



DLVC -Taverne

Rollenspielmanager „Die Legenden von Cysteron“

Vortrag 3. Iteration - Sommersemester 2015



Inhalt

1. Einleitung
2. Projektdetails
 1. Stand der letzten Iteration
 2. Rollenverteilung im Team
 3. Bearbeitete Stories
3. Fortschritt: Aktueller Stand
 1. Burndown chart
 2. Nicht Inhaltlich
 3. Inhaltlich
 4. Testabdeckung
4. Demo
5. Reflektionen
 1. Reflexionen des Arbeitsprozesses
 2. Vorschau auf letzte Iteration
 3. Fazit



Einleitung



Einleitung

➤ Was ist ein Charakter?

Charakterblatt

Name: *Monke Monke* *46: 85'*
 Sozialer Rang: *Große Groß*
 Voller Titel: *Groß von Therosanden* *Ki: 128/113/103/93/85*
 Rasse: *Windling* *128/120/112*
 Spezielle Fertigkeit: *ling MOcm*
 Gassenwissen
 Disziplin: *Elementarst*
 Anfangsfertigkeit: *Feuerbeherrschung*

Angst: *Drücken - Insekten*
 - *tiefer Höhlen - Gitterzorn*
 Attribute: *Verknüpfelt, Sanft, unantel,*
Herablassend, Blasphemisch, Feige,
Rivalisierend, Alterwürdig

Größe: *3 dm*
 Alter: *15 Jahre*
 Hautfarbe: *dunkel*
 Haarfarbe: *rostig braun*
 Augenfarbe: *grün*
 Talente: *Tausendsassa*

Bild Wappen

verbleibende Wertpunkte: *13*

Werte	Stärke	Geschick	Willenskraft	Wahrnehmung	Charisma	Zähigkeit	Energie
Kreis 1	20	19	13	14 +4	18	28	58
Kreis 2				18 +4			
Kreis 3				22			
Würfel	W12 +3	W12	W6	W10+3	W20+10	LP: 90+5	Ki: 105+3

Folgewerte:		Attributproben:		Sonstiges:	
Traglast:	48		W20 >		
Laufleistung:	33 +8 (-50)		W20 >		
Totaler Schaden:	3		W20 >		
Verletzungs-Wert:	32 -2		W		
Ohnmachts-Wert:	16		W		

Erfahrung:

	1	2	3	4	5	6	7	8	9	10	11	12
Level:	X	X	X	X	X	X	X	X	X	X	X	X
Benötigte Exp.:	100	150	225	330	500	750	1125	1650	2475	3700	5550	8325
Kreis:	1	X		2			3			4		
Faktor:	1			100			10.000			1.000.000		
Erfahrungspunkte:	5K			1247,5K								

+12.8K



Einleitung

- Was ist ein Charakter?
- Spieler Charakter

Charakterblatt

Ab: 857

Name: Menne Monke
Sozialer Rang: 6. beste Graf
Voller Titel: Graf
von Desosanden


Ki: 128/113/103/93/85
128/120/112

Rasse: Wendling
Spezielle Fertigkeit: 110cm
Gassenwissen
Disziplin: Elementarst
Anfangsfertigkeit: Feuerbeherrschung

Angst: Dracken - Insekten
tiefer Höhlen - Götterzorn
Attribute: Vorurteil, Sentimental,
Herablassend, Blasphemisch, Feig,
Rivalisierend, Alterwürdig

Größe: 3 cm
Alter: 15 Jahre
Hautfarbe: dunkel
Haarfarbe: rostig braun
Augenfarbe: grün
Tale: Tausendsassa

Bild/Wappen



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter

Handwritten character sheet for a game, titled "Tausendsassa".

Werte, Proben & Würfel:

Werte	Stärke	Geschick	Willenskraft	Wahrnehmung	Charisma	Zähigkeit	Energie
1	20	19	13	14	18	28	58
2				18	18		
3				22			

Würfel: W12 + 3, W12, W6, W10 + 3, W20 > 10, LP: 40 + 15, K: 105 + 20

verbleibende Wertpunkte: 13

Folgewerte:

Traglast:	47
Laufleistung:	33 + 8 (+90)
Totaler Schaden:	3
VerletzungsWert:	32 - 2
OhnmachtsWert:	16

Attributproben:

	W20 >
	W20 >
	W20 >
	W
	W

Sonstiges:

Erfahrung:

Level	1	2	3	4	5	6	7	8	9	10	11	12
Benötigte Exp.	100	150	225	350	500	750	1125	1650	2475	3700	5550	8325
Kreis:	1		X									
Faktor:	1			2			3			4		
Erfahrungspunkte:	5K			1247,5K								

Handwritten notes: "Tausendsassa", "Talente: Tausendsassa", "Werte, Proben & Würfel:", "Erfahrung:", "5K", "1247,5K", "412,8K".



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter



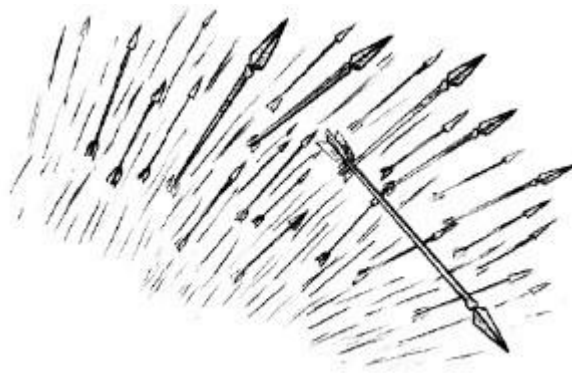
Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?



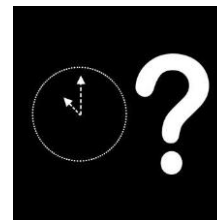
Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?
 - Gegnerrunde



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?
 - Gegnerrunde
 - Spielerrunde



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?
 - Gegnerrunde
 - Spielerrunde



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?
 - Gegnerrunde
 - Spielerrunde
- 2. Wie läuft ein Handel ab?



Einleitung

- Was ist ein Charakter?
 - Spieler Charakter
 - Nicht Spieler Charakter
- 1. Wie läuft ein Kampf ab?
 - Gegnerrunde
 - Spielerrunde
- 2. Wie läuft ein Handel ab?
 - Problem: Organisation



Projektdetails



Projektdetails: letzte Iteration

Spielgruppenmanager

- Erweiterung der Datenbank
- Bereitstellen aller Funktionen
- Programmweite Observer

Kampfsimulator: Gegnerrunde

- Spezifikation durch Kunden
- Design der GUI
- Bereitstellen der Grundfunktionen

Händler: Einfache Gegenstände

- Erweiterung der Datenbank
- Design der GUI
- Bereitstellen der Grundfunktionen



Projektdetails: Rollenverteilung

Kunde	Hans Klein
Teamleiter	Boris Prochnau
Team	Nooshin Naghavi Britta Heymann Andreas Kofer Boris Prochnau
Dank für: - Bilder - Grafikdesign	Sabine Weiß Lea Prochnau




Projektdetails: Stories

Stories sind
unverändert
geblieben


• Akzeptanztests gel.
nicht definierbar

Titel: Kampfsimulator (Gegnerrunde)

Nr. : 4

Beschreibung: 

Der Spielleiter bestreitet mit einer Spielergruppe ein Kampf gegen einer Gegnergruppe ohne Spezialfälle. Dabei sollte der Schaden und Trefferzone der Gegner sichtbar sein.

Akzeptanztest: 

- Trefferzone müssen sichtbar sein
- Beteiligung von vielen Spielern und Gegnern
- Lebenspunktverluste für Spieler anzeigen

Priorität der Story: 1

Geschätzte Zeit: 16 Paar-Std.

Begonnen am: 08.05.

Abgeschlossen am: 19.05.

Benötigte Zeit: 14.22 Paar-Std.



Projektdetails: Stories

Stories sind
unverändert
geblieben

- Akzeptanztests gel. nicht definierbar

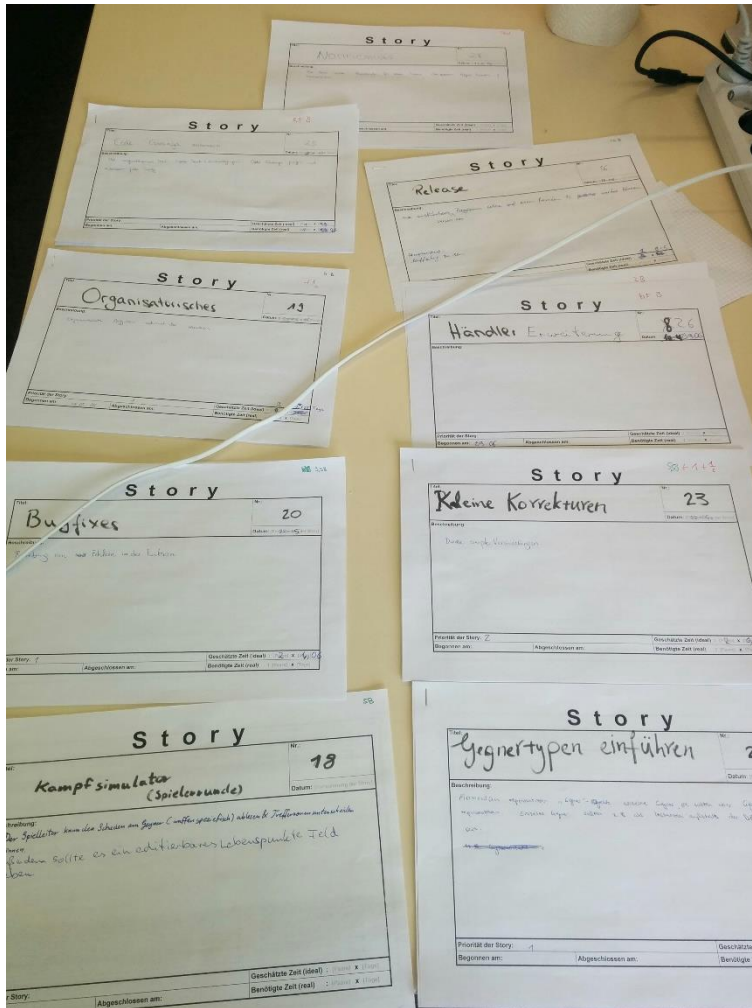
Tasks sind
Ablauforientierter

- Oft: Aufwand vs. Perfektion

Task	
Titel: [Kurze prägnante Benennung der Aufgabenstellung] GUI Elemente Funktionsfähig machen	
Task #: 3	Story #: 26
Datum: 09.06	
Abhängig von: [Nummern der Tasks, die vor diesem erledigt sein müssen.] 1, 2	
Beschreibung: [der Teilschritte, Betroffene Klassen und deren Verantwortlichkeiten und Kollaborationen. Grobe Implementationsskizze.]	
1. Trennen der Java Fxmls → Tabweise Implementierung der Controller 2. Initialisieren der Listen → ListView → Kategorie Tree Views 2. Jedem Element ein Klassenattribut zuweisen. 4. Vollständige Button Funktion herstellen 5. Korrekte Updates der Listen.	Notizen der Entwickler: [Probleme, Korrekturen an der Formulierung des Tasks. Hinweise zur Besonderheiten der Implementation.] <u>Punkt 1:</u> Aus Zeitgründen eingespart → War ein Fehler. <u>Punkt 5:</u> Eigentlich verschiedene Tasks für diese Ausrichtung und Gegenstände! • Außerdem geschickte Lösung der Kategorien umständlich.
Ausgegliederte Tasks: [Teilaufgaben, die erst während der Bearbeitung des Tasks erkannt wurden und zu aufwändig sind, um als Teil dieses Tasks angesehen zu werden. Jeweils Nummer und Titel.]	
Entwickler 1: Nooshin	Entwickler 2: Boris
Geschätzte Paar-Stunden: 4	
Bearbeitungszeiten: [Start- und Endzeiten der Bearbeitung, einflußlos des Datums auf die Werte.] Boris (1:30) Boris (2:30) Nooshin (3:30) Boris (3:30) 12.06. (3:30) 11.06. (2:00) 09.06. 13.06. (1:00)	
Begonnen am: 09.06.	Fertig am: 13.06
Bearbeitungszeit Summe: 21	



Projektdetails: Stories



Stories sind
unverändert
geblieben

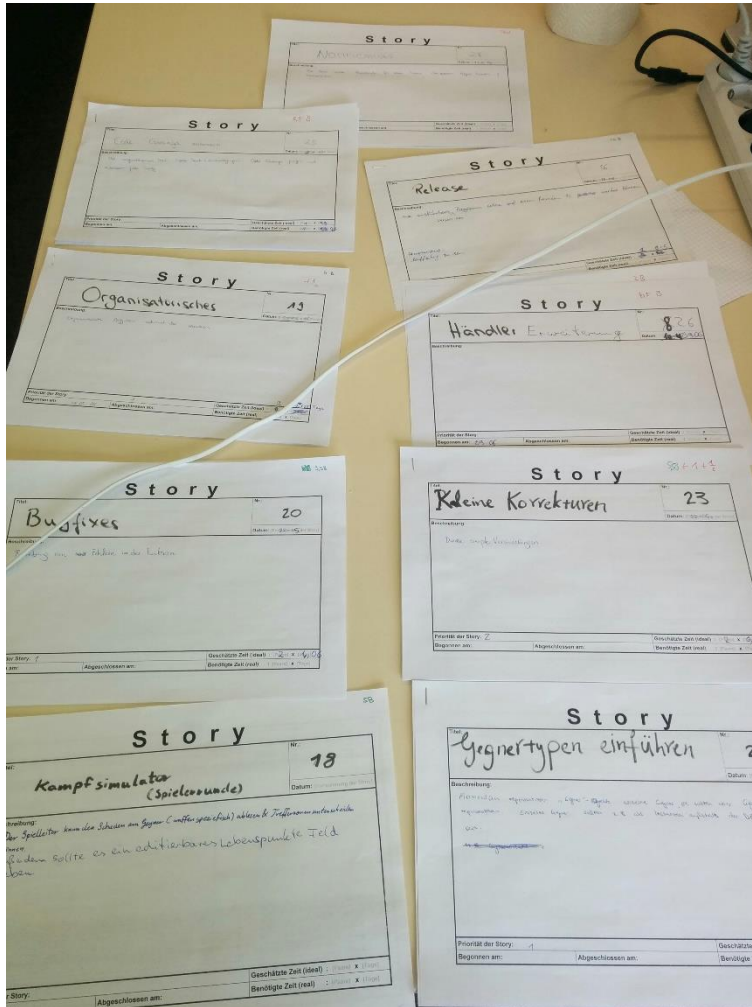
• Akzeptanztests gel.
nicht definierbar

Tasks sind
Ablauforientierter

• Manchmal: Aufwand
vs. Perfektion



Projektdetails: Stories



Stories sind
unverändert
geblieben

- Akzeptanztests
gel. nicht
definierbar

Tasks sind
Ablauforientierter

- Manchmal:
Aufwand vs.
Perfektion

Weniger „Out of
Burndown“ Zeit

- Es wurde mehr
erfasst



Projektdetails: Stories

Task

0,5 B

#: 23

Datum: 22.05.

Notizen der Entwickler: [Probleme, Korrekturen an der Formulierung des Tasks. Hinweise zur Besonderheiten der Implementation.]

Ausgelagerte Tasks: [Teilaufgaben, die erst während der...]

Stories sind
unverändert
geblieben

- Akzeptanztests
gel. nicht
definierbar

Tasks sind
Ablauforientierter

- Manchmal:
Aufwand vs.
Perfektion

Weniger „Out of
Burndown“ Zeit

- Es wurde mehr
erfasst



Projektdetails: Stories

Task

Datum: [der Formulierung des Tasks] 12.6.15

Notizen der Entwickler: [Probleme. Korrekturen an der Formulierung des Tasks. Hinweise zur Besonderheiten der Implementation.]

erantwortlichkeiten

Stories sind
unverändert
geblieben

- Akzeptanztests
gel. nicht
definierbar

Tasks sind
Ablauforientierter

- Manchmal:
Aufwand vs.
Perfektion

Weniger „Out of
Burndown“ Zeit

- Es wurde mehr
erfasst



Projektdetails: Stories

Geschätzte Zeit (ideal) : [Paare] x [Tage]	
58 + 1 + $\frac{1}{2}$	
Nr.:	23
Datum:	22.06.15

Stories sind
unverändert
geblieben

- Akzeptanztests
gel. nicht
definierbar

Tasks sind
Ablauforientierter

- Manchmal:
Aufwand vs.
Perfektion

Weniger „Out of
Burndown“ Zeit

- Es wurde mehr
erfasst
- Reibungslose
Übergänge



Projektdetails: Stories



Story Beispielfoto

Task Beispielfoto

1. Diese Iteration (Mehr Orientierung am Aufwand).
2. Gesamte Menge Beispielfoto (Aufwand vs. Priorität).



Projektdetails: Stories

- Später Hinzugefügt
- Später Erweitert
- Nicht abgeschlossen

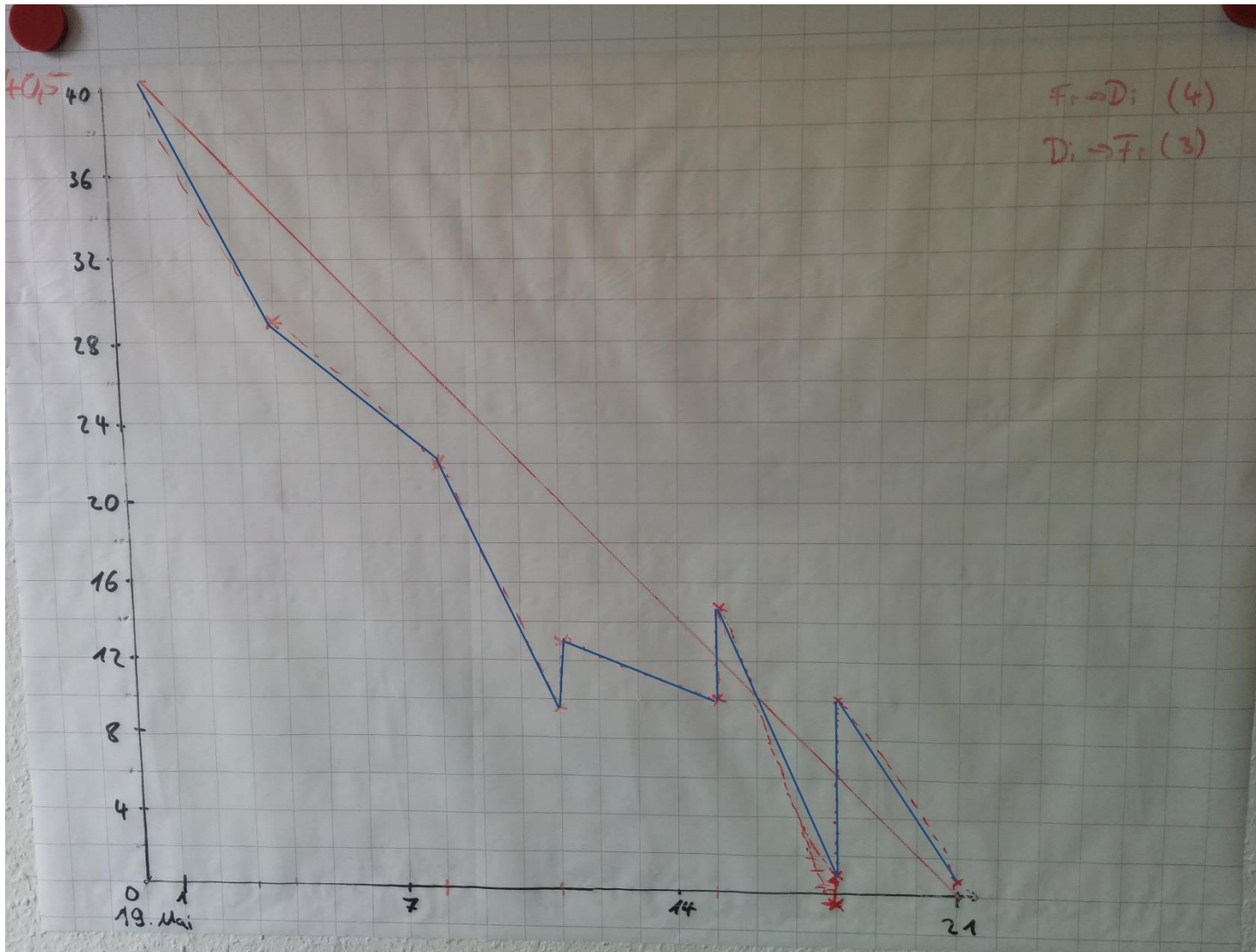
Nr	Titel	Prio	Bausteine
16	Release	1	4,5
20	Bugfix	1	9,5
18	Kampfsimulator : Spielerrunde	1	5 + x
25	Code Coverage verbessern	2	3,5
22	Gegnertypen einführen	1	8
27	Notificationsystem einführen	2	3
19	Organisatorisches	1	9 + 0,5
21	Vortrag	1	3
26	Händler Erweiterung	2	4,5
23	Feinschliff	2	5,5 + 5



Fortschritt: Aktueller Stand



Fortschritt: Burndown Chart



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablaufliste bei Tasks
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablaufliste bei Tasks
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablaufliste bei Tasks
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- **Vertraut mit Spielkonzept**



Erfahrung bei Stories und Entscheidungen:

- Ablaufliste bei Tasks
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

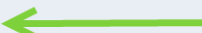
- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept

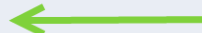


Erfahrung bei Stories und Entscheidungen:

- **Ablaufliste bei Tasks**
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests

Titel: Kampfsimulator (Gegnerrunde)

Beschreibung: 
Der Spielleiter bestreitet mit einer Spielergruppe ohne Spezialfälle. Dabei sollte der Schaden un

Akzeptanztest: 
-Trefferzone müssen sichtbar sein
- Beteiligung von vielen Spielern und Gegnern
- Lebenspunktverluste für Spieler anzeigen

Priorität der Story: 1

Begonnen am: 08.05. Abgeschlossen am:



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- **Ablaufliste bei Tasks**
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests

Kurzbeschreibung	Der Kampf wird gestartet
Akteure	Spielleiter
Vorbedingung	Es existiert eine Liste von
Ereignisfluss	1) Spielleiter wählt die 2) Spielleiter bestätigt 3) Kampffenster öffnet 4) Spielleiter kann eine
Nachbedingung	Schaden und Trefferzon angezeigt.
Alternativablauf	1) Spielleiter wählt eine



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- **Ablaufliste bei Tasks**
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests

Gegenstand	Beschreibung: [der Teilschritt, Betroffene Klassen und deren Verantwortlichkeiten und Kollaborationen, Grobe Implementationsskizze.]	
	1. Trennen der Java Exmls	→ Teilweise Implementierung der Controller
	2. Initialisieren der Listen	→ ListView → Kategorie Tree Views
	2. Jedem Element ein Klassenattribut zuweisen.	
	4. Vollständige Button Funktion herstellen	
5. Korrekte Updates der Listen.		
Entwickler 1: [Name]		Entwickler 2: [Name]



Fortschritt: Nicht Inhaltlich



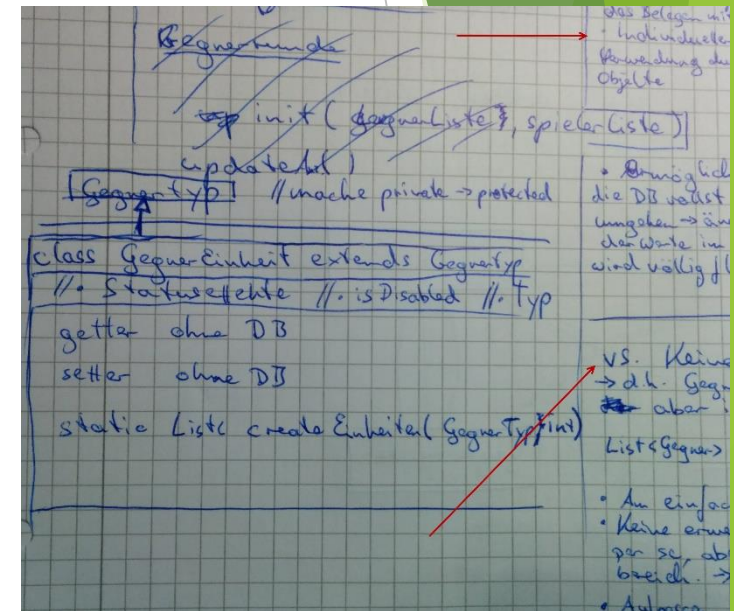
Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablauffliste bei Tasks
- Knappes Pro-Kontra, Aufwand v. Nutzen, CRC Cards etc. als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



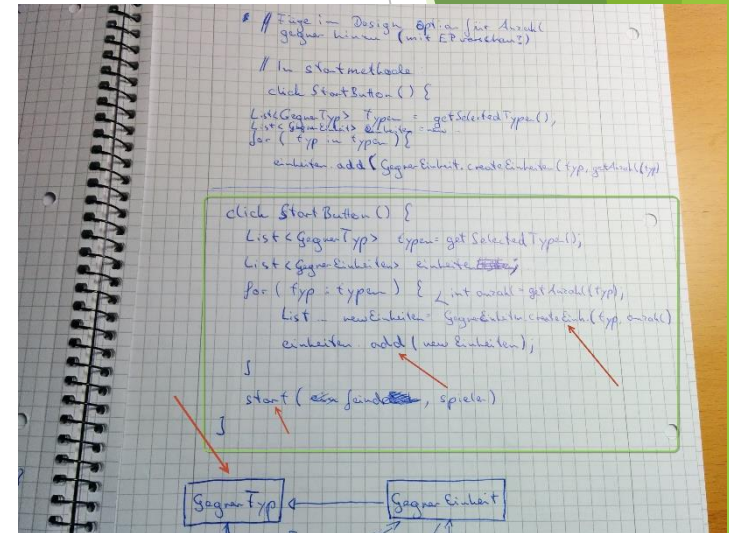
Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablauffliste bei Tasks
- Knappes Pro-Kontra, Aufwand v. Nutzen, CRC Cards etc. als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept



Erfahrung bei Stories und Entscheidungen:

- Ablauffliste bei Tasks
- Knappes Pro-Kontra, Aufwand v. Nutzen, CRC Cards etc. als Design Entscheidungs-Hilfen
- Viel Rücksicht auf Akzeptanztests

Titel: Kampfsimulator (Gegnerrunde)

Beschreibung:

Der Spielleiter bestreitet mit einer Spielergruppe ohne Spezialfälle. Dabei sollte der Schaden und T

Akzeptanztest:

- Trefferzone müssen sichtbar sein
- Beteiligung von vielen Spielern und Gegnern
- Lebenspunktverluste für Spieler anzeigen

Priorität der Story: 1

Begonnen am: 08.05.

Abgeschlossen am: 19



Fortschritt: Inhaltlich

- Erster Meilenstein

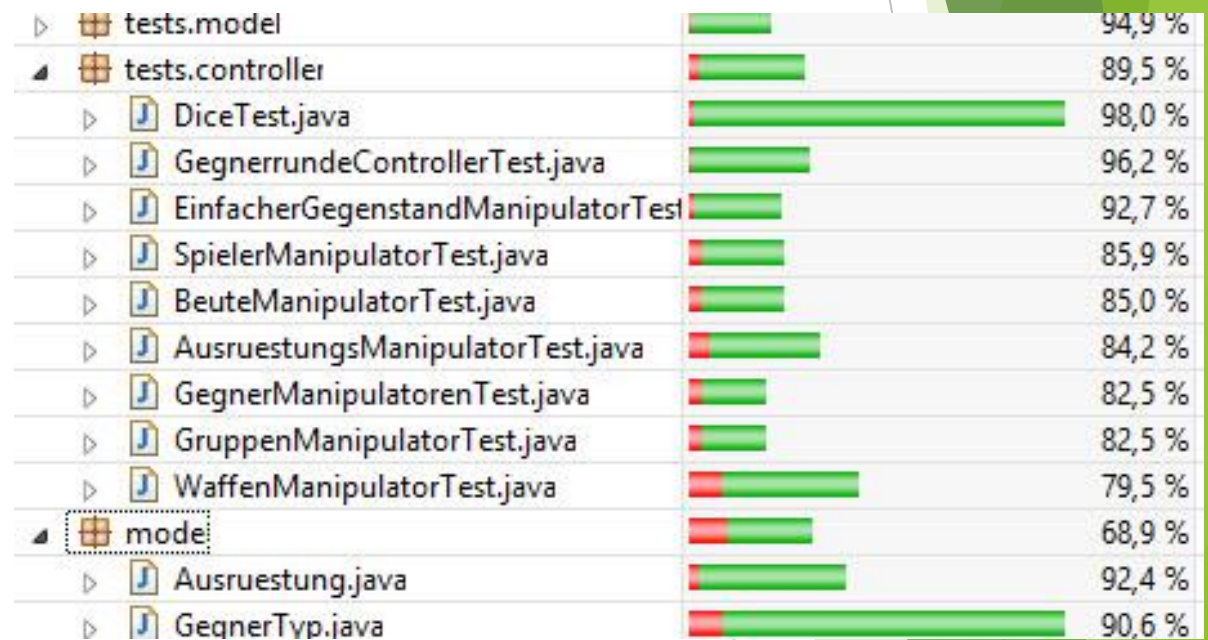


Fortschritt: Nicht Inhaltlich Testabdeckung

Code Coverage

- ▶ Versuch große Abdeckung zu erreichen
 - ▶ Erfolgreiche Übersicht
 - ▶ Stetige Erinnerung

Beispiel:

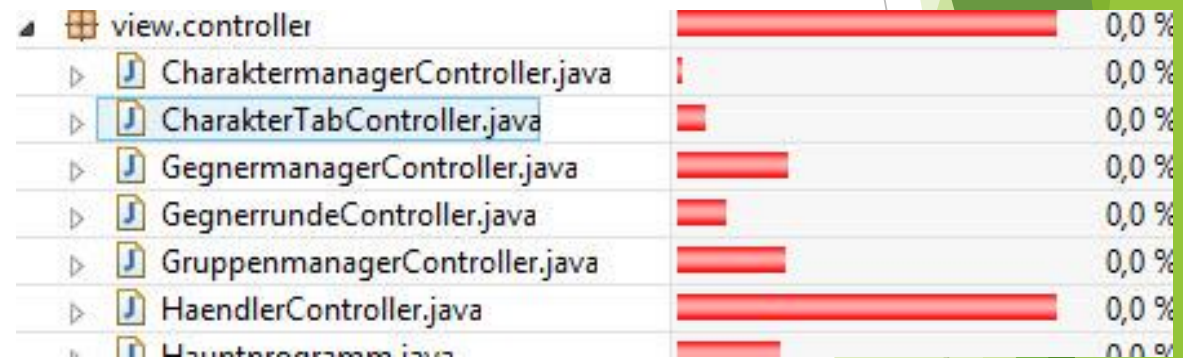


Fortschritt: Nicht Inhaltlich Testabdeckung

Code Coverage

- ▶ Versuch große Abdeckung zu erreichen
 - ▶ Erfolgreiche Übersicht
 - ▶ Stetige Erinnerung
- ▶ Problematische Bereiche:
 - ▶ Keine Abdeckung bei GUI

Beispiel:



Fortschritt: Nicht Inhaltlich Testabdeckung

Code Coverage

- ▶ Versuch große Abdeckung zu erreichen
 - ▶ Erfolgreiche Übersicht
 - ▶ Stetige Erinnerung
- ▶ Problematische Bereiche:
 - ▶ Keine Abdeckung bei GUI
 - ▶ Mangelhafte Interpretation

Beispiel:

```
try {  
    return getAllRows.getResultList();  
}  
catch (IllegalStateException getResultListExceptionOne) {  
    System.err.println("IllegalStateException: " +  
        getResultListExceptionOne.printStackTrace());  
    return null;  
}  
catch (QueryTimeoutException getResultListExceptionTwo) {  
    System.err.println("QueryTimeoutException: " +  
        getResultListExceptionTwo.printStackTrace());  
    return null;  
}  
catch (TransactionRequiredException getResultListExceptionThree) {  
    System.err.println("TransactionRequiredException: " +  
        getResultListExceptionThree.printStackTrace());  
    return null;  
}  
catch (PessimisticLockException getResultListExceptionFour) {  
    System.err.println("PessimisticLockException: " +  
        getResultListExceptionFour.printStackTrace());  
    return null;  
}  
catch (LockTimeoutException getResultListExceptionFive) {  
    System.err.println("LockTimeoutException: " +  
        getResultListExceptionFive.printStackTrace());  
    return null;  
}
```



Demo

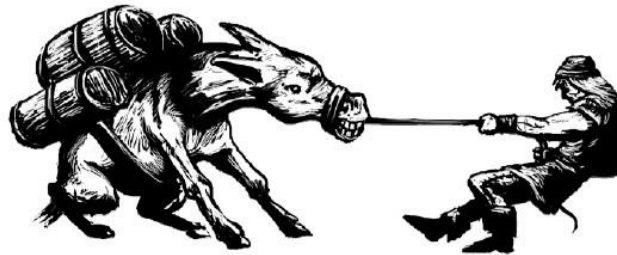


Umfang der Demo

Kampf



Handel



Reflexionen



Reflexionen: Arbeitsprozess

- Zum Mitnehmen -

- ▶ Vereinfachte Designentscheidung durch Agile Softwareentwicklung! (Kommt anders als gedacht)
 - ▶ Baut auf Basisfunktionen auf bzw. drum herum. Automatisch werden wichtige Eigenschaften zentriert.
- ▶ Gemeinsamer Code dank:
 - ▶ Coding Conventions
 - ▶ Sauberem Code (regelmäßiger Austausch)
 - ▶ Jeder Erfolg wurde bemerkt!
- ▶ Effiziente Designentwicklung dank regem Austausch (Gegnertypenbeispiel)
 - ▶ Zielorientiertes Arbeiten dank klarer Zielsetzung und Diskussion.



Reflexionen: Arbeitsprozess

(Aus welchen „Fehlern“ bzw. Ereignissen haben wir gelernt?)

- Gemeinschaft stärken (Ausflug)
- Unerwartetes gibt es immer: Release Laufzeit



Reflexionen: Vorschau auf nächste Iteration

- ▶ Handbuch erstellen
- ▶ Voarbggespräch mit dem Kunden für den finalen Release
- ▶ Fähigkeiten von Charakteren einführen
- ▶ Effekte im Kampf einführen
- ▶ Aufbesserungen für den Release abschließen und finales Kundengespräch



Reflexionen: Fazit

- ▶ Alle lernen stetig sehr viel
- ▶ Spaß am Arbeiten: Dank an Engagiertes Team
 - ▶ Nooshin bei der Arbeit
 - ▶ Andreas ruft wegen Ideen an
- ▶ Froh kein Teamleiter mehr zu sein
- ▶ Einfluss auf Privatleben
 - ▶ Sobald ich etwas nicht schätzen konnte, habe ich die Päckchen kleiner gemacht.



Danke für eure Aufmerksamkeit!

