

# ZETA

## Teamprojekt WS20/21 - SS21 HTWG Konstanz

Kevin Olomu MSI SE (Matnr.: 301338)

Jonas Weber MSI SE (Matnr.: 301901)

Jonas Wirth MSI SE (Matnr.: 301900)

Sebastian Reinbold MSI SE (Matnr.: 295401)

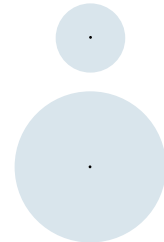
Tobias Spott MSI ITM (Matnr.: 289259)

Nicolai Stephan MSI SE (Matnr.: 293343)

Prof. Dr. Marko Boger

# Agenda

- **Live-Demo**
- **1. Semester**
  - Projektstand
  - Ziele
  - Aufgaben
  - Probleme
  - Live-Demo
  - Erreichte Ziele
- **2. Semester**
  - Projektstand
  - Ziele
  - Aufgaben
  - Probleme
  - Live-Demo
  - Erreichte Ziele
- **Offene Aufgaben**
- **Erledigte Aufgaben im Detail**
- **Sprintauswertung im Detail**

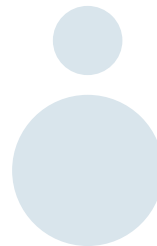


# Live-Demo

# Projektstand 1. Semester

## GITHUB

- Wiki: Unvollständiger Developer-Guide
- Wiki: Veraltete Inhalte (nicht mehr aktuelle GIFs / Beschreibungen)
- Branches: Viele veraltete Branches
- Issues: Viele veraltete, nicht beschriebene Issues



# Projektstand 1. Semester

## FRONTEND

- Oberflächen größtenteils auf Bootstrap bzw. plain HTML/JavaScript
- Viel ungenutzter Programmcode
- Viel ungenutztes CSS
- “Spaghetti-Code”
- Viele Duplikate
- Projektmappe unaufgeräumt
- Viele Fehler und Probleme



# Projektstand 1. Semester

## BACKEND

- Backend soweit funktionsfähig
- Projektmappe aufgeräumt
- Benötigte Features vorhanden / implementiert

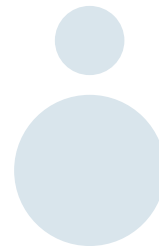
# Projektstand 1. Semester

## BUILD-PIPELINE

- Inkonsistent: Liefert teilweise nicht-nachvollziehbare Fehler

# Ziele 1. Semester

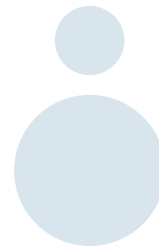
- Projekt lokal aufsetzen
- Wiki-Anleitungen vervollständigen (Developer-Guide)
- Kleine Aufgaben vom alten Projektteam übernehmen
- In das Projekt einfinden
- Vollständige Übernahme vorbereiten





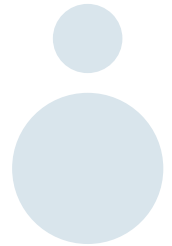
# Aufgaben 1. Semester

- Projektmappe lokal zum Laufen bringen
  - Docker
  - MongoDB
  - Frontend
  - Backend
  - Entwicklungsumgebung
- Developer-Guide aktualisiert
- Frontend: Projektmappe soweit möglich aufgeräumt
- Frontend: Alle Views auf Vuetify umgebaut
  - ca. 30 Views



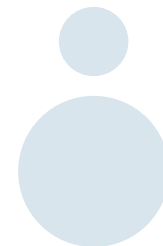
# Aufgaben 1. Semester

- Frontend: Duplikate soweit möglich entfernt
- Frontend: Spaghetti-Code soweit möglich entwirrt
- Frontend: Mehrere Bugs gefixt
- Sukzessive Einführung durch das alte Projektteam, um Projekt zu übernehmen
  - Coding-Sessions
  - Projekt-Sessions
  - Teilnahme an Sprint-Meetings



# Probleme 1. Semester

- Viele Fehler / Probleme beim Aufsetzen der Projektmappe unbekannt
  - Lokales Aufsetzen hat viel Zeit benötigt
  - Probleme konnten oft nur in größerer Runde / mit Hilfe des alten Teams gelöst werden
- Passende Einstiegsaufgaben finden
  - Viele Aufgaben zu umfangreich / zu kompliziert für Einstieg
- Parallel zur Vuetify Umstellung wurden neue Bootstrap-Views erstellt
  - Viel Nacharbeitungsaufwand
  - Konflikte beim Zusammenführen



# Live-Demo

# Erreichte Ziele 1. Semester

- ~~Projekt lokal aufsetzen~~
- ~~Wiki-Anleitungen vervollständigen (Developer Guide)~~
- ~~Kleine Aufgaben vom alten Projektteam übernehmen~~
- ~~In das Projekt einfinden~~
- ~~Vollständige Übernahme vorbereiten~~

# Projektstand 2. Semester

## GITHUB

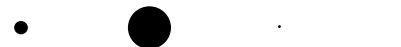
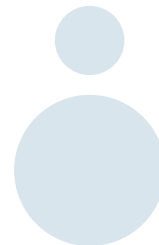
- Wiki: Veraltete Inhalte (nicht mehr aktuelle GIFs / Beschreibungen)
- Branches: Viele veraltete Branches



# Projektstand 2. Semester

## FRONTEND

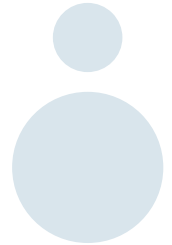
- Oberflächen vollständig in Vuetify
- “Spaghetti-Code”
- Viele Duplikate
- Projektmappe (teilweise) unaufgeräumt
- Viele Fehler und Probleme bei den vorhandenen Features
- Viele vorhandene Features sind nicht-intuitiv



# Projektstand 2. Semester

## BACKEND

- Backend soweit funktionsfähig
- Projektmappe aufgeräumt
- Benötigte Features nahezu vollständig vorhanden / implementiert





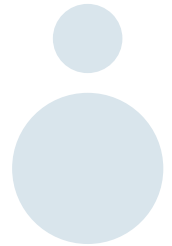
# Projektstand 2. Semester

## BUILD-PIPELINE

- Inkonsistent: Liefert teilweise nicht-nachvollziehbare Fehler

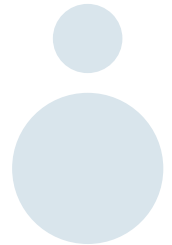
# Ziele 2. Semester

- Umzug von Github auf Jira
- Fehler / Bugs / Probleme der vorhandenen Features lösen
- Intuitivität der Oberfläche verbessern
- Model-Editor ausbauen
- GitHub Wiki mit aktuellen Beispielen versehen



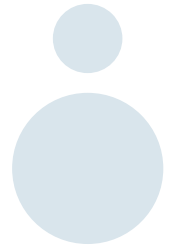
# Aufgaben 2. Semester

- Umzug von GitHub auf Jira
  - Jira angepasst
  - GitHub aufgeräumt
- Hauptmenü überarbeitet
  - ModellInstance-Filter gefixt
  - Sidebar öffnet sich fehlerhafterweise
  - Stepper-Fortschritt
- Stepper überarbeitet
  - Bugs gefixt (z.B. Ausleseprobleme)
  - UI und Funktionsweise angepasst
  - Aufbau in der Projektmappe angepasst
- ConceptEditor überarbeitet
  - Speicherung ConceptEditor
  - Leerer ConceptEditor bei neuem Projekt
  - Labels bearbeitbar und Synchronisierung Labels / PropertyPanel
  - Inline Editing gefixt



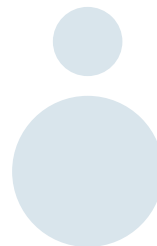
# Aufgaben 2. Semester

- ModelEditor überarbeitet
  - Einheitliches Design zum ConceptEditor
  - PropertyPanel fix
  - Unterschiedliche BugFixes (z.B. Edges falsch behandelt, Unterschiedliche Benennung Methods / Operations, ...)
- ModelEditor erweitert
  - Attribute und Methoden bearbeitbar in PropertyPanel
  - Metainformationen read-only in PropertyPanel
  - Nodedefinition über Shape / Style / Diagram umgesetzt (z.B. Schriftart, -größe, Farbe, Form)
  - Kantvalidierung
  - Kantendefinition über Shape / Style / Diagram umgesetzt
  - Labels
  - Gruppierungen
  - Nodes resizable



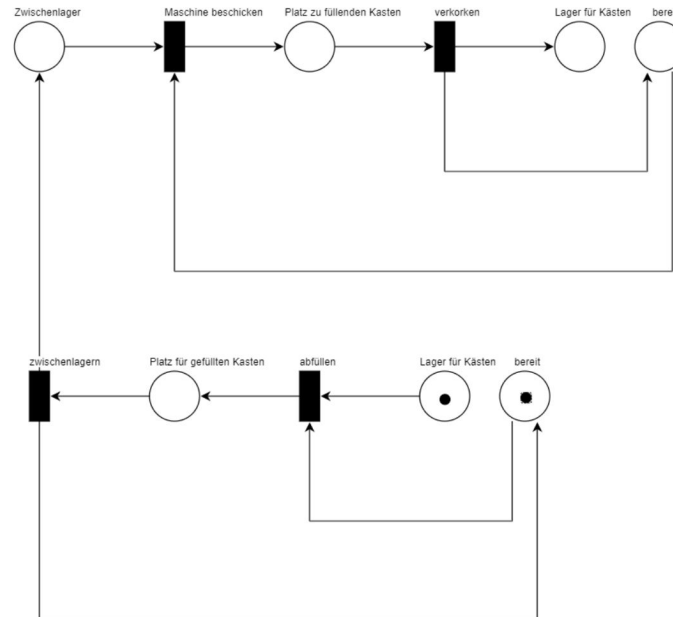
# Aufgaben 2. Semester

- Projektmappe weiter aufgeräumt
  - Nicht-verwendete Funktionen / CSS / usw. entfernt
  - Dateien / Komponenten neu angeordnet
  - Update yFiles auf 2.3
- CI gefixt



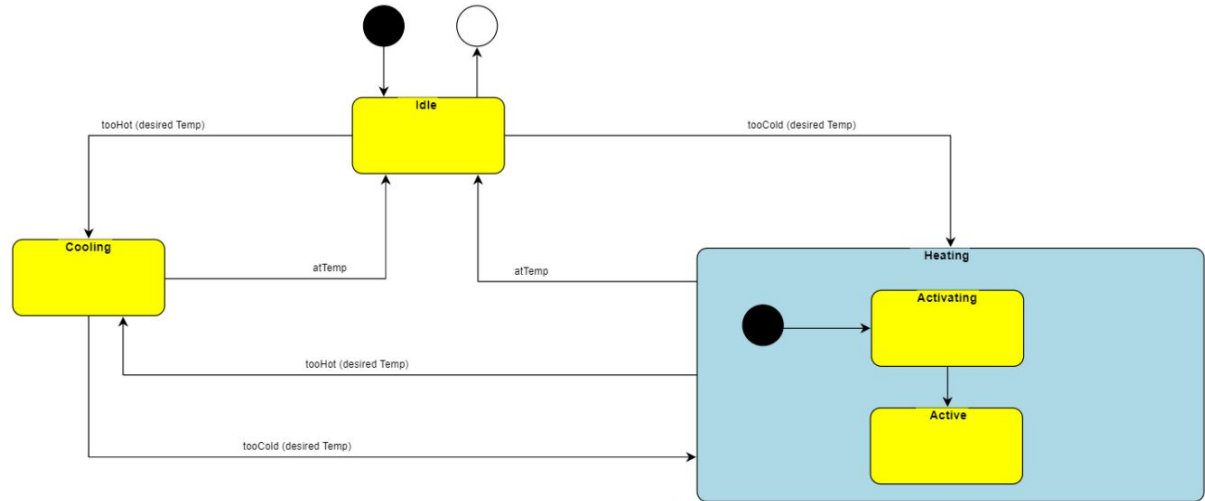
# Aufgaben 2. Semester

- Beispielprojekte
  - Petri-Netz



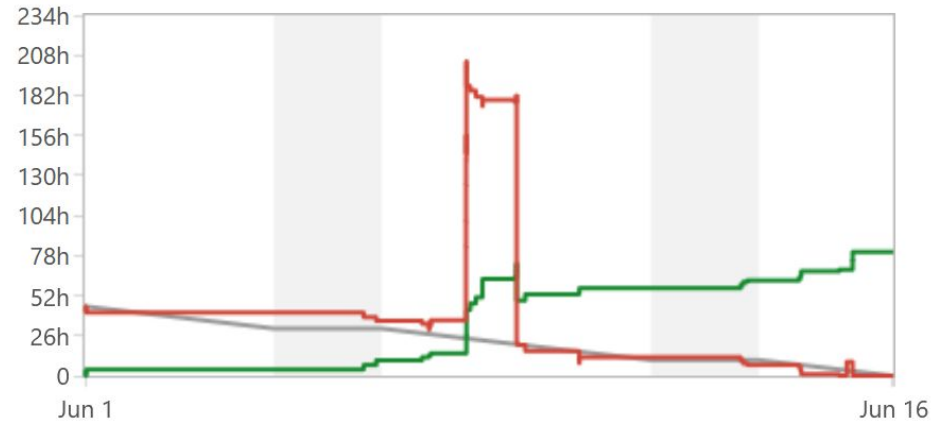
# Aufgaben 2. Semester

- Beispielprojekte
  - Zustandsdiagramm



# Probleme 2. Semester

- Viele Probleme in den bestehenden Komponenten verzögern Ausbau des ModelEditors
- Viel Arbeitszeit für Buganalyse und -behebung
- Viel Arbeitszeit für Ausbau der bestehenden Komponenten





# Live-Demo

# Erreichte Ziele 2. Semester

- ~~Umzug von Github auf Jira~~
- ~~Fehler / Bugs / Probleme der vorhandenen Features lösen~~
- ~~Intuitivität der Oberfläche verbessern~~
- ~~Model Editor ausbauen~~
- ~~GitHub Wiki mit aktuellen Beispielen versehen~~



# Offene Aufgaben

## MODEL-EDITOR SPEICHERUNG

- Backend-Anpassung
- Frontend-Anpassung

## DYNAMISCHE ATTRIBUTE

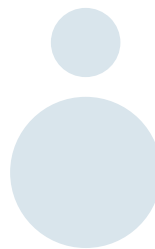
- Concept-Editor: Dynamische Attribute müssen eingebaut werden können
- DSLs: Dynamische Attribute müssen beschrieben werden können
- Model-Editor: Dynamische Attribute müssen verwendet werden können
- Backend: Vermutlich noch nicht implementiert

## READ-ONLY LABELS (MODEL-EDITOR)

- Concept-Editor und DSLs: Funktionen bereits vollständig
- Model-Editor: Labels sind aktuell immer bearbeitbar (Einschränkung durch yFiles)

## ROTIERBARKEIT LABELS (MODEL-EDITOR)

- Werden automatisch von yFiles rotiert
- DSLs: Rotierbarkeit implementieren
- Model-Editor: Rotierung darstellen



# Offene Aufgaben

## VARIABLENTYPEN

- Concept-Editor / DSLs: Variablentypen festlegen können
- Model-Editor: Eingaben anhand Variablentyp validieren
- Backend: Vorhanden

## METHODEN

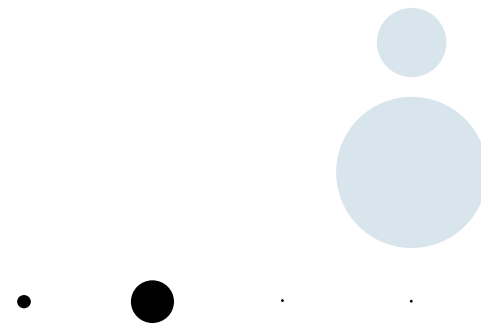
- Concept-Editor / DSLs: Methoden Aufbau festlegen können (Rückgabotyp, Methodenrumpf, ...)
- Model-Editor: Methoden anwenden
- Backend: teilweise vorhanden

## BUTTONS ALS GEO-ELEMENT (MODEL-EDITOR)

- Concept-Editor / DSLs: Buttons definieren können
- Model-Editor: Buttons anzeigen und Buttonklick auswerten
- Backend: Nicht vorhanden

## KOMPLEXE VIEW-ELEMENTE ALS GEO-ELEMENT (MODEL-EDITOR)

- Zum Beispiel: DropDown, Tabs, Gruppierungen, ...
- Concept-Editor / DSLs: Elemente definieren können
- Model-Editor: Elemente anzeigen und anwenden können
- Backend: Nicht vorhanden



# Offene Aufgaben

## FEHLENDE EDGES (z.B. REALISATION-EDGE) (CONCEPT-EDITOR)

- Concept-Editor: Aktuell auskommentiert. Funktionieren nicht vollständig.
- Backend: Nicht vorhanden

## BEZIEHUNGEN ZWISCHEN NODES (1 : 1, n : m, etc.)

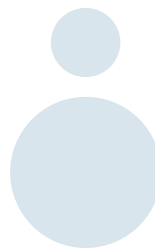
- Concept-Editor: Definieren können
- Model-Editor: Anwenden
- Backend: Nicht vorhanden

## MEHRERE LABELS FÜR EINE EDGE (CONCEPT-EDITOR)

- Concept-Editor: Anzeige, Bearbeitbarkeit, Speicherung
- Backend: Vermutlich vorhanden

## EDGES OHNE LABELS (CONCEPT-EDITOR)

- Concept-Editor: Speicherung, ID generieren
- Backend: Wird aktuell als ID verwendet. Problem



# Offene Aufgaben

## IS INTERFACE

- Concept-Editor: Für nodes festlegen können
- Model-Editor: Entsprechend darauf reagieren können?
- Backend: Vorhanden

## IS ABSTRACT (MODEL-EDITOR)

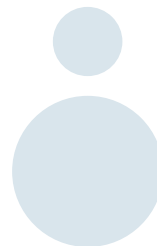
- Im ConceptEditor können nodes als Abstract definiert werden
- Model-Editor: Entsprechend darauf reagieren können?

## UNTERSCHIEDLICHE EDGE-TYPEN (MODEL-EDITOR)

- Im ConceptEditor können unterschiedliche Edge-Typen verwendet werden
- Model-Editor: Entsprechend darauf reagieren können?

## FEHLERBEHANDLUNG CODE-EDITOR (STEPPER)

- Anzeige von Fehlern in den DSLs extrem verbuggt
- Teilweise mehrfache Anzeige eines Fehlers
- Teilweise Anzeige von Fehlern anderer Projekte



# Offene Aufgaben

## STATISCHE TEXTE (MODEL-EDITOR)

- Zum Beispiel: Klassennamen, ...
- Concept-Editor / DSLs: Implementiert
- Model-Editor: Anzeigen können (read-only) der statischen Texte

## E-MAIL-VERSAND

- Backend: Funktioniert nicht
- Es werden aktuell keine E-Mails versendet

## PASSWORT VERGESSEN

- Workflow anpassen
- Aufgabe stammt noch vom alten Team

## HOSTING AUF ÖFFENTLICHEM SERVER

- Bisher nicht umgesetzt



# Erledigte Aufgaben im Detail

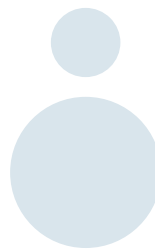
- Developer-Guide aktualisiert
  - Mögliche Probleme ermittelt und dokumentiert (inklusive Lösung und Bilder)
  - Mögliche Fehler ermittelt und dokumentiert (inklusive Lösung und Bilder)
  - Veraltete Inhalte entfernt
- Frontend: Projektmappe soweit möglich aufgeräumt
  - Dialogfenster ausgelagert
  - Overview-Folder aufgeräumt
  - Stepper-Komponenten sinnvoll angeordnet und verschoben
- Frontend: Alle Views auf Vuetify umgebaut
  - ca. 30 Views
  - Bootstrap-Komponenten entfernt
  - Vuetify-Komponenten eingebaut
  - JavaScript angepasst
  - CSS angepasst



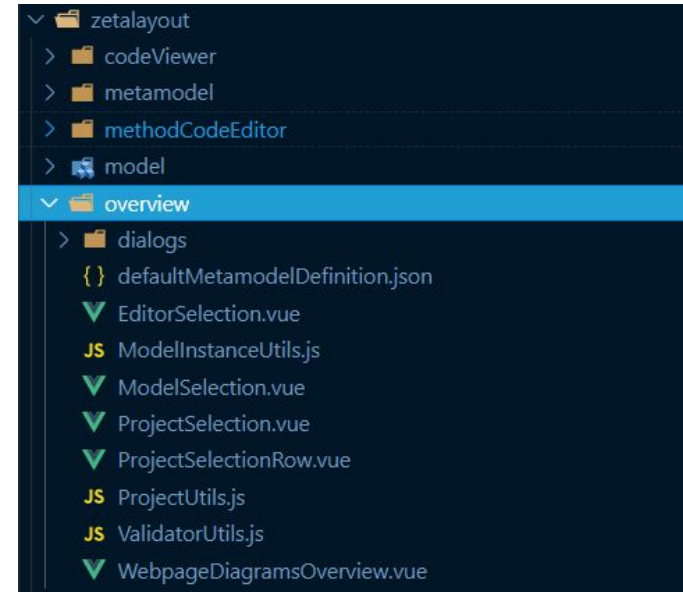
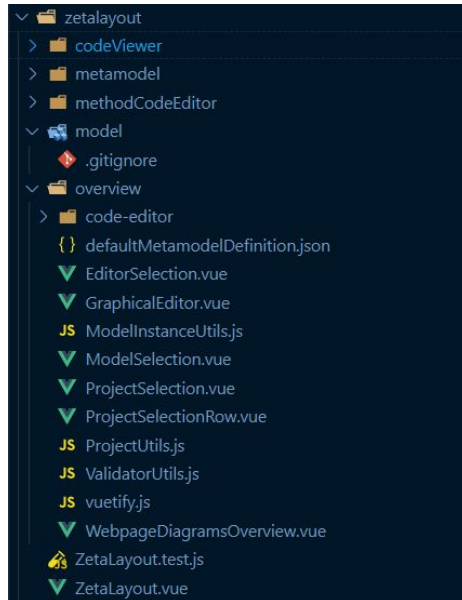


# Erledigte Aufgaben im Detail

- Frontend: Duplikate soweit wie möglich entfernt
  - Nicht-verwendete CSS-Klassen entfernt
  - Nicht-verwendete Javascript-Funktionen entfernt
  - Nicht-verwendete Dateien entfernt
  - Nicht-verwendete View-Komponenten entfernt
- Frontend: Spaghetti-Code soweit wie möglich entwirrt
  - Stepper-Komponente sinnvoll angeordnet
  - Komponenten ausgelagert / zusammengezogen
  - Dateien sortiert / verschoben
- Frontend: Eingabefelder nach Submit leeren
  - Bug gefixt
  - Submit leert die Eingabefelder, nachdem die Eingabe verarbeitet wurde
- Frontend: Bug: Enter Taste bei neuem Modell fehlerhaft
  - Bug gefixt
  - Enter-Taste bestätigt Model-Name
  - Model wird in Liste eingefügt



# Erledigte Aufgaben im Detail

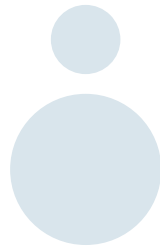


# Erledigte Aufgaben im Detail



# Erledigte Aufgaben im Detail

- Umzug von GitHub auf Jira
  - Jira-Tasks anlegen
  - In Jira einarbeiten / einlesen
  - Jira-Einstellungen anpassen
- GitHub aufräumen
  - Github-Issues prüfen und ggf. nach Jira überführen
  - Github-Issues abschließen
  - Github-Projekte aufräumen (Issues verarbeitet)
  - Github-Projekte abschließen
  - Github-Banches prüfen und ggf. abschließen
- ModelEditor einheitliches Design zum ConceptEditor
  - Toolbar zu dominant
  - Toolbar-Checkboxes / Toggle-Buttons nicht einheitlich



# Erledigte Aufgaben im Detail

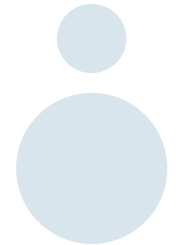
- Überarbeitung Stepper
  - Fortschritt abhängig von DSLs
  - Bug fixing
    - Klick auf Stepper-Item navigiert manchmal nicht
      - Die Haupt-Ansicht aktualisiert sich nicht (nur das Stepper-Element)
      - Reproduktion: Wenn der ConceptEditor zum ersten Mal angezeigt wird, Klick auf das Style-Stepper-Item.
    - Klick auf Stepper-Item zeigt manchmal einen leeren Bildschirm
      - Inhalt der Elemente wird nicht geladen. Es werden nur Container angezeigt
      - Reproduktion: Wenn der ConceptEditor zum ersten Mal angezeigt wird, Klick auf das Shape-Stepper-Item.
    - "Navigation" durch Klick in einen anderen TextEditor ist nicht intuitiv
      - Linker Anzeigebereich (links neben Shape) wird erst nach einem Tastendruck aktualisiert
      - Stepper-Element wird dabei nicht aktualisiert
      - Highlighting (markierte Zeile) bleibt in den nicht-aktiven TextEditoren erhalten
      - Es ist nicht mehr offensichtlich, in welcher DSL ich mich aktuell befinde
    - Bouncing-Effekt bei Alert-Klick

# Erledigte Aufgaben im Detail

- Überarbeitung Stepper
  - Bug fixing (2)
    - DSL-TextEditoren-Anzeige nicht intuitiv
      - Die einzelnen DSL-TextEditoren wird geladen, wenn auf ein Element navigiert wird
        - » TextEditoren, auf die bisher nicht navigiert wurde, werden nur als leere Hüllen angezeigt
      - TextEditoren werden nach dem erstmaligen Anzeigen weiterhin angezeigt, wenn zwischen den TextEditoren navigiert wird
      - Beim Navigieren zum ConceptEditor wird die Anzeige der TextEditoren manchmal zurückgesetzt (nicht immer)
    - Fehler- und Success-Meldungen werden teilweise mehrfach angezeigt. Reproduzierbar durch:
      - Zwischen ConceptEditor und Shape mehrfach wechseln
      - Fehlerhaften Eintrag in Shape erzeugen (z.B. true durch tru ersetzen)
      - Speichern
      - Es werden mehrere Fehlermeldungen angezeigt, je nachdem, wie oft zwischen ConceptEditor und Shape navigiert wurde
    - Continue Button funktioniert nicht richtig
      - Wird disabled, wenn beim letzten Item angekommen wird
      - Wird meist nicht mehr enabled, wenn zurück navigiert wird
        - » Wird manchmal enabled, wenn zum ConceptEditor navigiert wird (nicht immer)

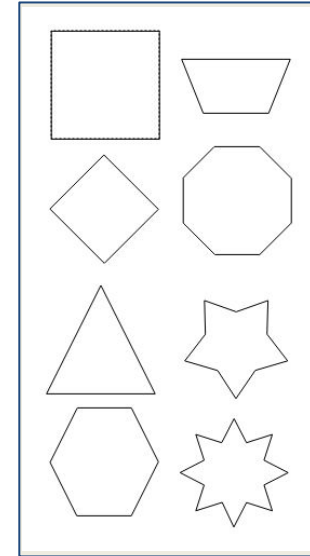
# Erledigte Aufgaben im Detail

- Überarbeitung Stepper
  - Stepper bei hover hervorheben
  - Aktuellen Step hervorheben
  - Bearbeitung der DSLs nur erlauben, wenn aktiv
  - Stepper Navigation durch Klick auf inaktive DSL
- Überarbeitung Hauptmenü
  - Stepper-Fortschritt anhand gespeicherter DSLs anzeigen
  - Filtern der Modell Instanzen pro Projekt / pro User
  - Wenn eine Modell Instanz existiert, wird das Concept für dieses Projekt blockiert
  - Sidebar öffnet sich fehlerhafterweise
- Speicherung ConceptEditor
  - Metamodell-Speicherung
  - Eigenschaften und Kanten
  - Probleme mit Kanten-Attributen und -Operationen
- Problemlösung PropertyPanel ModelEditor
  - PropertyPanel hat Fehler ausgelöst
  - PropertyPanel bei Klick auf Node und auf Edge öffnen



# Erledigte Aufgaben im Detail

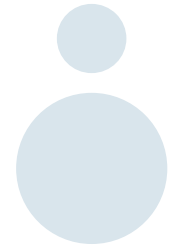
- Attribute und Methoden bearbeitbar machen
  - PropertyPanel im ModelEditor für node und edge
  - PropertyPanel zeigt Attribute und Methoden an
  - Wenn keine Attribute / Methoden definiert sind, wird dieser Bereich nicht angezeigt
  - Fix: Teilweise unterschiedliche Benennung (Methoden und Operationen)
- Shape / Style / Diagramm Attribute für nodes in ModelEditor
  - Neue Shapes eingeführt (Sterne, Trapez, Parallelogramm, ...)
  - Farbe, Größe, ... implementiert
- Shape / Style / Diagramm Attribute für edges in ModelEditor
  - Icons, Pfeildesign, ... implementiert
- Aufbau Petri-Netz Projekt
  - Shape, Style, Diagram aufgebaut
  - Beispiel-Diagramme erstellt
  - In GitHub-Wiki dokumentiert
  - Kantenvvalidierung ModelEditor
    - Nur Kanten zulassen, die im Concept definiert wurden
    - Bei nicht-zulässiger Kante negativ-Symbol anzeigen
    - Bei zulässiger Kante Verbindung aufbauen





# Erledigte Aufgaben im Detail

- Neues Projekt: Leeres Concept
  - Bisher wurde immer ein Beispiel-Concept geladen
  - Leeres Concept hat dann wiederum Probleme ausgelöst, die gefixt werden mussten
  - Programm hat immer ein Concept erwartet
- Metainformationen im PropertyPanel
  - Metainformationen anzeigen, aber nur lesend
  - Für nodes und edges
- CI fix
  - CI liefert seit Teamprojekt-Beginn immer wieder Fehler
  - CI liefert irgendwann nur noch Fehler
  - Bug fix
- ConceptEditor Labels bearbeitbar
  - Label-Inhalte direkt bearbeitbar im Diagramm
  - Synchronisation Kanten-Beschriftung / PropertyPanel
  - Änderung muss in beiden Bereichen (Label und PropertyPanel) ankommen
  - Abhängig vom Toggle Button
  - Bug fix (Attribute und Operationen waren nur einmal änderbar)



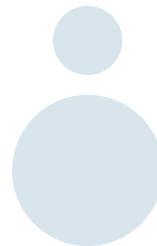
# Erledigte Aufgaben im Detail

- Aufbau Zustandsdiagramm Projekt
  - Shape, Style, Diagram aufgebaut
  - Beispiel-Diagramme erstellt
  - In GitHub-Wiki dokumentiert
- Nodes resizable im ModelEditor
  - Anhand Definition in den DSLs
  - Nur horizontal / nur vertikal / beides
  - Verhältnis beibehalten
  - Maximalgröße
- Labels im ModelEditor
  - Für nodes und edges anhand Definition in den DSLs
  - Farbe, Schriftart, Größe, Position ...
  - Synchronisation mit dem PropertyPanel
- Gruppierungen im ModelEditor
  - Nodes als Gruppe definieren (per Rechtsklick)
  - Nodes als Kind in die Gruppe einfügen können
  - Beim Löschen einer Gruppe auch die Kinder löschen



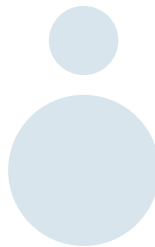
# Erledigte Aufgaben im Detail

- Attribute und Methoden im Concept-Editor links
  - Position anpassen
- Update yFiles auf 2.3
- GitHub-Wiki
  - Fehlerhafte bzw. veraltete Links entfernt / ersetzt
  - Neue Bilder und Beispiele hinzugefügt
  - Wiki Menü angepasst an aktuelles Projekt
  - Veraltete Informationen entfernt / archiviert
  - Aktuellen Stand dokumentiert
  - Struktur des Wikis angepasst
  - Vom Anfang der Entwicklung (Developer Guide) bis zum Ergebnis ist alles dokumentiert

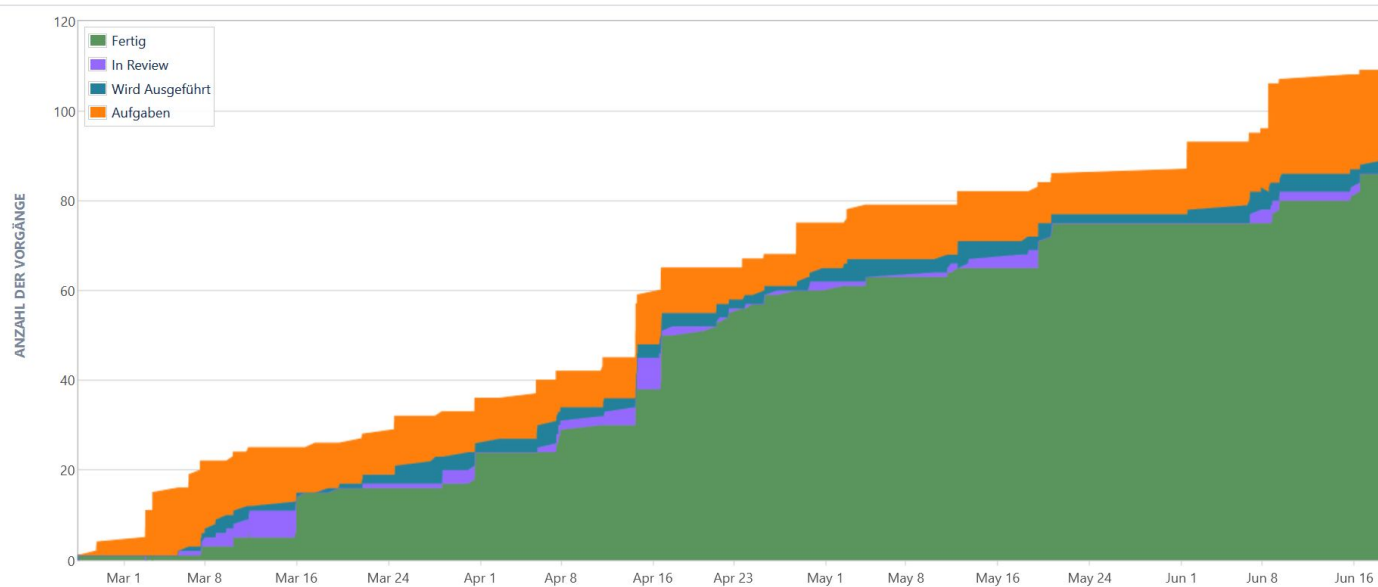


# Sprintauswertung im Detail

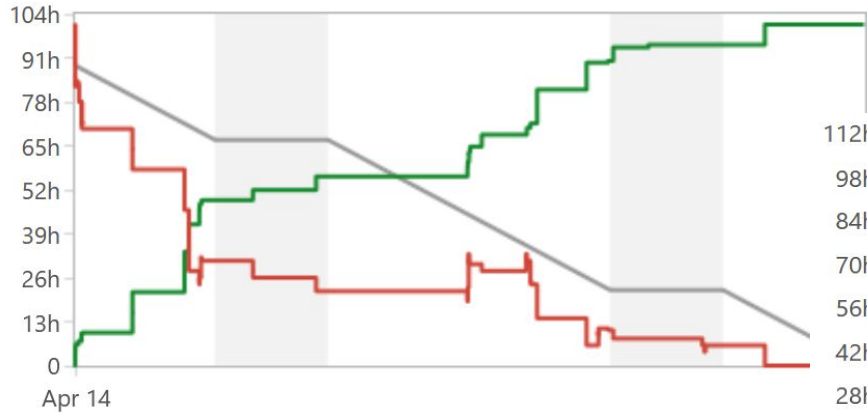
- 33 bearbeitete Pull Requests
- 168 erstellte Commits
  - ca. 21.000 Zeilen Code hinzugefügt
  - ca. 10.000 Zeilen Code entfernt
- 52 bearbeitete Tickets
  - 37 Stories
  - 15 Bugs
- 650 geleistete Stunden (Soll 624h für 6,5 Sprints)
  - Stories 465h
  - Bugs 107h
  - Meetings 78h



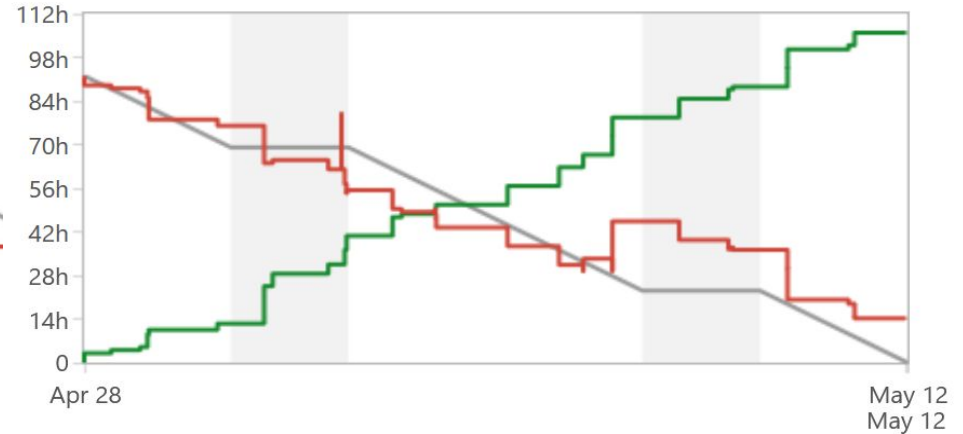
# Sprintauswertung im Detail



# Sprintauswertung im Detail



Sprint 5



**Vielen Dank**  
**für Ihre Aufmerksamkeit**