Dokumentation Projekt M326

Inhaltsverzeichnis

[Einleitung 3](#_Toc408430438)

[Aufgabenstellung 3](#_Toc408430439)

[Aufgabenplanung 3](#_Toc408430440)

[Realisierung 5](#_Toc408430441)

[Analyse 5](#_Toc408430442)

[Design 5](#_Toc408430443)

[Implementation 6](#_Toc408430444)

[Dokumentation 6](#_Toc408430445)

[Programmbeschreibung 7](#_Toc408430446)

[Testkonzept 7](#_Toc408430447)

[Fazit 7](#_Toc408430448)

[Was haben wir erreicht? 7](#_Toc408430449)

[Was konnten wir nicht fertigstellen? 7](#_Toc408430450)

[Welche Probleme hatten wir? 7](#_Toc408430451)

[Was muss nächstes mal besser gelingen? 7](#_Toc408430452)

[Was haben wir gelernt? 7](#_Toc408430453)

[Fazit David 7](#_Toc408430454)

[Schlusswort 8](#_Toc408430455)

[Anhang 8](#_Toc408430456)

# Einleitung

Im Modul 326 der Lehre als Applikationsentwickler gilt es eine Projektarbeit nach der objektorientierten Vorgehensweise zu realisieren. Dies beinhaltet die objektorientierte Analyse, das objektorientierte Designen und die objektorientierte Programmierung. Die Projektarbeit dient dazu, dass die Lernenden lernen, wie ein solches Projekt ablaufen soll und wie man aus der realen Welt ein oder mehrere Objekte analysiert und für die objektorientierte Implementierung in Diagrammen abbildet. Anhand dieser Schritte werden IT-Projekte im optimalen Fall durchgeführt.

## Aufgabenstellung

Die Aufgabenstellung ist wie in der Einleitung beschrieben ein Projekt nach dem entsprechenden Vorgehen abzuwickeln. Das Projekt wird allerdings nicht alleine gemacht sondern in Gruppen mit bis zu fünf Personen.

Der wesentliche Teil der Projektarbeit handelt von der Umsetzung einer Modelleisenbahn der realen Welt in ein Programm mit welchem man einen Zug auf einer Strecke im Kreis fahren lassen kann. Das Programm soll folgende Klassen beinhalten:

* Gerade Schiene
* Gebogene Schiene
* Weiche
* Dreiwegweiche
* Kreuzung
* Einfache Kreuzungsweiche
* Trafo
* Lok
* Güterwagen
* Postwagen

Die Abgaben des Projekts beinhalten folgende Elemente:

* Dokumentation des Vorgehen nach OO
* Arbeitsjournal
* JAR-File der Implementation

## Aufgabenplanung

Für das Projekt fallen einige Tätigkeiten an, welche erledigt werden müssen. Folgend sind die Elemente, welche mindestens vorhanden sein müssen aufgelistet:

* Pflichtenheft
* Use-Case-Diagramm
* Use-Case-Diagramm beschrieben
* Paketdiagramm
* Klassendiagramm
* Beschreibung Klassendiagramm
* Assoziations-Identifikation
* Assoziations-Identifikation beschrieben
* Generalisierungs-Identifikation
* Generalisierungs-Identifikation beschrieben
* Attribute vollständig beschrieben
* Sequenzdiagramm
* Implementation gemäss Klassen- und Sequenzdiagramm
* Testfälle JUnit implementiert
* JAVADoc
* Dokumentation des gesamten Projekts
* Reflexion des gesamten Projekts

Für die Projektarbeit sind folgende Daten in der Schule vorgesehen, geplant mit den Aufgabenbereichen die wir bis zum jeweiligen Datum beendet haben möchten.

* 20.11.2014
  + Projektbeginn
* 27.11.2014
  + Objektdiagramm
  + Beginn der Dokumentation
* 04.12.2014
  + Klassendiagramm
  + Use-Case-Diagramm
* 11.12.2014
  + Paketdiagramm
  + Beschreibung Use-Case-Diagramm
  + Klassendiagramm Teilbeschreibung
  + Assoziations-Identifikation
  + Assoziations-Identifikation beschrieben
* 18.12.2014
  + Generalisierungs-Identifikation
  + Generalisierungs-Identifikation beschrieben
  + Attribute vollständig beschrieben
  + Sequenzdiagramm
  + Grosser Teil der Dokumentation
* 08.01.2015
  + Dokumentation fertig
  + Arbeitsjournal komplett
  + Implementation komplett
  + JAVADoc komplett
  + Testfälle implementiert
  + Reflexion jedes Gruppenmitglieds

An jedem dieser Termine gibt es eine Sprintbesprechung bei welcher diskutiert wird, wie weit wir gekommen sind.

# Realisierung

Bei der Realisierung sind wir mit dem vorgegebenen Schema vorgegangen. Und zwar war dies zuerst die Analyse, dann das Design und schlussendlich die Implementation zu machen. Wir begonnen also mit der Analyse.

## Analyse

Im Analyseprozess ging es vorerst darum, sich mit dem Projektauftrag auseinander zu setzen um sich eine Vorstellung zu machen, wie der Analyseprozess durchgeführt werden soll. Wir nahmen also die Modelleisenbahn und analysierten die einzelnen Bestandteile, welche diese beinhalten soll. Diese sind im Kapitel Aufgabenstellung beschrieben.

Im ersten Schritt erstellten wir nur mal ein Objektdiagramm, welches noch keine Attribute beinhaltete. Diese sollten dann anschliessend folgen.

Der zweite Schritt war den Objekten ihre Eigenschaften, die sie aufweisen müssen, zu zuweisen. Dieser Prozess beinhaltete schon mehr Gedankenarbeit, denn welche Eigenschaften benötigen die Objekte wirklich für die endgültige Implementation? Der Punkt bei diesem Schritt ist, dass man noch nicht soweit vorausschauen soll. Ob schlussendlich mehr oder weniger Eigenschaften benötigt werden kann man künftig entscheiden und verbessern.

Wir haben ein Dokument erstellt, in welchem alle Objekte mit ihren Eigenschaften und wie sie funktionieren beschrieben sind. Ausserdem hat jedes Objekt auch ein Bild dazu bekommen. Der Analyseprozess war somit weitgehend abgeschlossen.

Die Dateien sind im Anhang zu finden.

## Design

In einem ersten Designschritt galt es das Use-Case-Diagramm zu erstellen. Dieses und das Objektdiagramm sollten dazu dienen, die Grundlagen für das Klassendiagramm herzugeben. Für das Klassendiagramm mussten wir uns erstmals überlegen, welche Klassen in das Klassendiagramm gehören. Denn wir brauchen schlussendlich mehr Klassen, als es Objekte als Klassen zu implementieren gilt.

Ein erster Entwurf des Klassendiagramms sollte zuerst mal erstellt werden. Dies ging und wir besprachen meist in zweier Gruppen, wie das Klassendiagramm zu verbessern ist. Der erste Entwurf hatte noch keine Assoziationen und Methoden. Die Assoziationen sollten in einem zweiten Schritt erstellt werden. Was wir uns überlegen mussten, war welche Klassen welche Beziehungen zu welchen Klassen hatten.

Nachdem wir die zweite Version dessen erstellt hatten, war der nächste Schritt beziehungsweise dank der grossen Anzahl an Personen in unserer Gruppe während der Erstellungszeit des Klassendiagramms das Sequenzdiagramm zu erstellen. Das Sequenzdiagramm sollte ein Beispiel darstellen, welche Methoden unsere Objekte benutzen und aufzeigen, wie schlussendlich ein Teil unserer Main-Methode aussehen könnte. Das Sequenzdiagramm sollte später dann bei der Implementierung dienen. Auch diente das Sequenzdiagramm nachdem es fertig gestellt wurde der Implementierung der Methoden in das Klassendiagramm der jeweiligen Klassen. So konnte das Klassendiagramm weitgehend zu Ende gebracht werden.

Nebenbei erstellten wir auch noch das Paketdiagramm, welches auch notwendig für die Vollständigkeit unseres Projekts und unseres objektorientierten Designs ist.

Die Dateien und Diagramme, welche hier beschrieben wurden, sind im Anhang zu finden.

## Implementation

Bei der Implementation ging es vorerst darum, alle Klassen, die wir im Klassendiagramm beschrieben hatten, in Code umzusetzen. Dieser Vorgang war der einfachste der Implementation. Es bestand somit noch keine wirkliche Logik. Diese sollte später folgen.

Die Methoden, die wir im Klassendiagramm beschrieben hatten, waren der zweite Schritt der Implementation. Diese galt es nun in Code umzusetzen. Das heisst beispielsweise es mussten Vorgänge wie die Möglichkeit, Schienen aneinanderzuhängen, Weichen umzustellen, Wagons aneinanderzuhängen und viel mehr möglich sein. Die Logik konnte vorerst nur durch die Implementation noch nicht getestet werden. Als nächstes galt es nämlich zuerst die Main-Methode zu erstellen. Das Programm kann nun durch unsere Main-Methode den Zug über die Schienen fünf Runden fahren lassen und schlussendlich wieder beim Bahnhof anhalten. Er kann die innere Runde nehmen, aber auch die äussere. Dass dies sichtbar ist, wurde dies durch Textausgaben sichergestellt.

Nebst der JAVADOC-Dokumentation, welche wir schlussendlich noch tätigen mussten, galt es auch noch für die Methoden der Klassen, welche wir implementiert hatten und Logik enthalten JUnit-Tests zu schreiben. Diese Tests sind vorhanden und testen, ob die Methoden den richtigen Rückgabewert haben und somit funktionieren oder eben nicht funktionieren. Dies entwickelt sich für uns zu einem starken Vorteil, denn wenn etwas nicht so läuft wie es sollte bemerken wir dies schnell und können so die Verbesserungen an unseren Methoden schnell durchführen.

## Dokumentation

Die Dokumentation ist einer der wichtigsten Bestandteile eines Programmes. Sie dient der Wartbarkeit und der Nachvollziehbarkeit einer anderen Person des gesamten Projekts und des Programms. Diese Dokumentation wurde von uns während des gesamten Projekts aktualisiert. Den grössten Teil allerdings konnten wir erst gegen den Schluss des Projekts erledigen, was selbstverständlich ist, denn das Programm ist ja auch erst eher gegen Ende des Projekts fertig.

# Programmbeschreibung

*Nach Packages gegliedert???*

# Testkonzept

# Fazit

## Was haben wir erreicht?

## Was konnten wir nicht fertigstellen?

## Welche Probleme hatten wir?

## Was muss nächstes mal besser gelingen?

## Was haben wir gelernt?

## Fazit David

Vorerst, als wir die Information zugeteilt bekamen, dass das ganze Modul 326 eine Projektarbeit ist, war ich alles andere als begeistert. Jedoch hat sich meine Begeisterung ein bisschen gesteigert bis zum Ende Projekt. Im Nachhinein kann ich sagen ist es doch eine gute Idee den Lernstoff so zu übermitteln. Man konnte einiges lernen, gerade beim Ablauf und bei der Abwicklung eines solchen Projekts. In dieser Art hatte ich dies noch nie. Ich lernte auch mehr über das objektorientierte Analysieren und die weiteren Schritte die man durchlaufen muss.

Probleme gab es bei mir bei der Klassendiagrammerstellung und auch mit der Implementation anhand eines objektorientierten Designs habe ich noch Mühe. Bei beiden Tätigkeiten fehlen mir Praxis und manchmal auch Verständnis, was aber durchaus durch Nachfragen bei Teamkollegen verbessert werden konnte. Ich werde mich aber künftig noch mehr mit dem Thema beschäftigen müssen.

Negative Aspekte, die das Projekt ein bisschen geprägt hatten, waren beispielsweise die Zeit. Wir mussten definitiv zu viel Freizeit dafür investieren. Man sollte in der Schule genügend Zeit für solche Projekte bekommen und tun wir dies nicht, so sollten die Projekte kleiner werden.

Zudem war der Lerneffekt in allen Bereichen, die dieses Projekt umschloss nicht gerade allzu gross, weil wir fünf Personen sind die daran gearbeitet hatten.

Wir oder zumindest ich wurde ein bisschen ins kalte Wasser geschmissen mit dem Projekt, kann aber nun sagen, dass ich aus diesem Projekt einiges mitnehmen kann für die Zukunft, falls es in dieser Form in meinem Arbeitsleben wieder auftreten wird.

# Schlusswort

# Anhang

|  |  |
| --- | --- |
| **Beschreibung** | **Dokument** |
| Kapitel Analyse,Design,Implementation!!!!!!!!!! |  |
|  |  |