



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:	44		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 Groß
Voller Titel: Groß
von Nesosanden


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

Rasse: Windling
Spezielle Fertigkeit: flieg 110cm
Gassenwissen
Disziplin: Elementarst
Anfangsfertigkeit: Feuerbeherrschung

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

Größe: 3 cm
Alter: 15 Jahre
Hautfarbe: dunkel
Haarfarbe: rostig braun
Augenfarbe: grün
Talent: 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			100			10.000			1.000.000		
Erfahrungspunkte:	5K			1247,5K								

Handwritten notes: "Tausendsassa", "Talente: Tausendsassa", "verbleibende Wertpunkte: 13", "Erfahrungspunkte: 5K, 1247,5K, 12.8K".



Einleitung

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



Einleitung

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



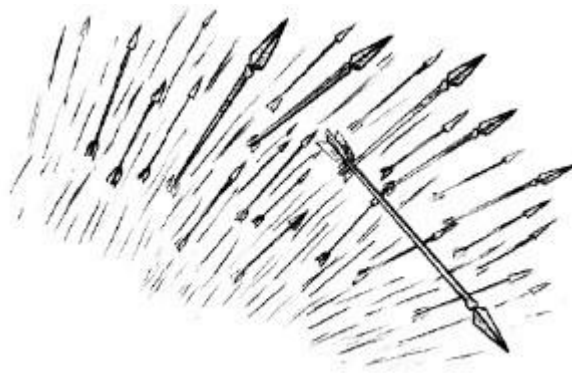
Einleitung

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



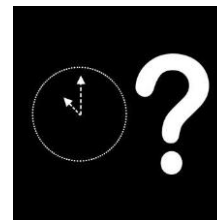
Einleitung

- 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

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



Projektdetails

Stories

Task		
Titel: [Kurze prägnante Benennung der Aufgabenstellung] GUI Elemente Funktionsfähig machen		
Task #: 3	Story #: 26	Datum: [das Formulierung des Tasks] 09.06
Abhängig von: [Nummern der Tasks, die vor diesem erledigt sein müssen.] 1, 2		Notizen der Entwickler: [Probleme, Korrekturen an der Formulierung des Tasks. Hinweise zur Besonderheiten der Implementation.] Punkt 1: Aus Zeitgründen eingespart → War ein Fehler. Punkt 5: Eigentlich verschiedene Tasks für Klass Ausrichtung und Gegenstände! • Außerdem geschickte Lösung der Kategorien umständlich.
Beschreibung: [der Teilschritte, Betroffene Klassen und deren Verantwortlichkeiten und Kollaborationen. Grobe Implementationsskizze.] 1. Trennen der Java Fxmls → Teilweise Implementierung der Controller 3. Initialisieren der Listen → ListView → Kategorie Tree Views 2. Jedem Element ein Klassenattribut zuweisen. 4. Vollständige Button Funktion herstellen 5. Korrekte Updates der Listen.		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
Bearbeitungszeiten: [Start- und Endzeiten der Bearbeitung eines Tasks auf die Uhrzeit des Vorgesetzten genau.] Boris (1:30) 12.06. Boris (0:30) 11.06. Nooshin (3:00) 09.06. Boris (1:00) 13.06. 		Geschätzte Paar-Stunden: 4
Begonnen am: 09.06.		Fertig am: 13.06
		Bearbeitungszeit Summe: 21

Stories sind unverändert geblieben

Tasks sind Ablauforientierter

•Manchmal: Aufwand vs. Perfektion



Projektdetails

Stories

The image shows a project management form with a green box highlighting the number '053' in the top right corner. The form has a header section with the word 'Task' and a box for a number, which contains '23'. Below this, there is a section for 'Notizen der Entwickler' (Developer Notes) and a section for 'Ausgewählte Tasks' (Selected Tasks). The form is partially filled out with handwritten text, including 'rsch. Modellklassen' and 'en'.

	Stories sind unverändert geblieben	
	Tasks sind Ablauforientierter	•Manchmal: Aufwand vs. Perfektion
	Weniger „Out of Burndown“ Zeit	•Es wurde mehr erfasst



Projektdetails

Stories

Task

1/2 B

23 Datum: [der Formulierung des Tasks] 12.6.15

n müssen.]

erantwortlichkeiten

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

wo] x [Tage]

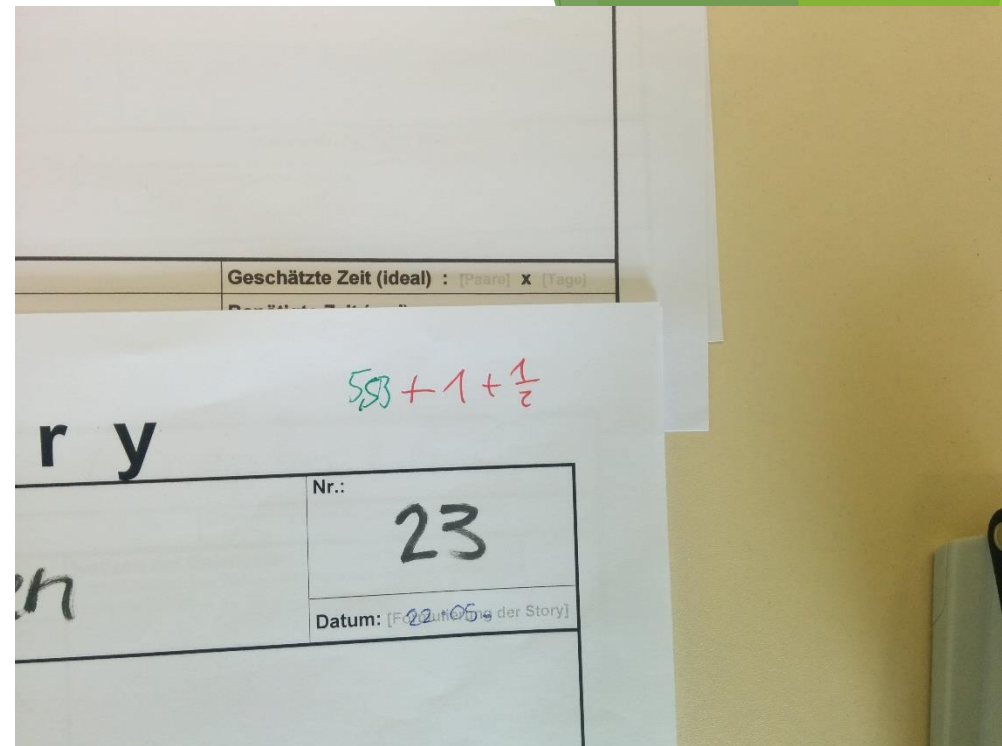
ve] x [Tage]

	Stories sind unverändert geblieben	
	Tasks sind Ablauforientierter	•Manchmal: Aufwand vs. Perfektion
	Weniger „Out of Burndown“ Zeit	•Es wurde mehr erfasst



Projektdetails

Stories



Projektdetails: Stories

- Später Hinzugefügt
- Später Erweitert
- Noch aktiv

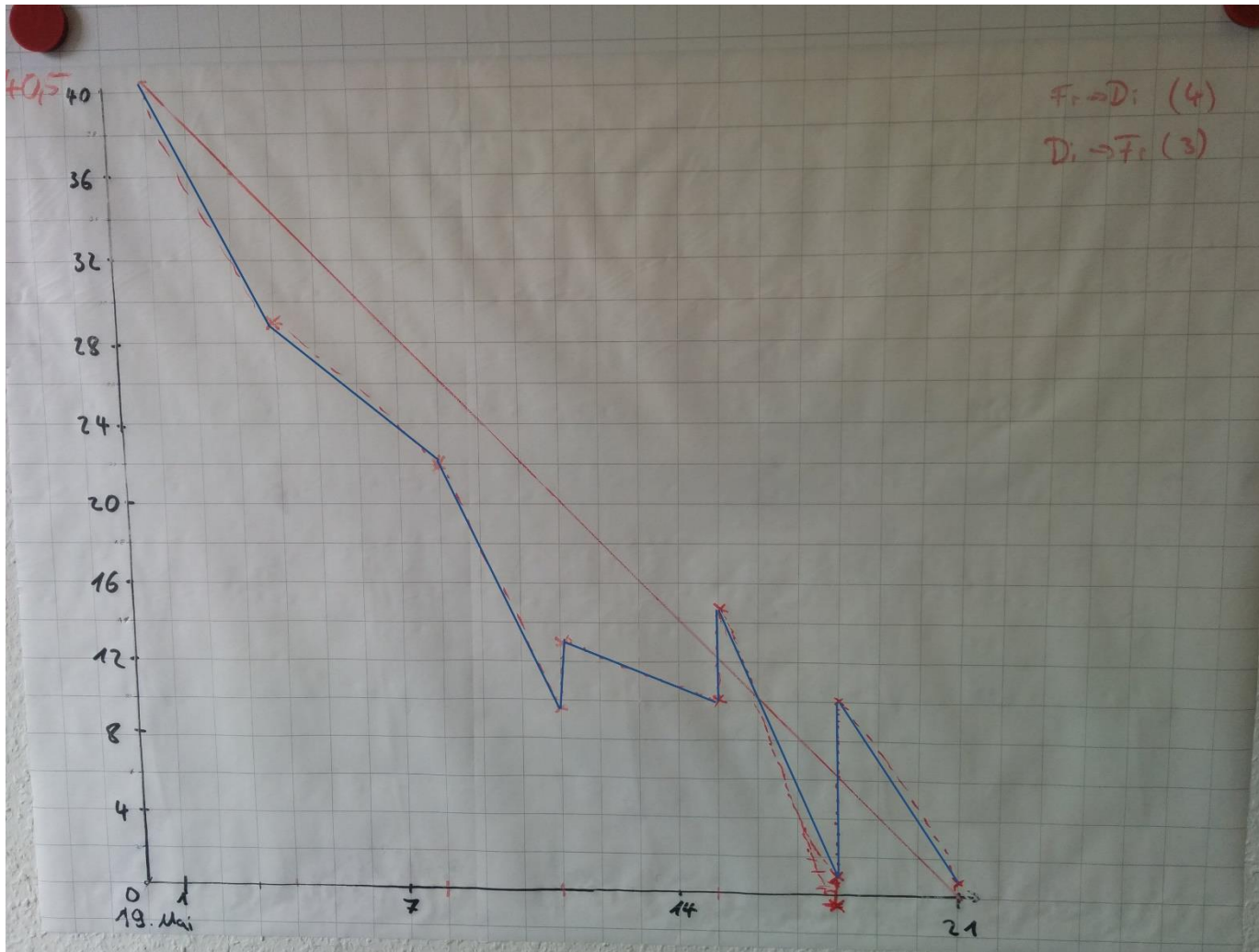
Nr	Titel	Prio	Bausteine
16	Release	1	4,5
20	Bugfix	1	9,5
18	Kampfsimulator : Spielerrunde	1	5 + x
25	Code Coverage verbessern	2	3,5
22	Gegnertypen einführen	1	8
27	Notificationsystem einführen	2	3
19	Organisatorisches	1	9 + 0,5
21	Vortrag	1	3
26	Händler Erweiterung	2	4,5
23	Feinschliff	2	5,5 + 6



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



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

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



Erfahrung bei Stories und Entscheidungen:

- Ablafliste bei Tasks
- Knappes Pro-Kontra (CRC Cards etc.) als Design Entscheidungs-Hilfen



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



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



Fortschritt: Nicht Inhaltlich



Parallelisierung (Relaxes Extreme Programming)

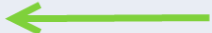
- Bildung von Bereichsexperten
- Mehr Sensitivität bei Arbeitsteilung
- Vertraut mit Spielkonzept

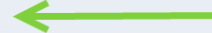


Erfahrung bei Stories und Entscheidungen:

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

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

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 Trefferzone 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

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: /		Entwickler 2: D



Fortschritt: Nicht Inhaltlich



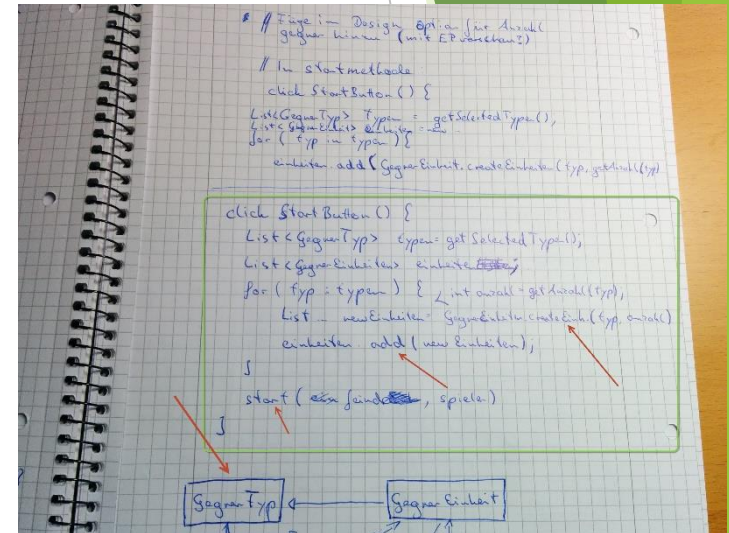
Parallelisierung (Relaxes Extreme Programming)

- 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



Fortschritt: Nicht Inhaltlich



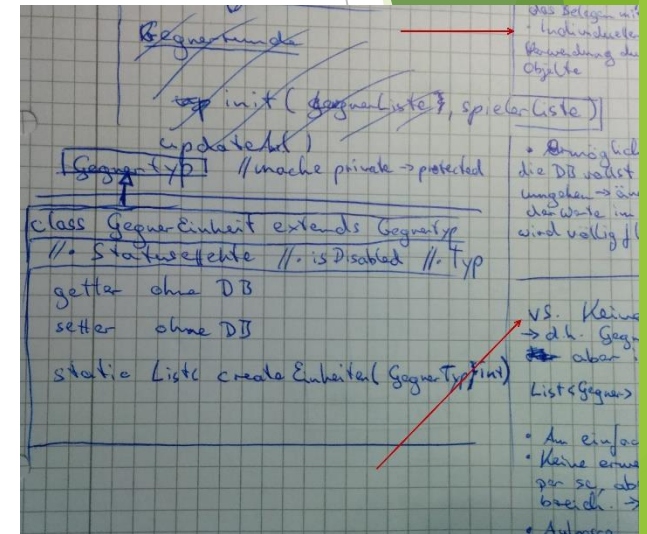
Parallelisierung (Relaxes Extreme Programming)

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



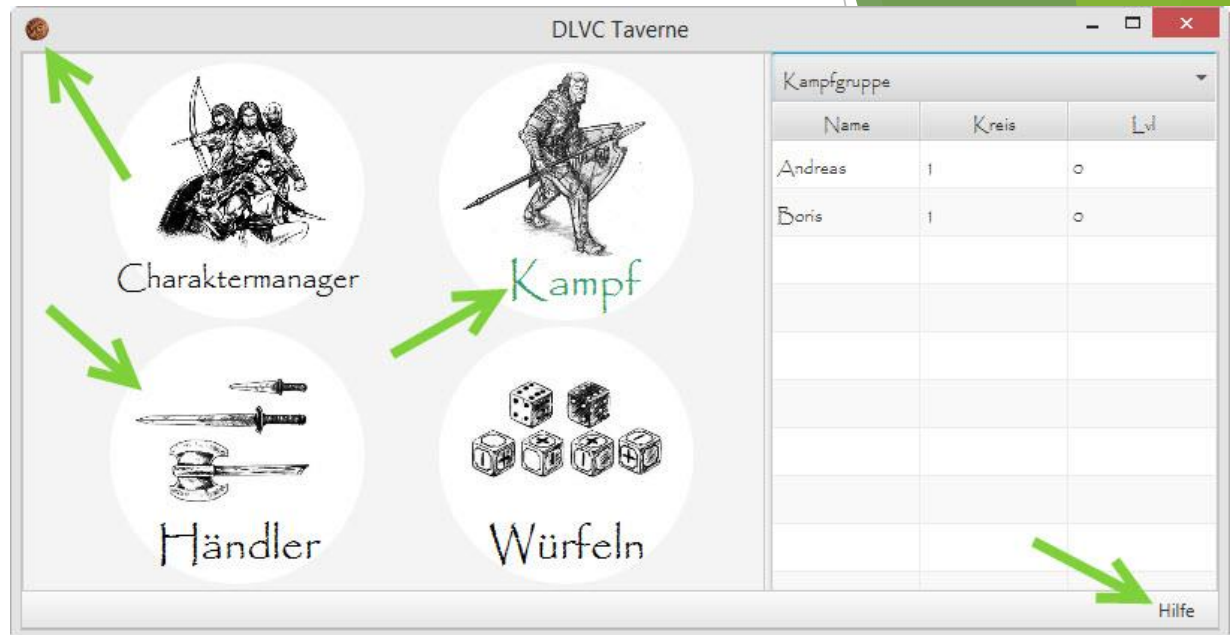
Erfahrung bei Stories und Entscheidungen:

- Ablaufliste bei Tasks
- Knappes Pro-Kontra, Aufwand v. Nutzen, CRC Cards etc. als Design Entscheidungs-Hilfen



Fortschritt: Inhaltlich

- Alle Fenster
 - Programmicon Verfügbar
 - Neue Schriftart
 - Hilfe Button
 - Notification Framework
- Hauptmenü
 - Runde Buttons
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
- Händler: Gegenstände und Ausrüstung



Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
 - Gegnertypen sind verfügbar
 - Man kann sie nach Kreis sortieren
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
- Händler: Gegenstände und Ausrüstung

Kampf - Teilnehmerauswahl

Spielercharakter: Supa Dupa Gruppe

Suche...

Gegner:

- ☒ Kreis 1
- ☒ Kreis 2
- ☒ Kreis 3
- ☒ Kreis 4

Filter

Kreis	Level	Name
1	0	Bogenschütze
2	0	Schwertkämpfer

Kampfteilnehmer:

Andreas

Boris

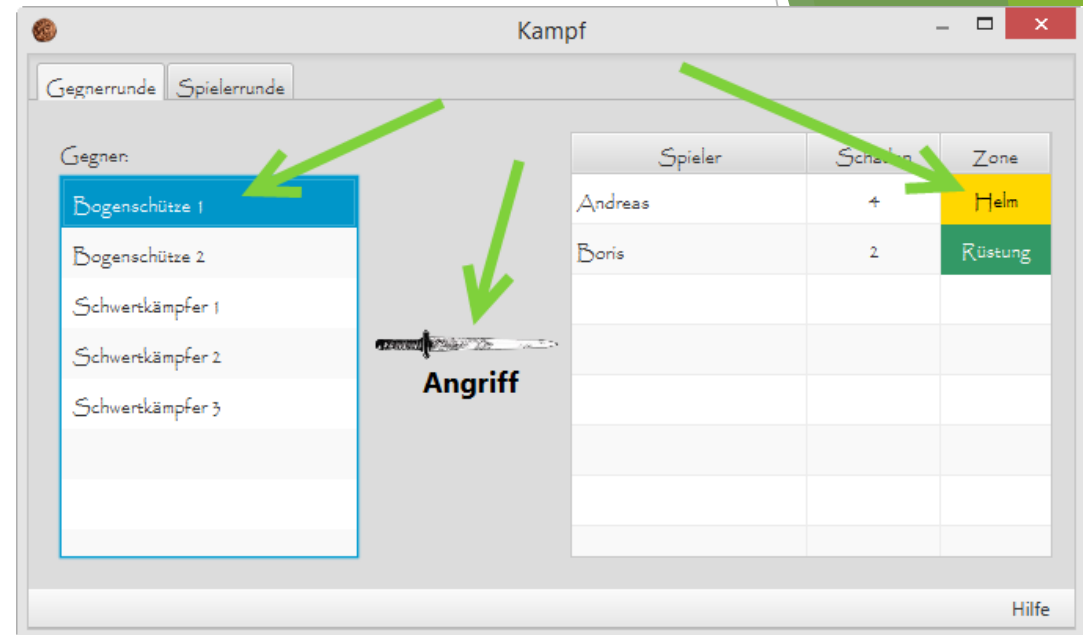
Gegnertyp	Anzahl
Bogenschütze	2
Schwertkämpfer	3

Kampf!



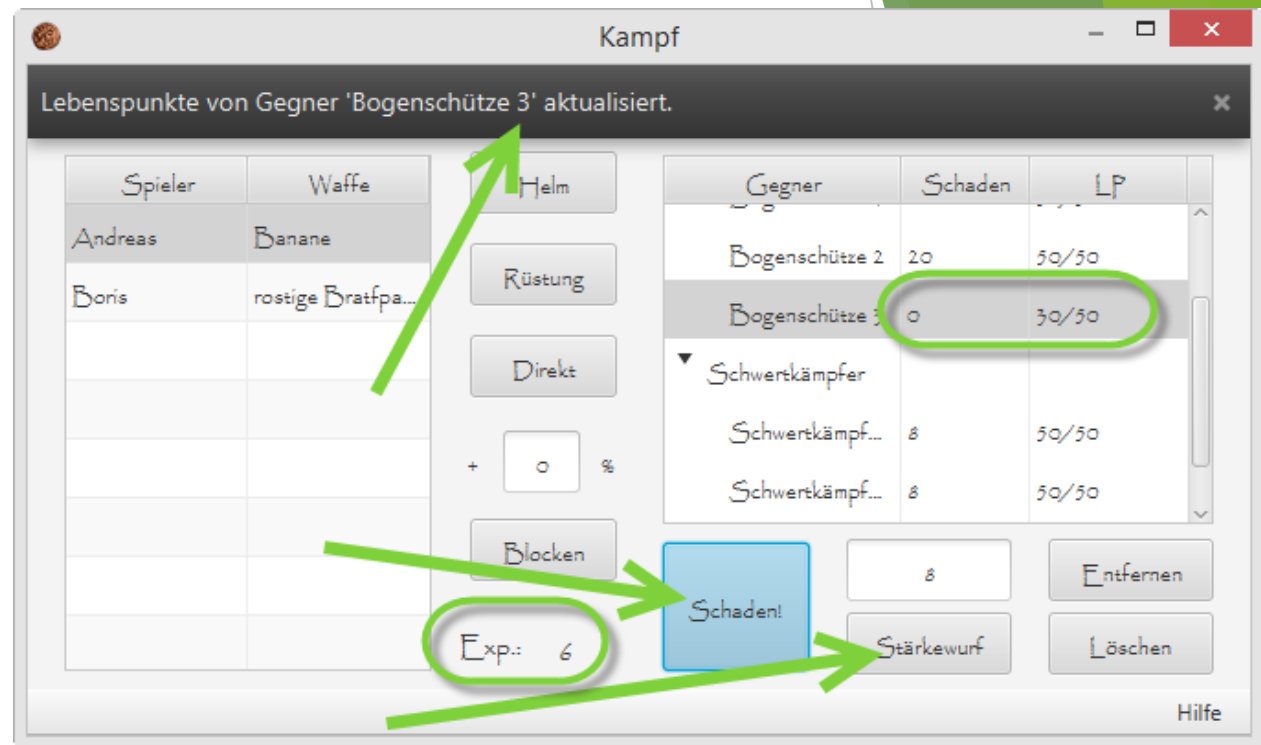
Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
 - Angriff Button
 - Nummerierte Gegnereinheiten
 - Farbliches Highlight der Trefferzone
- Kampf: Spielerrunde
- Händler: Gegenstände und Ausrüstung



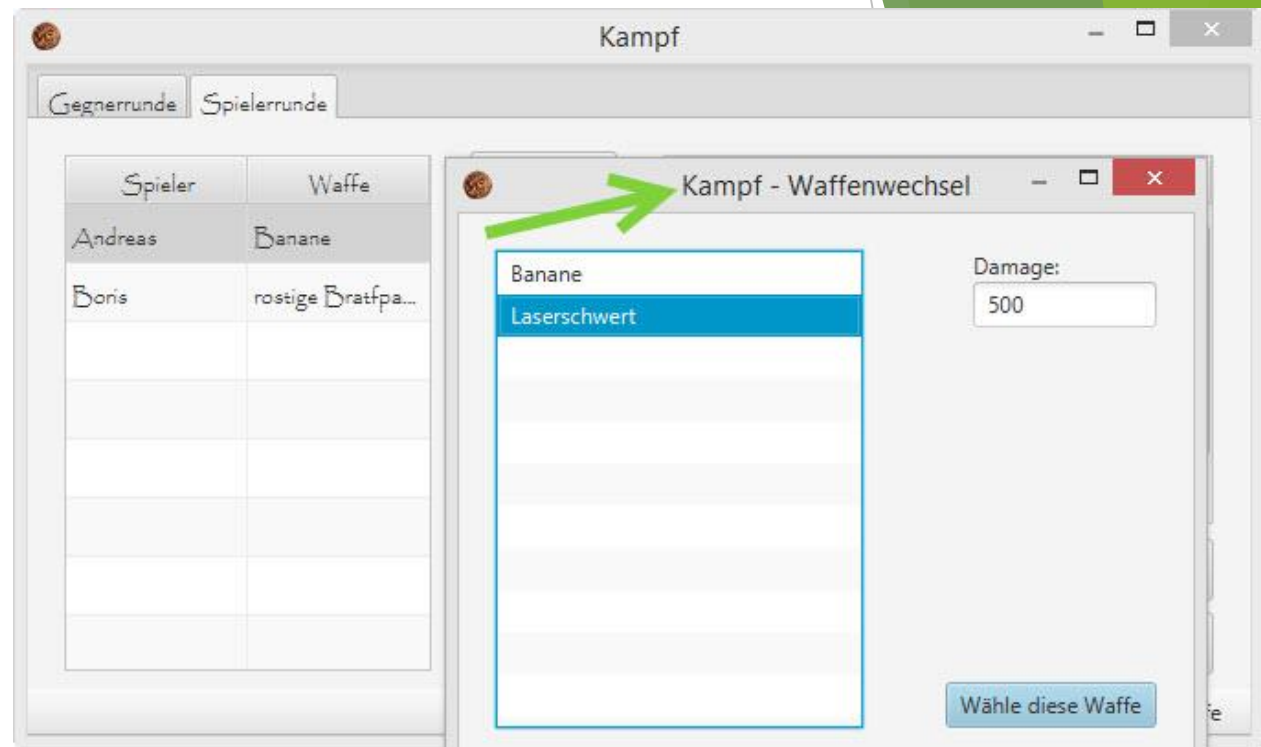
Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
 - Schaden kann angezeigt und zugefügt werden
 - Der Gegner kann einen Stärke und Blockwurf durchführen
 - Man kann kampfunfähige Gegner aus dem Kampf entfernen
 - Gegner können gelöscht werden
 - Erfahrung kann abgelesen werden.
 - Notificationframework und Tooltips sind vorhanden
- Händler: Gegenstände und Ausrüstung



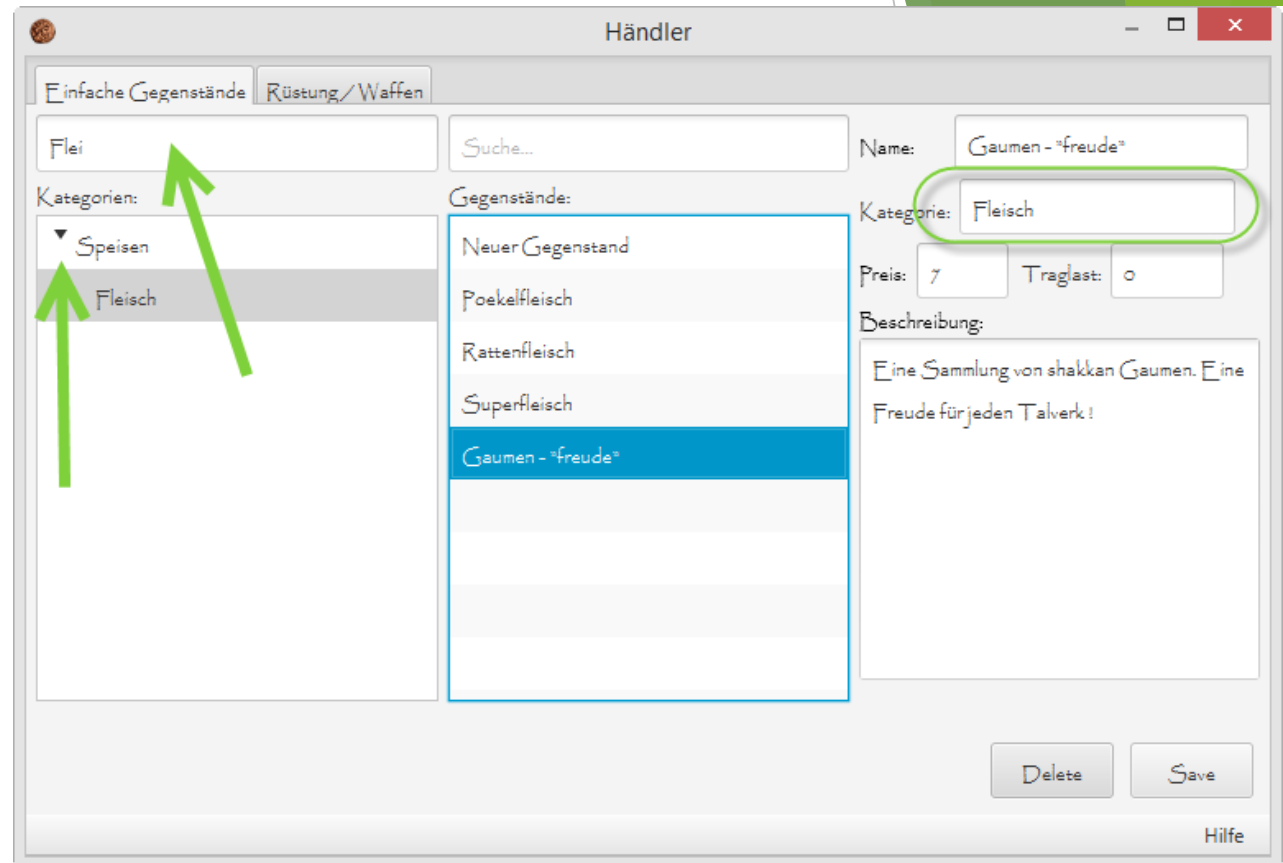
Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
 - Schaden kann angezeigt und zugefügt werden
 - Der Gegner kann einen Stärke und Blockwurf durchführen
 - Man kan kampfunfähige Gegner aus dem Kampf entfernen
 - Gegner können gelöscht werden
 - Erfahrung kann abgelesen werden.
 - Notificationframework und Tooltips sind vorhanden
- Händler: Gegenstände und Ausrüstung



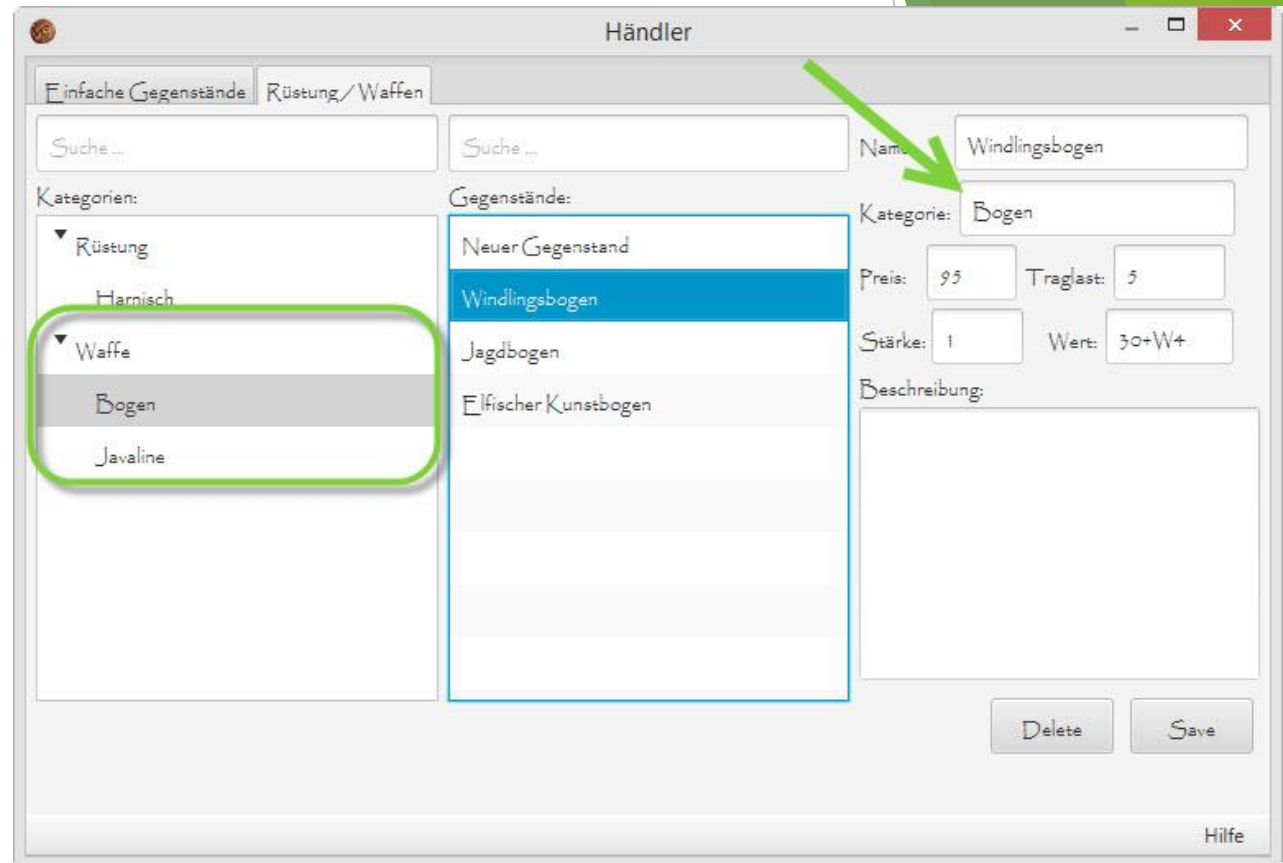
Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
- Händler: Gegenstände und Ausrüstung
 - Einfaches hinzufügen von Kategorien UND Subkategorien
 - Suche nach Gegenständen und (Sub)-Kategorien



Fortschritt: Inhaltlich

- Alle Fenster
- Hauptmenü
- Kampf: Teilnehmerauswahl
- Kampf: Gegnerrunde
- Kampf: Spielerrunde
- Händler: Gegenstände und Ausrüstung
 - Einfaches hinzufügen von Kategorien UND Subkategorien
 - Suche nach Gegenständen und (Sub)-Kategorien
 - Waffen u. Rüstungen werden getrennt behandelt (Wert/Stärke)
 - Neue Gegenstände werden automatisch der gewählten Kategorie zugeordnet.

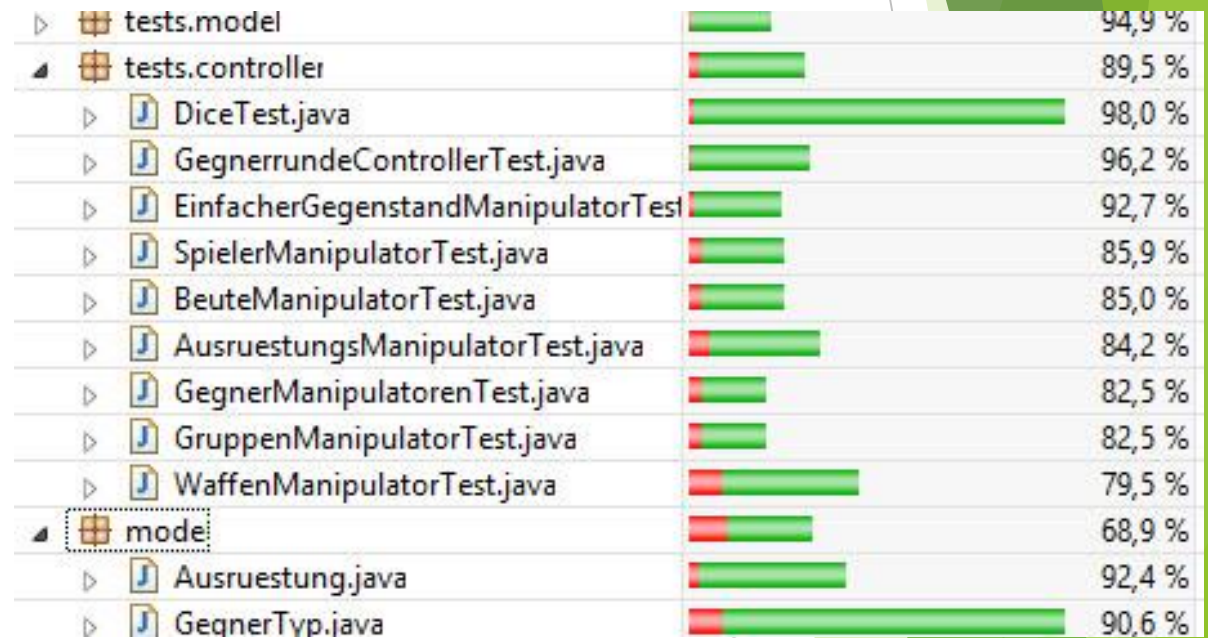


Fortschritt: Nicht Inhaltlich Testabdeckung

Code Coverage

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

Beispiel:

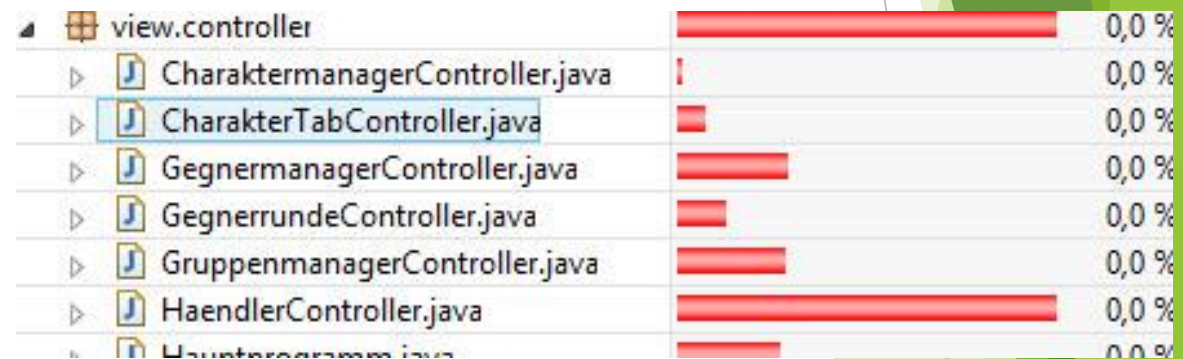


Fortschritt: Nicht Inhaltlich Testabdeckung

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

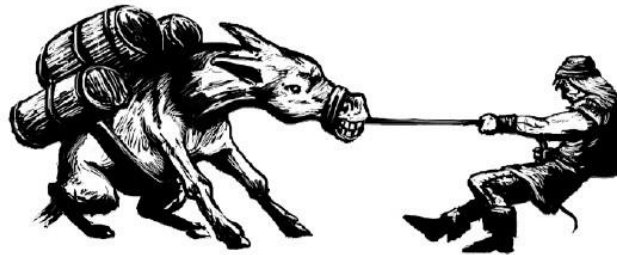


Umfang der Demo

Kampf



Handel



Reflexionen



Reflexionen: Arbeitsprozess

- ▶ Agile Softwareentwicklung hat zu einfacherem Design geführt
 - ▶ Erweiterungen finden auf lauffähigen Grundfunktionen statt. Das öffnet Sicht auf alternative Ideen (Händler).
- ▶ Einfaches „gemeinsames“ coden dank:
 - ▶ Coding Conventions
 - ▶ Sauberem Code (regler Austausch)
 - ▶ Jeder Erfolg wurde bemerkt! -> stete Motivation
- ▶ Unerwartete Hindernisse:
 - ▶ Laufzeit der Releaseversion



Reflexionen: Vorschau auf nächste Iteration

- ▶ Kundengespräch vor der finalen Iteration
- ▶ Fähigkeiten für Spieler und Nicht Spieler Charaktere verfügbar machen
- ▶ Kampfende erweitern (z.B. Loot, Kampftabelle)
- ▶ Kundengespräch vor Release



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!

