

Softwaretechnik 1 - 1. Tutorium

Tutorium 03

Felix Bachmann | 15.05.2017

KIT - INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION (IPD)



- 1 Orga
- 2 Wasserfallmodell
- 3 Durchführbarkeitsuntersuchung
- 4 Lastenheft
- 5 Pflichtenheft
- 6 UML-Klassendiagramm
- 7 \LaTeX
- 8 Tipps

Keine Lösungen ins Forum schreiben!



Ausführbare jar datei

Beitrag zensiert, Bemerkung des Moderators:

Bitte keine Lösungen direkt schreiben!!



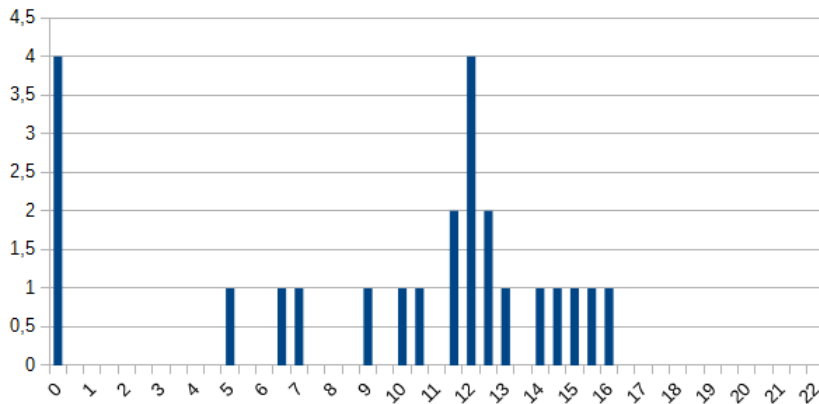
Ausführbare jar datei

Beitrag zensiert, Bemerkung des Moderators:

Keine Lösungen direkt schreiben!!

1. Übungsblatt Statistik

$n = 24$



Allgemein

generell ohne Abzug:

- gleiche Abgabe bei allen Aufgaben

generell mit Abzug: (bis zu -2P)

- CheckStyle nicht beachtet
- JavaDoc nicht vollständig / nicht aussagekräftig
- zu wenige commits / nicht aussagekräftige commit-messages

Aufgabe 1 (Altsoftware vorbereiten)

- *.properties falsch / nicht verschoben (ist Ressource!)
- in src.xml wurden *.launch-Dateien nicht hinzugefügt

Aufgabe 2 + 3 (Modultests + Testüberdeckung)

- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)

Aufgabe 2 + 3 (Modultests + Testüberdeckung)

- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)
- equals() reicht nicht aus, um Gleichheit der Bilder zu prüfen

Aufgabe 2 + 3 (Modultests + Testüberdeckung)

- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)
- equals() reicht nicht aus, um Gleichheit der Bilder zu prüfen
- new File() erstellt kein File, sondern nur einen “pointer” auf einen Pfad (siehe File.createNewFile() oder File.mkdir())

Aufgabe 2 + 3 (Modultests + Testüberdeckung)

- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)
- equals() reicht nicht aus, um Gleichheit der Bilder zu prüfen
- new File() erstellt kein File, sondern nur einen “pointer” auf einen Pfad (siehe File.createNewFile() oder File.mkdir())
- benutzt relative Pfade (beginnen im jmjrst.main-Ordner)

Aufgabe 2 + 3 (Modultests + Testüberdeckung)

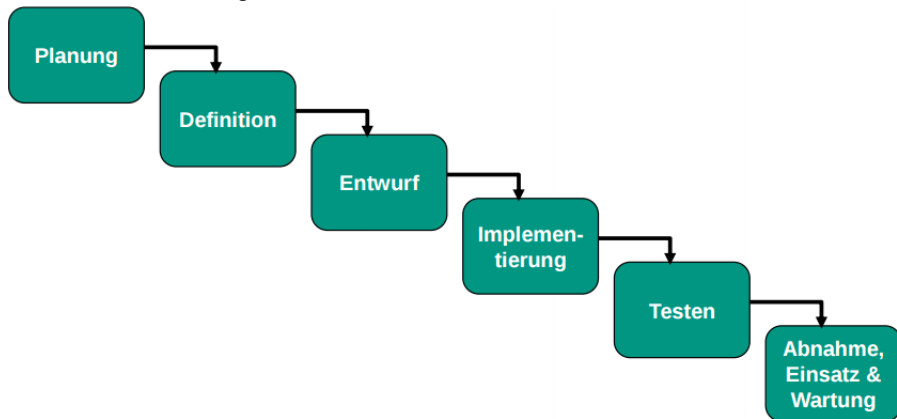
- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)
- equals() reicht nicht aus, um Gleichheit der Bilder zu prüfen
- new File() erstellt kein File, sondern nur einen “pointer” auf einen Pfad (siehe File.createNewFile() oder File.mkdir())
- benutzt relative Pfade (beginnen im jmjrst.main-Ordner)
- Testklasse in gleiches Paket wie zu testenden Klasse

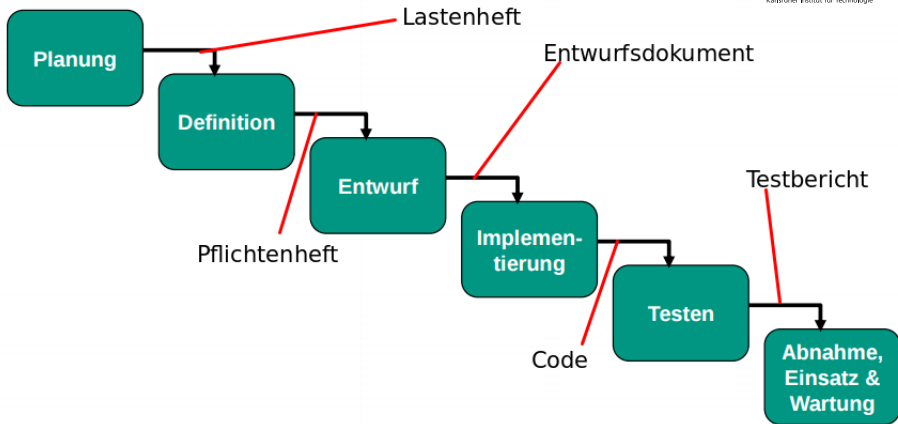
Aufgabe 2 + 3 (Modultests + Testüberdeckung)

- auch bei Drehung um 0° ist Überprüfung des Bildes nötig (Dimensionen + Pixel)
- equals() reicht nicht aus, um Gleichheit der Bilder zu prüfen
- new File() erstellt kein File, sondern nur einen “pointer” auf einen Pfad (siehe File.createNewFile() oder File.mkdir())
- benutzt relative Pfade (beginnen im jmjrst.main-Ordner)
- Testklasse in gleiches Paket wie zu testenden Klasse
- fügt Abhängigkeiten in die jmjrst.main-pom.xml ein, **nicht** in die von iMage

■ Was ist das?

dokumentengetriebenes Prozessmodell, das die (möglichen) Phasen der Softwareentwicklung beschreibt





für das 2. ÜB:

- Lastenheft
- Durchführbarkeitsuntersuchung (weiteres Artefakt der Planung)

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

1 Fachlich

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- ① Fachlich
- ② Alternativen

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- 1 Fachlich
- 2 Alternativen
- 3 Personell

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- 1 Fachlich
- 2 Alternativen
- 3 Personell
- 4 Risiken

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- 1 Fachlich
- 2 Alternativen
- 3 Personell
- 4 Risiken
- 5 Ökonomisch

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- 1 Fachlich
- 2 Alternativen
- 3 Personell
- 4 Risiken
- 5 Ökonomisch
- 6 Rechtlich

Grundlegende Frage

Ist das Projekt in dem jeweiligen Szenario überhaupt durchführbar?

- 1 Fachlich
- 2 Alternativen
- 3 Personell
- 4 Risiken
- 5 Ökonomisch
- 6 Rechtlich

Fürs Übungsblatt

Denkt euch was (logisches) aus!

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

1 Zielbestimmung

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- 1 Zielbestimmung
- 2 Produkteinsatz

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- 1 Zielbestimmung
- 2 Produkteinsatz
- 3 Funktionale Anforderungen

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- 1 Zielbestimmung
- 2 Produkteinsatz
- 3 Funktionale Anforderungen
- 4 Produktdaten

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- 1 Zielbestimmung
- 2 Produkteinsatz
- 3 Funktionale Anforderungen
- 4 Produktdaten
- 5 Nichtfunktionale Anforderungen

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- ① Zielbestimmung
- ② Produkteinsatz
- ③ Funktionale Anforderungen
- ④ Produktdaten
- ⑤ Nichtfunktionale Anforderungen
- ⑥ Systemmodelle
 - Szenarien
 - Anwendungsfälle

Grundlegende Aufgabe

Das Lastenheft sammelt die Anforderungen des Auftraggebers an den Auftragnehmer.

- ① Zielbestimmung
- ② Produkteinsatz
- ③ Funktionale Anforderungen
- ④ Produktdaten
- ⑤ Nichtfunktionale Anforderungen
- ⑥ Systemmodelle
 - Szenarien
 - Anwendungsfälle
- ⑦ Glossar

Zielbestimmung vs. Funktionale Anforderungen

Zielbestimmung vs. Funktionale Anforderungen

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Funktionale Anforderungen: konkrete Auflistung von Funktionen

Zielbestimmung vs. Funktionale Anforderungen

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Funktionale Anforderungen: konkrete Auflistung von Funktionen

Funktionale Anforderungen vs. Nichtfunktionale Anforderungen

Zielbestimmung vs. Funktionale Anforderungen

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Funktionale Anforderungen: konkrete Auflistung von Funktionen

Funktionale Anforderungen vs. Nichtfunktionale Anforderungen

- Funktionale Anforderungen: Funktionen des Produkts
- Nichtfunktionale Anforderungen: "Meta"-Eigenschaften des Produkts

Zielbestimmung vs. Funktionale Anforderungen

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Funktionale Anforderungen: konkrete Auflistung von Funktionen

Funktionale Anforderungen vs. Nichtfunktionale Anforderungen

- Funktionale Anforderungen: Funktionen des Produkts
- Nichtfunktionale Anforderungen: "Meta"-Eigenschaften des Produkts

Zielbestimmung vs. Produkteinsatz

Zielbestimmung vs. Funktionale Anforderungen

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Funktionale Anforderungen: konkrete Auflistung von Funktionen

Funktionale Anforderungen vs. Nichtfunktionale Anforderungen

- Funktionale Anforderungen: Funktionen des Produkts
- Nichtfunktionale Anforderungen: "Meta"-Eigenschaften des Produkts

Zielbestimmung vs. Produkteinsatz

- Zielbestimmung: allgemeine Beschreibung, was das Produkt können soll
- Produkteinsatz: Rahmenbedingungen (Zielgruppe, Anwendungsbereiche)

Grundlegende Aufgabe

Erweiterung des Lastenheftes, sodass exakt abgebildet ist **was** (noch nicht **wie**) zu implementieren ist.

- ① Zielbestimmung
- ② Produkteinsatz
- ③ **Produktumgebung**
- ④ Funktionale Anforderungen
- ⑤ Produktdaten
- ⑥ Nichtfunktionale Anforderungen
- ⑦ **Globale Testfälle**
- ⑧ Systemmodelle
 - Szenarien
 - Anwendungsfälle
 - **Objektmodelle** \implies UML-Klassendiagramme (heute)
 - **Dynamische Modelle** \implies nächstes Mal
 - **Benutzerschnittstelle** \implies Zeichnungen/Screenshots
- ⑨ Glossar

Produkteinsatz vs. Produktumgebung

Produkteinsatz vs. Produktumgebung

- Produkteinsatz: Rahmenbedingungen (Zielgruppe, Anwendungsbereiche)
- Produktumgebung: Rahmenbedingungen bzgl. Software/Hardware

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes.

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase.

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte.

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte. falsch

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte. falsch
- Das Pflichtenheft beschreibt nur, was zu implementieren ist und nicht wie.

Wahr oder falsch?

- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte. falsch
- Das Pflichtenheft beschreibt nur, was zu implementieren ist und nicht wie. wahr

Wahr oder falsch?

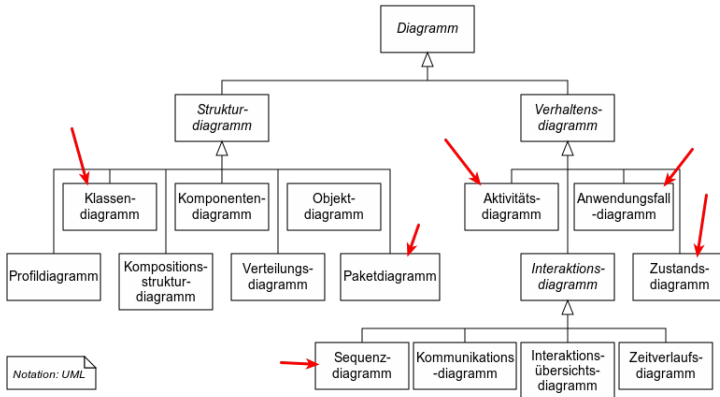
- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte. falsch
- Das Pflichtenheft beschreibt nur, was zu implementieren ist und nicht wie. wahr
- Nicht-funktionale Anforderungen sind sowohl Teil des Pflichtenhefts als auch des Lastenhefts.

Wahr oder falsch?

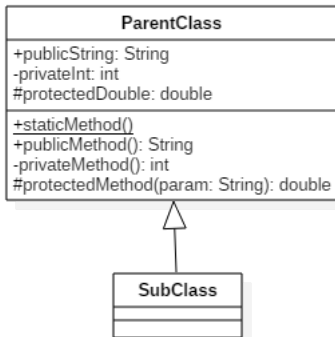
- Das Lastenheft ist eine Verfeinerung des Pflichtenheftes. falsch
- Das Lastenheft ist das Ergebnis der Planungsphase. wahr
- Nicht-funktionale Eigenschaften beschreiben, was das Produkt nicht tun sollte. falsch
- Das Pflichtenheft beschreibt nur, was zu implementieren ist und nicht wie. wahr
- Nicht-funktionale Anforderungen sind sowohl Teil des Pflichtenhefts als auch des Lastenhefts. wahr

UML? Kann man das essen?

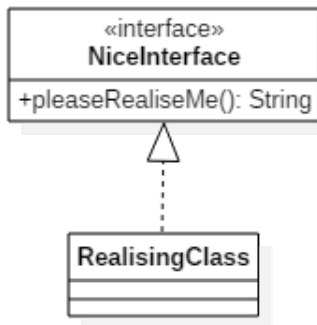
- UML = **U**nified **M**odeling **L**anguage
- grafische Modellierungssprache, strenge Syntax



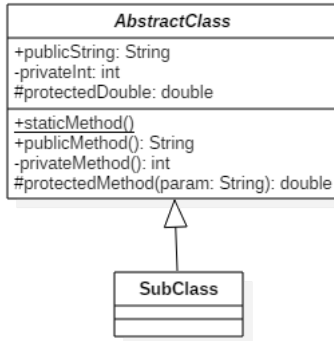
UML-Klassendiagramm: Klassen + Vererbung



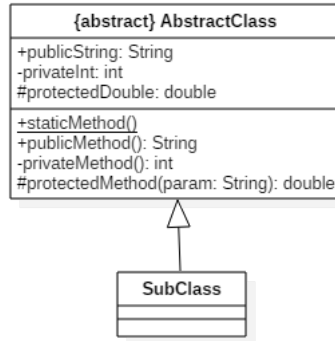
- - ist private: von Exemplaren derselben Klasse sichtbar (aber von allen!)



UML-Klassendiagramm: Abstrakte Klassen



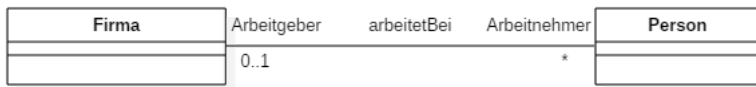
(a) UML-Notation



(b) Notation für Abgaben

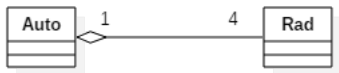


(c) so nicht,...

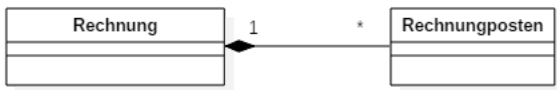


(d) ..., sondern so!

UML-Klassendiagramm: Aggregation + Komposition



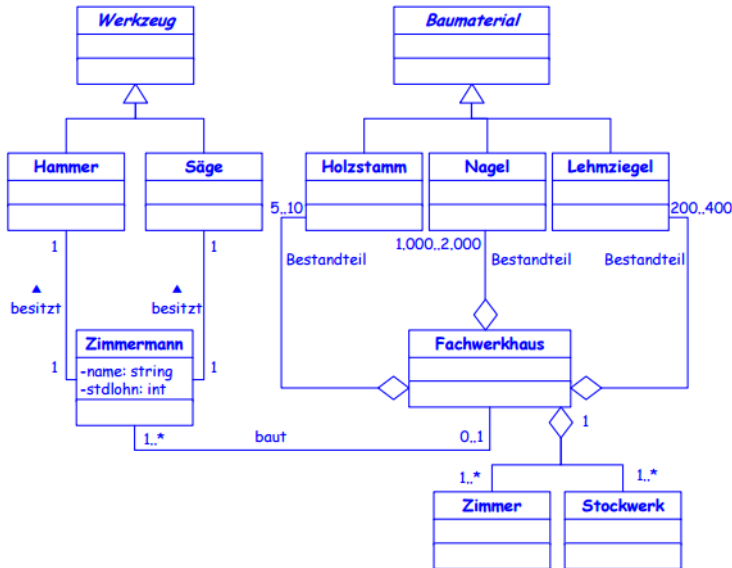
(e) Aggregation (Teil-Ganzes-Beziehung)



(f) Komposition (Aggregation, Teil kann ohne Ganzes nicht existieren)

Modellieren Sie das Szenario möglichst vollständig als UML-Klassendiagramm. Modellieren Sie keine Methoden. Geben Sie Attribute, Multiplizitäten, Restriktionen, Assoziationsnamen sowie Rollen an.

Ein Fachwerkhaus besteht aus 5 bis 10 Holzstämmen, 200 bis 400 Lehmziegeln sowie 1.000 bis 2.000 Nägeln. Jedes Baumaterial, egal ob Holzstamm, Lehmziegel oder Nagel, ist Bestandteil in genau einem Fachwerkhaus. Jedes Fachwerkhaus hat eine bestimmte Anzahl an Zimmern und Stockwerken. Für den Bau eines Fachwerkhauses ist mindestens ein Zimmermann zuständig, welcher einen Namen sowie einen individuellen Stundenlohn besitzt. Zum Bau des Fachwerkhauses verwendet jeder Zimmermann sein eigenes Werkzeug, bestehend aus genau einem Hammer sowie genau einer Säge. Jeder Zimmermann kann an maximal einem Fachwerkhaus gleichzeitig bauen.



- auf dem Blatt müsst ihr L^AT_EX für die Dokumente benutzen
 - ⇒ verträgt sich besser mit git als z.B. LibreOffice o.Ä.
 - ⇒ wird euch an der Uni immer wieder begegnen (leichtes Setzen von Formeln etc.)
- nicht wie z.B. Word WYSIWYG, sondern WYSIWYAF / WYSIWYM
- Vorteile:
 - gut versionierbar
 - leicht Formeln erstellbar
 - nach Eingewöhnung recht intuitiv
 - multifunktional (Dokumente, Präsentationen, ...)
- Nachteile:
 - Einarbeitung notwendig :(

Installation einer Distribution notwendig, z.B.:

- MikTeX für Windows
- TeX Live für Linux, Mac, Windows

Editoren machen das Schreiben von L^AT_EX-Dokumenten angenehmer

- Texmaker
- TeXstudio (erweiterter Texmaker, mein Favorit)
- TeXclipse (Plugin für Eclipse)
- ...

L^AT_EX: Basic Document Structure

```
\documentclass[option1, option2, ...]{class}  
...  
\usepackage{package1}  
\usepackage[options]{package2}  
\usepackage{package3, package4}  
\usepackage[options]{package5, package6}  
...  
...  
begin{document}  
...  
...  
...  
\end{document}
```

The preamble

The content

Aufgabe 1 + 3: Lastenheft + Durchführbarkeitsuntersuchung

- lasst euch was (sinnvolles) einfallen
- benutzt \LaTeX

Aufgabe 1 + 3: Lastenheft + Durchführbarkeitsuntersuchung

- lasst euch was (sinnvolles) einfallen
- benutzt \LaTeX

Aufgabe 2: Klassendiagramme

- achtet auf Schlüsselwörter (“ist ein“, “enthält ein“, “besteht aus“, ...)

Aufgabe 1 + 3: Lastenheft + Durchführbarkeitsuntersuchung

- lasst euch was (sinnvolles) einfallen
- benutzt \LaTeX

Aufgabe 2: Klassendiagramme

- achtet auf Schlüsselwörter (“ist ein“, “enthält ein“, “besteht aus“, ...)

Aufgabe 4 + 5: Geometrify + cmd-Programm

- an einigen Stellen sind Aufgaben etwas vage
 \implies überlegt euch, was Sinn macht

Abgabe

- Deadline am 24.5 um 12:00
- Dokumente ausdrucken
- Klassendiagramme handschriftlich

Bis dann! (dann := 29.05.17)



DEVELOPERS