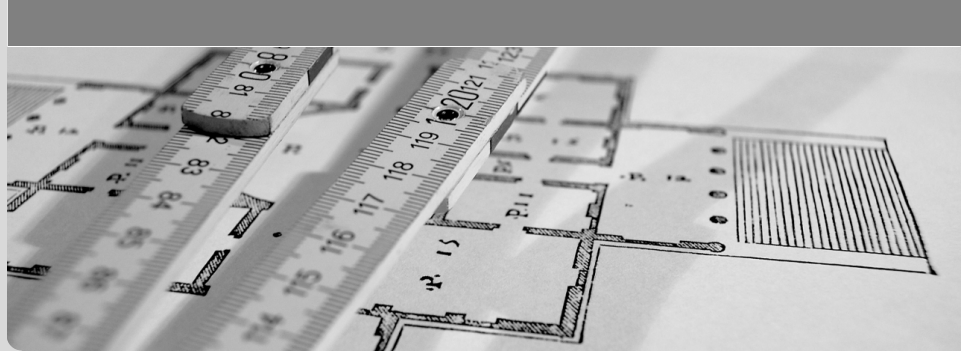


# Softwaretechnik

## 1. Tutorium

Lastenheft, UML, Linguistische Analyse

Tino Fuhrmann | 8. Mai 2015



- 1 Blatt 0 und 1
- 2 Wasserfallmodel
- 3 Planung
  - Lastenheft
  - Durchführbarkeit
- 4 UML
- 5 Linguistische Analyse

- Exception.java werfen ist schlechter Stil
- leerer Javadoc ist auch fehlerhaft

## ■ 1.1

- Patch nicht angewendet
- richtiges Projekt hochladen
- gitignore vollständig machen!
- Hinweis: 0 Punkte bei Checkstyle falls keine Doku via commits. Wenn doch gemacht: Mail an mich mit Matrikelnummer und Datei + Zeilen

## ■ 1.2

- Tests in Paket der getesteten Klasse (Pfad:  
`src/test/java/org/jis/generator/GeneratorTest.java`)

## ■ 1.3

- File/Generator generiert den Pfad nicht komplett, sondern versucht die Datei in dem angegebenen Verzeichnis zu erstellen
- schauen, was tatsächlich in die Zip gepackt wird (Ordner zusätzlich)

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung
- Definition

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung
- Definition
- Entwurf



Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung
- Definition
- Entwurf
- Implementierung

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung
- Definition
- Entwurf
- Implementierung
- Testen

Welche Phasen hat das Wasserfallmodell (richtige Reihenfolge)?

- Planung
- Definition
- Entwurf
- Implementierung
- Testen
- Abnahme, Einsatz und Wartung

Kapitel?

- Zielbestimmungen

Kapitel?

- Zielbestimmungen
- Produkteinsatz

## Kapitel?

- Zielbestimmungen
- Produkteinsatz
- Funktionale Anforderungen

## Kapitel?

- Zielbestimmungen
- Produkteinsatz
- Funktionale Anforderungen
- Produktdaten

## Kapitel?

- Zielbestimmungen
- Produkteinsatz
- Funktionale Anforderungen
- Produktdaten
- Nichtfunktionale Anforderungen



## Kapitel?

- Zielbestimmungen
- Produkteinsatz
- Funktionale Anforderungen
- Produktdaten
- Nichtfunktionale Anforderungen
- Systemmodelle mit Szenarien und Anwendungsfällen

## Kapitel?

- Zielbestimmungen
- Produkteinsatz
- Funktionale Anforderungen
- Produktdaten
- Nichtfunktionale Anforderungen
- Systemmodelle mit Szenarien und Anwendungsfällen
- Glossar

- Benutzer soll mit anderen über Internetdienst Brettspiele spielen können
- Spiele: Mühle, Dame und Schach
- Modulare Integrierbarkeit weiterer Spiele
- Eigenständige Registrierung der Nutzer
- Angemeldeter Nutzer kann sich
  - alle Spiele des Systems ansehen
  - neue Spiele eröffnen
  - zu Spielen herausfordern/herausgefordert werden.
- Spiele sind öffentlich einsehbar
- Jeder Benutzer hat Daten, die er geheim oder öffentlich halten will
- Jeder Benutzer erhält eine Wertungszahl (ELO) für jedes Spiel
- Es gibt Auszeichnungen für schnellster/langsamster Spieler des Tages

Wofür soll das Produkt genutzt werden?

Welche Funktionen soll das Produkt haben?

Welche Daten werden gespeichert?

Anforderungen: **Messbare** Anforderungen, die nicht funktional sind.

Welche Akteure?



Welche Akteure?

Welche Anwendungsfälle gibt es?

Definitionen von verwendeten Begriffen.

Kann das Projekt durchgeführt werden?

- Fachliche Durchführbarkeit?
  - softwaretechnisch?
  - Entwicklungs/Zielmaschinen verfügbar?

Kann das Projekt durchgeführt werden?

- Fachliche Durchführbarkeit?
  - softwaretechnisch?
  - Entwicklungs/Zielmaschinen verfügbar?
- Alternative Lösungsvorschläge?
  - Kauf + Anpassung von Standardsoftware?

Kann das Projekt durchgeführt werden?

- Fachliche Durchführbarkeit?
  - softwaretechnisch?
  - Entwicklungs/Zielmaschinen verfügbar?
- Alternative Lösungsvorschläge?
  - Kauf + Anpassung von Standardsoftware?
- Personelle Durchführbarkeit?
  - Qualifizierte Fachkräfte vorhanden?

Kann das Projekt durchgeführt werden?

- Fachliche Durchführbarkeit?
  - softwaretechnisch?
  - Entwicklungs/Zielmaschinen verfügbar?
- Alternative Lösungsvorschläge?
  - Kauf + Anpassung von Standardsoftware?
- Personelle Durchführbarkeit?
  - Qualifizierte Fachkräfte vorhanden?
- ökonomische Durchführbarkeit?
  - Aufwands + Termschätzung
  - Wirtschaftlichkeitsrechnung

Kann das Projekt durchgeführt werden?

- Fachliche Durchführbarkeit?
  - softwaretechnisch?
  - Entwicklungs/Zielmaschinen verfügbar?
- Alternative Lösungsvorschläge?
  - Kauf + Anpassung von Standardsoftware?
- Personelle Durchführbarkeit?
  - Qualifizierte Fachkräfte vorhanden?
- ökonomische Durchführbarkeit?
  - Aufwands + Termschätzung
  - Wirtschaftlichkeitsrechnung
- Rechtliche Gesichtspunkte
  - Datenschutz, relevante Standards
- Risiken bei der Durchführung

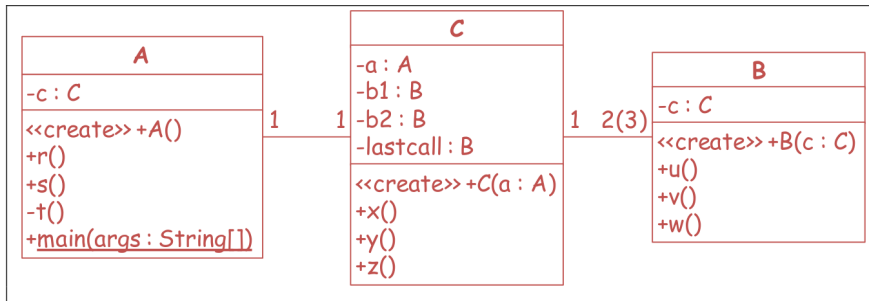
## Aufgabe: Zeichnet zu folgendem Code ein Klassendiagramm!

```
01 public class A {  
02     private C c;  
03  
04     /**  
05      * Hier geht es los.  
06      */  
07     public static void  
main(String[] args) {  
08         A a = new A();  
09         a.r();  
10     }  
11  
12     public A() {  
13         c = new C(this);  
14     }  
15  
16     public void r() {  
17         c.y();  
18     }  
19  
20     public void s() {  
21         t();  
22         c.z();  
23         t();  
24     }  
25  
26     private void t() {  
27     }  
28 }  
29
```

```
30 public class B {  
31     private C c;  
32  
33     public B(C c) {  
34         this.c = c;  
35     }  
36  
37     public void u() { }  
38  
39     public void v() {  
40         this.u();  
41     }  
42  
43     public void w() {  
44         c.x();  
45     }  
46 }
```

```
47 public class C {  
48     private A a;  
49     private B b1, b2;  
50     private B lastCall;  
51  
52     public C(A a) {  
53         this.a = a;  
54         b1 = new B(this);  
55         b2 = new B(this);  
56     }  
57  
58     public void x() {  
59         if (b1 == lastCall) {  
60             b2.u();  
61         } else {  
62             b1.u();  
63         }  
64     }  
65  
66     public void y() {  
67         lastCall = b1;  
68         b1.w();  
69         a.s();  
70         lastCall = b2;  
71         b2.w();  
72     }  
73  
74     public void z() {  
75         b1.v();  
76         b2.v();  
77     }  
78 }
```



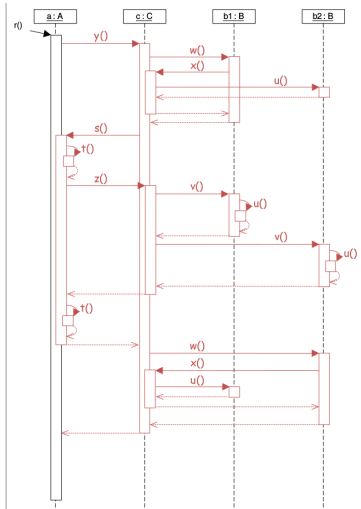


Aufgabe: Zeichnet zu folgendem Code ein Sequenzdiagramm des Aufrufs von `r()`!

```
01 public class A {  
02     private C c;  
03  
04     /**  
05      * Hier geht es los.  
06      */  
07     public static void  
main(String[] args) {  
08         A a = new A();  
09         a.r();  
10     }  
11  
12     public A() {  
13         c = new C(this);  
14     }  
15  
16     public void r() {  
17         c.y();  
18     }  
19  
20     public void s() {  
21         t();  
22         c.z();  
23         t();  
24     }  
25  
26     private void t() {  
27     }  
28 }  
29
```

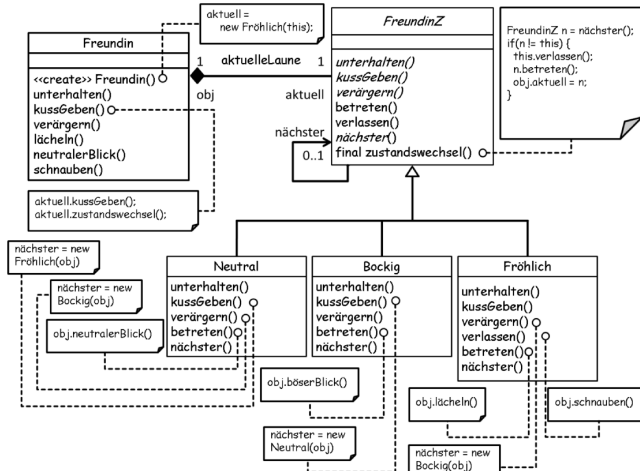
```
30 public class B {  
31     private C c;  
32  
33     public B(C c) {  
34         this.c = c;  
35     }  
36  
37     public void u() { }  
38  
39     public void v() {  
40         this.u();  
41     }  
42  
43     public void w() {  
44         c.x();  
45     }  
46 }
```

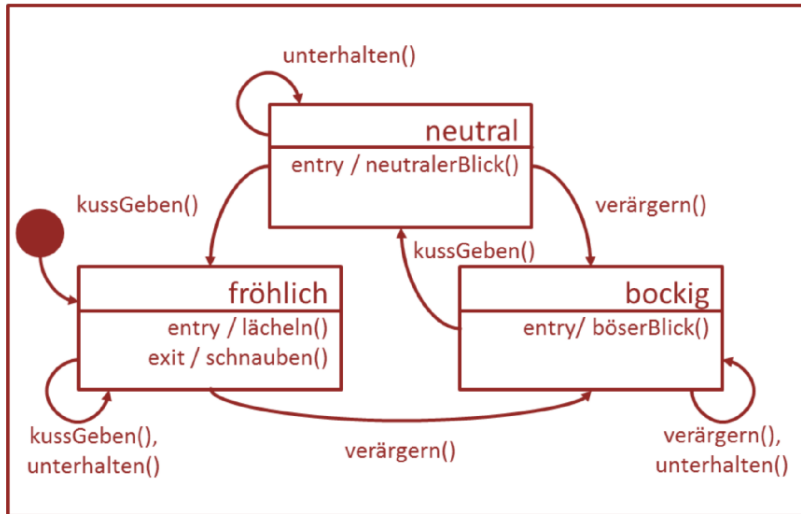
```
47 public class C {  
48     private A a;  
49     private B b1, b2;  
50     private B lastCall;  
51  
52     public C(A a) {  
53         this.a = a;  
54         b1 = new B(this);  
55         b2 = new B(this);  
56     }  
57  
58     public void x() {  
59         if (b1 == lastCall) {  
60             b2.u();  
61         } else {  
62             b1.u();  
63         }  
64     }  
65  
66     public void y() {  
67         lastCall = b1;  
68         b1.w();  
69         a.s();  
70         lastCall = b2;  
71         b2.w();  
72     }  
73  
74     public void z() {  
75         b1.v();  
76         b2.v();  
77     }  
78 }
```



# Zustandsdiagramm

Aufgabe: Zeichnet zu folgendem Klassendiagramm ein Zustandsdiagramm.





Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. (**DON**) den ausgefüllten Bogen (**HAB**) dem Sachbearbeiter (**RECP**) zurückgegeben hat, ...

Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. (**DON**) den ausgefüllten Bogen (**HAB**) dem Sachbearbeiter (**RECP**) zurückgegeben hat, ...

... wird er vom Arzt untersucht.



Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. (**DON**) den ausgefüllten Bogen (**HAB**) dem Sachbearbeiter (**RECP**) zurückgegeben hat, ...

... wird er vom Arzt untersucht.

... wird er (**PAT**) vom Arzt (**AG**) untersucht (**ACT**).

Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. (**DON**) den ausgefüllten Bogen (**HAB**) dem Sachbearbeiter (**RECP**) zurückgegeben hat, ...

... wird er vom Arzt untersucht.

... wird er (**PAT**) vom Arzt (**AG**) untersucht (**ACT**).

Der Arzt führt bei Bernd B. eine Auskultation mit einem Stethoskop durch.

Nachdem Bernd B. den ausgefüllten Bogen dem Sachbearbeiter zurückgegeben hat, ...

Nachdem Bernd B. (**DON**) den ausgefüllten Bogen (**HAB**) dem Sachbearbeiter (**RECP**) zurückgegeben hat, ...

... wird er vom Arzt untersucht.

... wird er (**PAT**) vom Arzt (**AG**) untersucht (**ACT**).

Der Arzt führt bei Bernd B. eine Auskultation mit einem Stethoskop durch.  
Der Arzt (**AG**) führt (**ACT**) bei Bernd B. (**PAT**) eine Auskultation (**ACT**) mit einem Stethoskop (**INSTR**) durch (**ACT**).

# Ende

Fragen?