sowie Lobby **Sascha**

- GameLogic

wenn noch was fehlt? **Jonas**

- HighScore: richtiger Pfad setzen (damit jar extern funktioniert) **Sascha + Hilfe!**
- Login: many Clients can log on to server
- Manual

**WICHTIG: vor FR 21.05.2021**

modifizierte Schachregeln ergänzen (am besten Anleitung mit unserer GUI)  
**Jonas**

- Libraries in maven central

- Screenshot of game

TODO

- final version: Protocol code (in source code)

**SA 22.05.2021**

vor Abgabe unbedingt erledigen (unnötige Methoden löschen)

- final version: Protocol document

Raffael?

- QA report + advanced

**Edi**

- Use of at least 2 libraries actively (X JUnit, JavaFX)

wahrscheinlich, will Loris es aktiver haben,)

- Video of gameplay (20s - 2 min)

TODO

- Unit-Test (meaningful and green)

**Edi**

15.05.2021

Aktueller Stand der Planung: und wer, was macht

soweit so gut / ongoing; unbedingt spätestens Samstag vor Abgabe kontrollieren

bereits zugeordnete Aufgabe

**deadlines: KEINE AUFSCHIEBUNG MÖGLICH** (wenn nichts steht **SO 23.05**, oder noch nicht bestimmt)

- Source Code

detaillierte Beschreibung einer Klasse mit Javadoc (oracle docs als beispiel)

UnitTests "normale" Kommentare (erklärung von Zügen usw.)

momentan: Klassen Field, Data "normal" kommentieren -> optimal vor "for" und "if" und je nach dem auch "else"

- **Architecture:** Have documentation outlining the overall program architecture

Siehe EMail von Simon vom 14.05.2021

- **Archiving:** Your repository contains an outreach/ folder with the specified content

Siehe EMail von Simon vom 14.05.2021

- Cmd still works

- Dear Diary: **Raffael**

- GUI

allg. "SCHÖNE UND EINHEITLICHE GUI"

"It's your turn" nicht mehr im Titel setzen: wenn möglich im Field **Sascha**  
**erledigt**

"You are black" **Sascha** erledigt

evtl. Button zum Spiel verlassen nach schachmatt (anstatt Feld sofort verschwinden lassen) **wird wahrscheinlich nicht gemacht**

beim Spiel: aufzeigen, welche Schritte die Figur laufen kann **zu aufwändig, keine Zeit**

Animierte Figuren: **Raffael** (soll nicht weiter fortgesetzt werden, wenn bis **auf FR 14.05.** nicht erledigt ist) -> **wird nicht mehr fortgesetzt**

Button für Manual resp. Anleitung in Field sowie Lobby **Sascha**

- GameLogic

wenn noch was fehlt? **Jonas**

- HighScore: richtiger Pfad setzen (damit jar extern funktioniert) **Sascha + Hilfe!**
- Login: many Clients can log on to server
- Manual

**WICHTIG: vor FR 21.05.2021**

modifizierte Schachregeln ergänzen (am besten Anleitung mit unserer GUI)

**Jonas**

- Libraries in maven central

- Screenshot of game

TODO

- final version: Protocol code (in source code)

**SA 22.05.2021**

vor Abgabe unbedingt erledigen (unnötige Methoden löschen)

- final version: Protocol document

Raffael?

- QA report + advanced

**Edi**

- Use of at least 2 libraries actively (X JUnit, JavaFX)

wahrscheinlich, will Loris es aktiver haben,)

- Video of gameplay (20s - 2 min)



**FR 21.05.2021**

- Unit-Test (meaningful and green)

**Edi**

- Website: Raffael

**17.05.2021**

Besprechung mit Loris und Simon wegen Differenzen im Team. (alle)

Outcome: Projekt wird gemeinsam fortgesetzt und beendet, müssen noch genauer planen, wer, was macht.

Nach Besprechung hat uns Raffael verlassen...

Er wird vermutlich ein wenig den HTML-Code im outreach-Ordner machen und dann weiter an seiner Website arbeiten.

Jonas soll noch die Manual so gestalten, dass sie in die GUI implementiert werden kann.

Sascha arbeitet an Architecture. Und sobald Manual bereit ist, wird sie in die GUI implementiert.

Edi ist am QA-Report dran.

nächstes Meeting: DO 10:30, Sascha ruft alle ins Zoom-Meeting an.

**20.05.2021**

aktueller Stand:

Manual in GUI implementiert. Architecture fehlt noch ein Kapitel. QA-Report is on progress. Website is on progress.

Gemeinsames programmieren:

SocketException - Problem endlich gefunden und abgefangen (mit try catch block)

**22.05.2021**

Aufnahme des Video-Clips. Sascha wird alles noch bearbeiten und schneiden.

Gemeinsames Programmieren von "QUIT", so dass Spieler/innen das Spiel jederzeit verlassen können. Wenn in irgendeiner GUI (ausser Listen) das "Kreuzchen" geklickt wird, wird der/die Spieler/in ausgelogged.

Organisatorisches:

Raffael ist immer noch an der Website dran (keine Kapazität für anderweitiges)

Wer gerne eine Aussage über den Verlauf des Projekts auf der Website abgebildet haben möchte, soll bis 20.00 am Sonntag den Text an Raffael senden.

Edi ist an QA-Report dran (wird noch den Code am Sonntag rezyklieren und "verschönern")

Source Document, Architecture Final Version, Video Clip (Sascha)

Bilder resp. Screenshots hochladen in richtigen Ordner (Jonas)

Planung Präsentation:

jede Person ungefähr 2 min.

Präsentationscomputer Game: Edi

Präsentationscomputer Folie: Raffael

Logo (5)

schon drin

About A Game (10)

Jonas

About A Game advanced (10)

Meilenstein 1

Jonas

QA (15)

Edi

Technology (10) (Tools, Libraries, ...)

Meilenstein 3

Raffael

We are Smarter Now (20)

Sascha

Shall we Play A Game (70)

Game Skript

Moderator : Raffael

Server: Jonas

(Boni)

Free to Play (3): Game available for download (auf Start Folie)

Website (5): auf startFolie

Nächstes Meeting

Mittwoch 26.05.2021