

# Requirements and Design Documentation (RDD)

Version 2.5

## SE2P – Praktikum – WS 13/14

Duske, Natalia, 2063265, natalia.duske@haw-hamburg.de  
Schick, Jannik, 2086534, jannik.schick@haw-hamburg.de  
Küpelikilinc, Rutkay, 2081831, rutkay.kuepelikilinc@haw-hamburg.de  
Kloth, Philipp, 2081738, philipp.kloth@haw-hamburg.de

### Änderungshistorie:

Version	Author	Datum	Anmerkungen
0.1	Natalia Duske	03.10.2013	
1.1	Natalia Duske	20.10.2013	USE-CASE-Diagramm, Klassendiagramm, Komponentendiagramm hinzugefügt
2.1	Natalia Duske	17.11.2013	Anlagemodellierung mit Petri-Netzen Klassendiagramm angepasst
2.2	Philipp Kloth	18.11.2013	Regressionstest angepasst
2.3	Natalia Duske	26.11.2013	Petrinetze aktualisiert, Klassendiagrammangepasst
2.4	Natalia Duske	10.12.2013	Systemtest
2.5	Natalia Duske	12.12.2013	Lessons Learned, Glossar, Abkürzungen, Unit/Komponententest, Testprotokolle hinzugefügt

# Inhalt

Motivation.....	3
Randbedingungen.....	3
Entwicklungsumgebung.....	3
Werkzeuge.....	3
Sprachen.....	3
Requirements und Use Cases.....	4
Anforderungen.....	4
Use-Case-Diagramm.....	7
Design.....	8
System Architektur.....	8
Datenmodell.....	9
Verhaltensmodell.....	10
Implementierung.....	13
Patterns.....	13
Testen.....	14
Unit Test/Komponenten Test.....	14
Integration Test/System Test.....	16
Regressionstest.....	20
Abnahmetest.....	21
Testplan.....	21
Projektplan.....	22
Verantwortlichkeiten.....	22
PSP und Zeitplan.....	22
Lessons Learned.....	22
Glossar.....	24
Abkürzungen.....	24
Anhänge.....	24

## 1. Motivation

Zu entwickeln ist ein System für eine Anlage, das die richtige Sortierung der Werkstücke durchführt.

Richtige Werkstücke sind solche, die über eine Bohrung verfügen. Dabei wird zwischen einem Werkstück mit Bohrung ohne Metalleinsatz und mit Metalleinsatz unterschieden. Alle anderen Werkstücke sollen aussortiert werden.

Die Fabrikanlage verfügt über zwei Fließbänder mit jeweils einer Ampel, Höhenmesser, Weiche und Rutsche.

Die Stakeholder sind:

- 1.Entwicklerteam
- 2.Bedienpersonal
- 3.Inhaber/Auftraggeber
- 4.Expertenteam

## 2. Randbedingungen

### 2.1. Entwicklungsumgebung

Betriebssysteme:

- 1.MS Windows 7
- 2.QNX 6.5

Entwicklungsumgebung:

- 1.QNX Momentics 6.5.0

Simulator:

- 1.VBOX QNX Simulation

### 2.2.Werkzeuge

- 1.Microsoft Project Professional 2013
- 2.Microsoft Office 2010
- 3.Microsoft Visio 2013
- 4.Visual Paradigm for UML 10.1
- 5.SourceTree
- 6.HPetriSim

### 2.3. Sprachen

C++

### 3. Requirements und Use Cases

#### 3.1. Anforderungen

##### 3.1.1 Zu flache Werkstücke

Vorbedingung: Das Werkstück muss auf dem Laufband 1 sein und das Laufband soll einwandfrei funktionieren und die Ampel grün anzeigen.

Ablauf: Das zu flache Werkstück wird an den Anfang des Laufbandes 1 gelegt und durch Höhenmesser als zu flach erkannt. Die Weiche bleibt geschlossen und das Werkstück wird auf die Rutsche geschoben.

Nachbedingung: keine

Funktionale Anforderungen: Erkennen und Aussortieren von zu flachen Werkstücken

Nicht-funktionale Anforderungen:

##### 3.1.2 Werkstück mit Metalleinsatz und Bohrung nach unten

Vorbedingungen: Anfang des Laufbandes 1 ist frei, Laufband läuft und die Ampel leuchtet grün.

Ablauf: Das Werkstück mit Bohrung und Metalleinsatz wird an den Anfang des Laufbandes 1 gelegt. Das Werkstück passiert die Höhenmessung und es wird kein Loch erkannt, obwohl die Größe in Ordnung ist. Das Laufband 1 wird angehalten und die gelbe Signalleuchte blinkt. Das Werkstück muss vom Personal umgedreht werden. Am Laufband 2 wird die Bohrung nach oben erkannt und zurück an den Anfang des Laufbands 2 gefahren und das Band 2 wird zum Umdrehen angehalten. Die gelbe Signalleuchte von Band 2 blinkt und das Werkstück muss erneut von Hand umgedreht werden. Wird beim zweiten Durchlauf auf Band 2 die Bohrung nach oben erkannt, bleibt die Weiche geschlossen und das Werkstück wird auf die Rutsche aussortiert. Ist beim zweiten Durchlauf nach dem Umdrehen des Werkstücks die Bohrung unten, wird die Weiche geöffnet und das Werkstück durchgelassen. Danach wird die Weiche wieder geschlossen. Das Werkstück erreicht das Ende des Laufbands 2. Das Laufband wird angehalten und das Bedienpersonal nimmt das Werkstück vom Laufband ab.

Randbedingungen: beim Umdrehen darf die dafür vorgesehen maximale Zeit nicht überschritten werden. Beim Erreichen des Laufbands 2, darf auf dem Laufband 2 kein anderes Werkstück liegen.

Nachbedingungen: nach dem erfolgreichen Durchlauf wird das Band 2 gestoppt sobald keine weiteren Übermittlungen vom Band 1 eintreffen bzw. das Band läuft weiter wenn weiteres Werkstück auf das 2. Band transportiert werden muss.

Funktionale Anforderungen: Bohrung und Metalleinsatz erkennen, die Informationen zu dem Werkstück über die serielle Schnittstelle vom Band 1 nach Band 2 weiterreichen, Informationen zum Werkstück erfassen.

Nicht-funktionale Anforderungen: Das Werkstück muss innerhalb der bestimmten Zeit umgedreht werden.

##### 3.1.3 Werkstück mit Metalleinsatz und Bohrung nach oben

Vorbedingungen: Anfang des Laufbandes 1 ist frei, Laufband 1 läuft und die Ampel leuchtet grün.

Ablauf: Das Werkstück mit Bohrung und Metalleinsatz wird am Anfang des Laufbandes 1 gelegt. Das Werkstück passiert die Höhenmessung und es wird ein Loch mit Metalleinsatz erkannt. Das Laufband 1 wird angehalten und die gelbe Signalleuchte blinkt. Das Werkstück muss vom Personal umgedreht werden. Wird die Bohrung am Laufband 2 nach unten erkannt wird die Weiche geöffnet und das Werkstück durchgelassen. Danach wird die Weiche wieder geschlossen. Das Werkstück erreicht das Ende des Laufbands 2. Das Laufband wird angehalten und das Bedienpersonal nimmt das Werkstück vom Laufband ab.

Wird jedoch auf Band 2 die Bohrung nach oben erkannt, bleibt die Weiche geschlossen und das

Werkstück wird an den Bandanfang zurück gefahren und umgedreht. Wird beim zweiten Durchlauf auf dem Band 2 die Bohrung nach oben erkannt, bleibt die Weiche geschlossen und das Werkstück wird auf die Rutsche aussortiert. Wird jedoch keine Bohrung nach der Umdrehung erkannt, die Weiche wird geöffnet und das Werkstück wird an das Ende des Bandes transportiert.

Randbedingungen: beim Umdrehen darf die dafür vorgesehene maximale Zeit nicht überschritten werden. Beim Erreichen des Laufbandes 2, darf auf dem Laufband 2 kein anderes Werkstück liegen.

Nachbedingungen: nach dem erfolgreichen Durchlauf wird das Band 2 gestoppt sobald keine weiteren Übermittlungen vom Band 1 eintreffen bzw. das Band läuft weiter wenn weiteres Werkstück auf das 2. Band transportiert werden muss.

Funktionale Anforderungen: Bohrung und Metalleinsatz erkennen, die Informationen zu dem Werkstück über die serielle Schnittstelle vom Band 1 nach Band 2 weiterreichen, Informationen zum Werkstück erfassen.

Nicht-funktionale Anforderungen: Das Werkstück muss innerhalb der bestimmten Zeit umgedreht werden.

### **3.1.4 Werkstück ohne Metalleinsatz mit Bohrung nach unten**

Vorbedingungen: Anfang des Laufbandes 1 ist frei, Laufband 1 läuft und die Ampel leuchtet grün.

Ablauf: Das Werkstück mit Bohrung wird am Anfang des Laufbandes 1 gelegt. Das Werkstück passiert die Höhenmessung und es wird kein Loch erkannt, obwohl die Größe in Ordnung ist. Das Laufband 1 wird angehalten und die gelbe Signalleuchte blinkt. Das Werkstück muss vom Personal umgedreht werden. Am Laufband 2 wird die Bohrung nach oben erkannt, die Weiche wird geöffnet und das Werkstück durchgelassen. Danach wird die Weiche wieder geschlossen. Das Werkstück erreicht das Ende des Laufbandes 2. Das Laufband wird angehalten und das Bedienpersonal nimmt das Werkstück vom Laufband ab.

Randbedingungen: beim Umdrehen darf die dafür vorgesehene maximale Zeit nicht überschritten werden. Beim Erreichen des Laufbandes 2, darf auf dem Laufband 2 kein anderes Werkstück liegen.

Nachbedingungen: nach dem Wegnehmen vom Werkstück wird das Laufband angehalten wenn das Laufband 1 keine weiteren Werkstücke meldet.

Funktionale Anforderungen: Bohrung erkennen, die Informationen zu dem Werkstück über die serielle Schnittstelle vom Band 1 nach Band 2 weiterreichen, Informationen zum Werkstück erfassen.

Nicht-funktionale Anforderungen: Das Werkstück muss innerhalb der bestimmten Zeit umgedreht werden.

### **3.1.5 Werkstück ohne Metalleinsatz mit Bohrung nach oben**

Vorbedingungen: Anfang des Laufbandes 1 ist frei, Laufbänder laufen langsam und einwandfrei und die Ampel leuchtet grün.

Ablauf: Das Werkstück mit Bohrung wird am Anfang des Laufbandes 1 gelegt. Das Werkstück passiert die Höhenmessung und es wird ein Loch erkannt. Das Werkstück darf auf Laufband 2 weitergeleitet werden. Am Laufband 2 wird die Bohrung nach oben erkannt, die Weiche wird geöffnet und das Werkstück durchgelassen. Danach wird die Weiche wieder geschlossen. Das Werkstück erreicht das Ende des Laufbandes 2. Das Laufband wird angehalten und das Bedienpersonal nimmt das Werkstück vom Laufband ab.

Randbedingungen: Beim Erreichen des Laufbandes 2, darf auf dem Laufband 2 kein anderes Werkstück liegen.

Nachbedingungen: nach dem Wegnehmen vom Werkstück wird das Laufband angehalten wenn das Laufband 1 keine weiteren Werkstücke meldet.

Funktionale Anforderungen: Bohrung und die Position der Bohrung erkennen, die Informationen zu dem Werkstück über die serielle Schnittstelle vom Band 1 nach Band 2 weiterreichen, Informationen zum Werkstück erfassen.

Nicht-funktionale Anforderungen:

### **3.1.6 Rutsche ist voll**

Vorbedingungen: Laufband läuft einwandfrei und die Ampel leuchtet grün.

Ablauf: Der Sensor erkennt, dass die Rutsche voll ist, d.h. ein Werkstück liegt bereits ganz oben. Das Laufband wird angehalten und die Ampel blinkt schnell rot (1Hz). Das Bedienpersonal betätigt die Quittierungstaste. Die Ampel leuchtet rot. Das Bedienpersonal leert die Rutsche. Danach wird vom Personal erneut die Quittierungstaste betätigt und die Ampel wechselt von rot auf grün. Das Laufband läuft wieder.

Randbedingungen: keine

Nachbedingungen: Die Rutsche wurde komplett geleert.

Funktionale Anforderungen: Platzmangen auf der Rutsche erkennen und dem System melden.

Nicht-funktionale Anforderungen:

### **3.1.7 Verschwinden von Werkstücken auf dem Band 1**

Vorbedingungen: Laufband läuft einwandfrei und die Ampel leuchtet grün. Ein bereits vom System erfasstes Werkstück verschwindet vom Laufband.

Ablauf: Der Sensor erkennt, dass ein zuvor vom System registriertes Werkstück nicht mehr auf dem Laufband vorhanden ist. Das Laufband wird angehalten und die Ampel blinkt schnell rot (1Hz). Das Bedienpersonal betätigt die Quittierungstaste. Die Ampel leuchtet rot. Es wird nach dem fehlenden Werkstück seitens des Bedienpersonals gesucht. Danach wird vom Personal erneut die Quittierungstaste betätigt und die Ampel wechselt von rot auf grün. Das Laufband läuft wieder.

Randbedingungen: Das Werkstück soll auffindbar sein

Nachbedingungen:

Funktionale Anforderungen: fehlendes Werkstück erkennen und Fehler melden.

Nicht-funktionale Anforderungen:

### **3.1.8 Hinzufügen von Werkstücken auf dem Band 1**

Vorbedingungen: Laufband läuft einwandfrei und die Ampel leuchtet grün. Eins dem System unbekanntes, d.h. nicht registriertes Werkstück wird vom System erkannt.

Ablauf: Der Sensor erkennt, dass ein zuvor vom System nicht registriertes Werkstück sich auf dem Laufband befindet. Das Laufband wird angehalten und die Ampel blinkt schnell rot (1Hz). Das Bedienpersonal betätigt die Quittierungstaste. Die Ampel leuchtet rot. Das Bedienpersonal entfernt alle Werkstücke vom Laufband und betätigt erneut die Quittierungstaste. Die Ampel wechselt von rot auf grün. Das Laufband läuft wieder.

Randbedingungen: Die Werkstücke dürfen nicht aufeinander gestapelt werden.

Nachbedingungen: Alle sich auf dem Laufband befindenden Werkstücke wurden entfernt.

Funktionale Anforderungen: neues nicht registriertes Werkstück erkennen und Fehler melden

Nicht-funktionale Anforderungen: schnelle Erkennung des Werkstücks, da dieses nicht dazu gehört

### **3.1.9 Verhalten auf dem Band 2**

Vorbedingungen: Die Software sorgt dafür, dass auf das Band 2 nur die dafür vorgesehenen Werkstücke kommen. Die Flachen Werkstücke werden schon auf dem Band 1 aussortiert.

Ablauf:

Randbedingungen: Das Laufband läuft nur, wenn sich auf dem Band ein Werkstück befindet

und/oder die Daten für das nächste Werkstück vom Band 1 übermittelt werden.

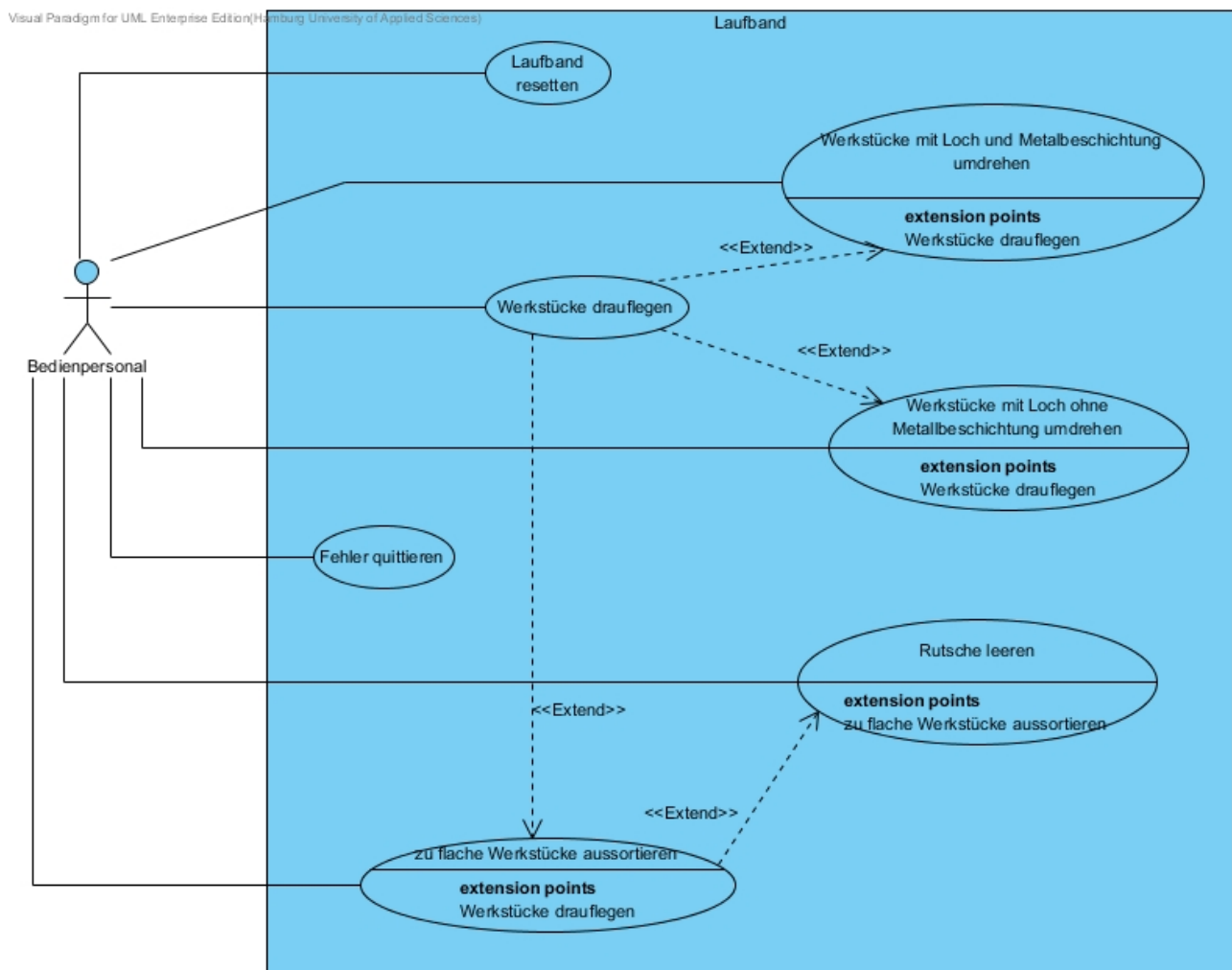
Nachbedingungen: Alle sich auf dem Laufband befindenden Werkstücke wurden entfernt.

Funktionale Anforderungen: Auf dem Band dürfen sich nur die Werkstücke befinden die eine richtige Höhe haben und über eine Bohrung (mit und ohne Metalleinsatz) befinden.

Nicht-funktionale Anforderungen: Es dürfen bis auf das Umdrehen der Werkstücke mit Metalleinsatz keine weiteren manuellen Eingriffe erfolgen.

### 3.2.Use-Case-Diagramm

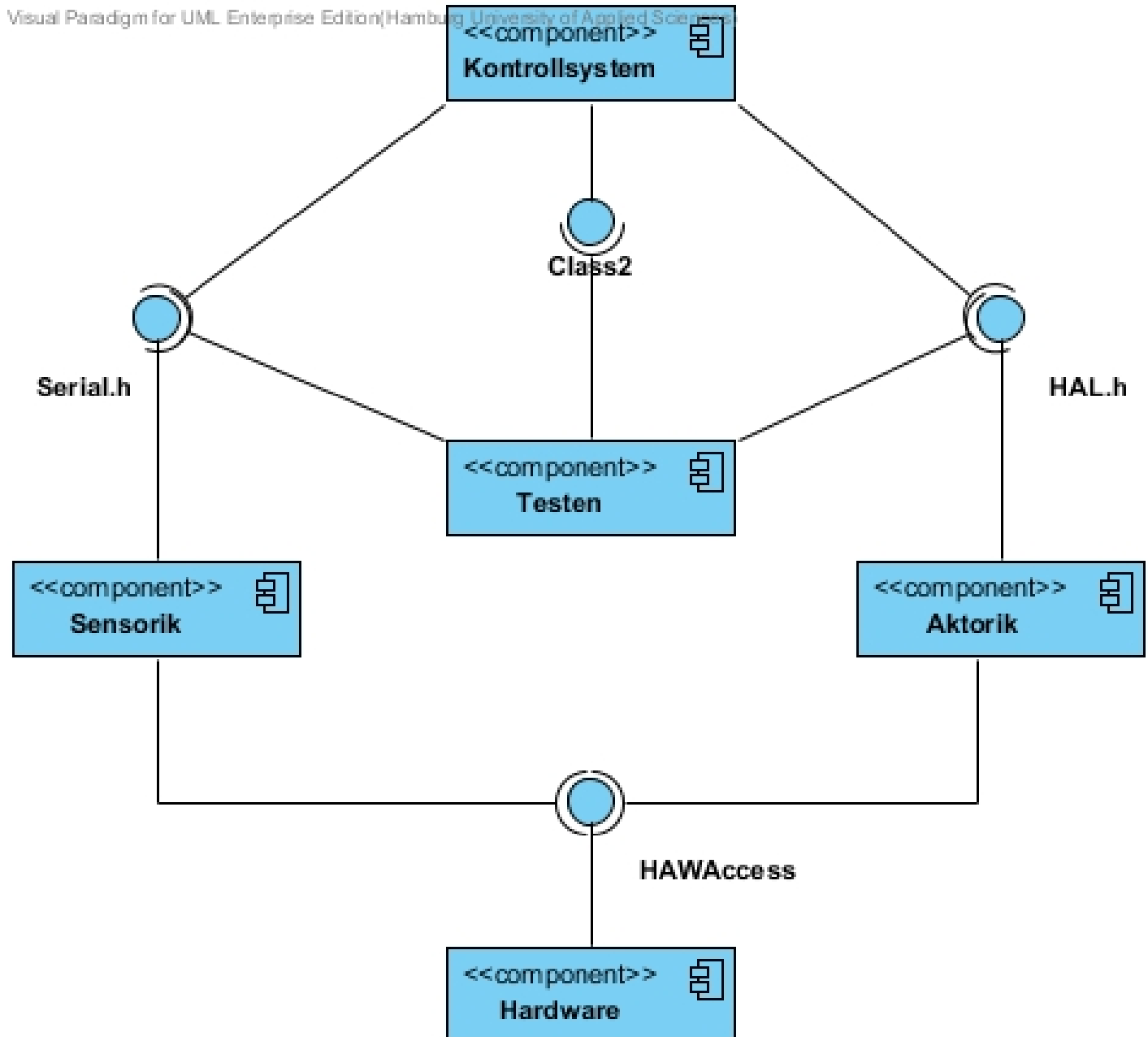
Use-Case-Diagramm stellt Stakeholder – Bedienerpersonal dar und beschreibt die groben Abläufe der Anlage bzw. zeigt wie das Bedienerpersonal auf die Anlage einwirkt und welche wesentlichen Möglichkeiten ihm zur Verfügung stehen um die Sortieranlage zu bedienen.



## 4. Design

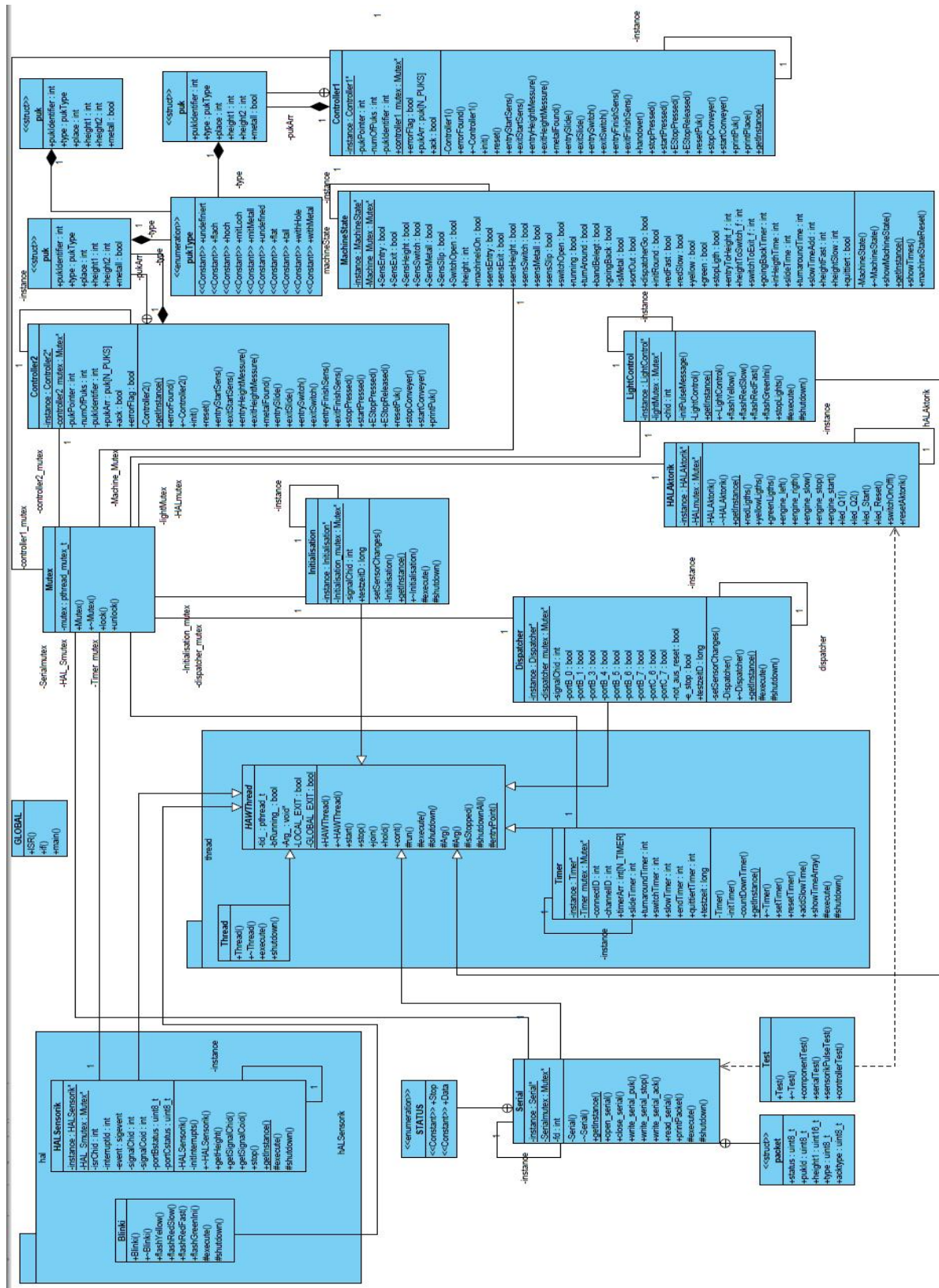
### 4.1. System Architektur

Auf der untersten Ebene des Systems befindet sich die Hardware. Sie bietet lediglich eine Schnittstelle um mit bestimmten Teilen der Hardware arbeiten zu können. Diese Schnittstelle wird von den HAL-Klassen (HAL Sensorik und HAL Aktorik) benutzt um HW-Funktionalitäten in Anspruch zu nehmen. Die HAL-Klassen stellen somit eine Zwischenschicht dar um die Kommunikation des Kontrollsystems mit der Hardware möglich zu machen.





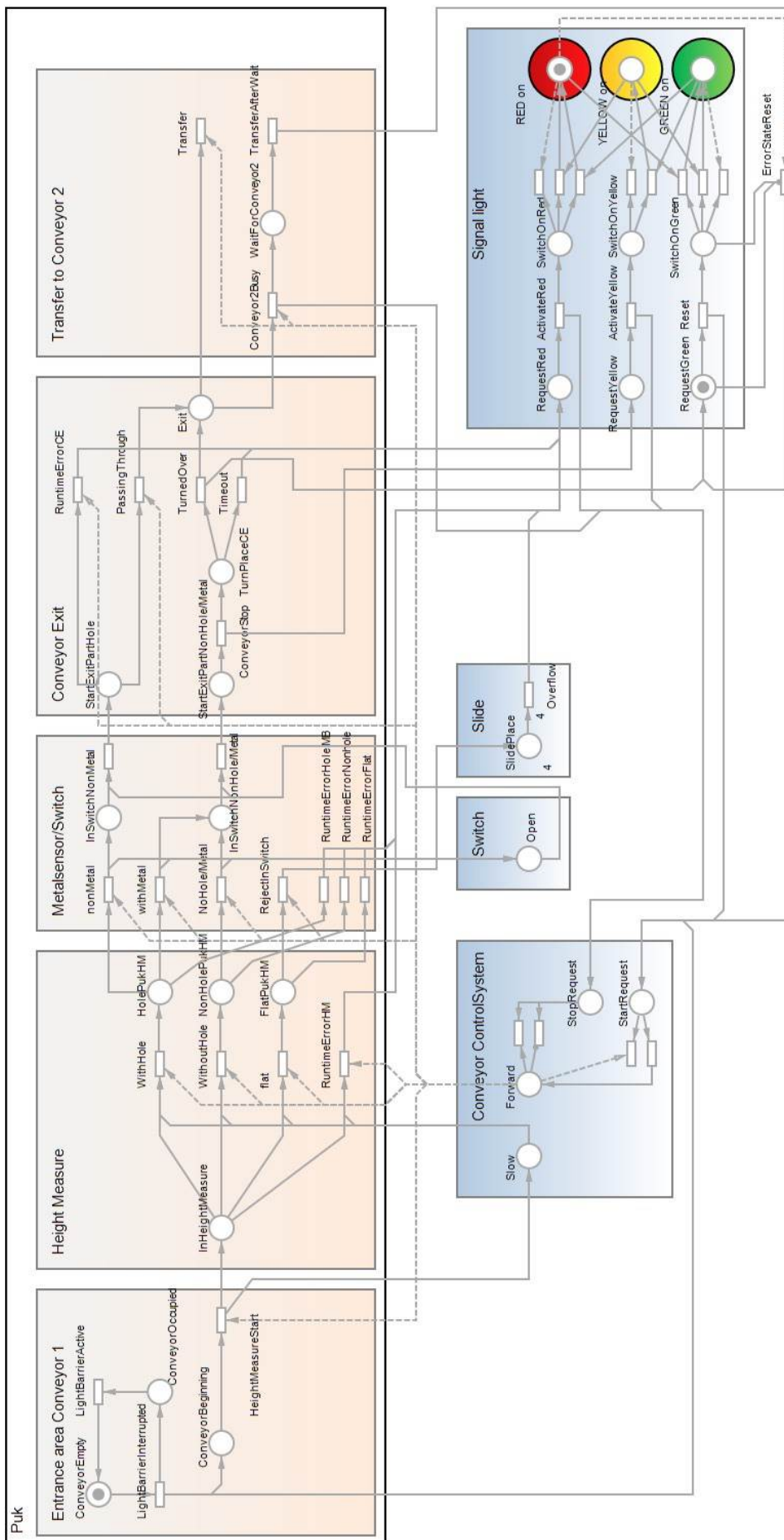
## 4.2. Datenmodell



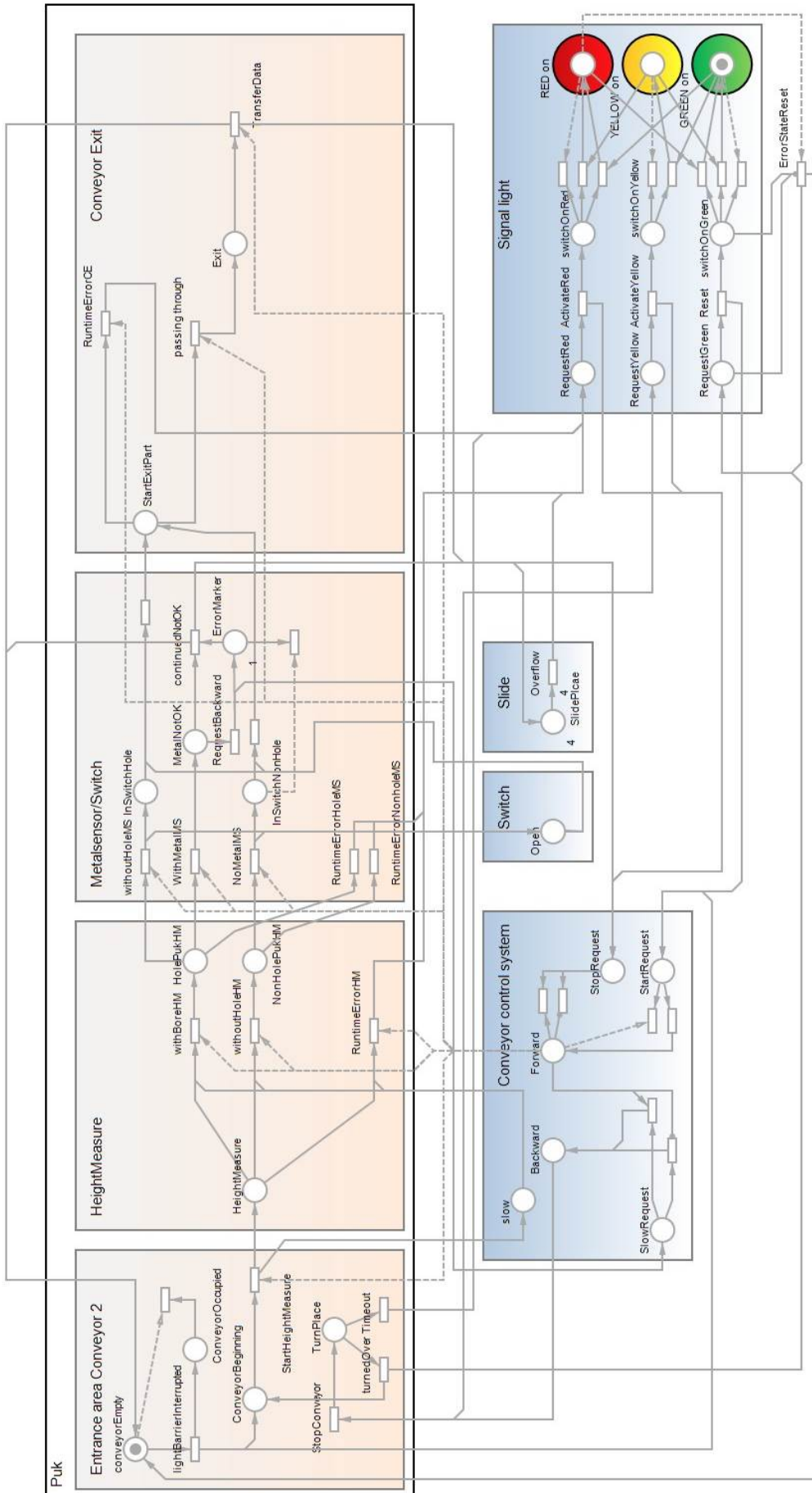
### 4.3. Verhaltensmodell

Das Verhaltensmodell wird durch Petrinetze dargestellt. Jedes Laufband hat ein eigenes Petrinetz und zwar aus der Sicht eines einzelnen Pucks. Die Ampel und die Steuerung der Laufbänder beziehen sich jedoch auf die Arbeit der gesamten Anlage als Ganzes und werden von mehreren Pucks gleichzeitig angesprochen. Die Darstellung der Ampel und der Bandsteuerung ist im Petrinetz für einen Puck enthalten um die Reaktion auf bestimmte Ereignisse zu verdeutlichen.

### 4.3.1. Band 1



### 4.3.2. Band 2



## 5. Implementierung

Die Implementierung des Petrinetzes erfolgt über 2 Klassen. 1. Controller1 welcher das Petrinetz von Band 1 abarbeitet und des Weiteren Controller2 der das gesamte Petrinetz von Band 2 abarbeitet. Ein Dispatcher horcht auf eingehende Signale und stößt damit den Controller an. Der Controller bedient die Pucks über einen FIFO Puffer und prüft die Stellen ab um daraufhin die Transaktion zu schalten.

Der Timer wurde realisiert über eine eigene Klasse die einen Timer startet der alle 1ms einen Impuls bekommt und die Zeiten der Pucks jeweils um 1 runterzählt. Bei 0 ist dementsprechend ein Timeout vorhanden.

Beim Start der Anlage muss ein Initialisierungslauf durchgeführt werden um die Zeiten der Anlage zu messen und entsprechend einzuspeichern. Dafür müssen die Pucks an der rechten Kante losfahren und 2 mal durch die Anlage fahren. Dabei wird einmal schnell durchgefahren und einmal langsam in der Höhenmessung um die genauen Zeiten zu haben. Danach ist die Anlage startbereit

```
HALSensorik::getInstance()->start(NULL);
Timer::getInstance()->start(NULL);
Initialisation::getInstance()->start(NULL);
LightControl::getInstance()->start(NULL);
MachineState::getInstance()->green = true;

while (MachineState::getInstance()->dispatcherGo) {
    usleep(50000);
}
MachineState::getInstance()->stopLigth = true;
```

Bei der Fehlerbehandlung wird die Quittierung der Fehler mit Hilfe von „Reset“ Taste realisiert. Wenn der Fehler auftritt, blinkt das rote Licht schnell und kennzeichnet den Zustand „anstehend unquittiert“. Durch das Drücken der „Reset“ Taste leuchtet rotes Licht dauerhaft rot und der Fehler muss manuell behoben werden. Nachdem das geschehen ist, muss die „Reset“ Taste erneut gedrückt werden, damit die Anlage in den betriebsbereiten Zustand wechselt.

### 5.1.Patterns

Singleton Pattern

Observer Pattern  
- Dispatcher

## 6. Testen

### 6.1. Unit Test/Komponenten Test

In diesem Abschnitt wird die Anlage auf die richtige Reaktion der eintretenden Ereignisse getestet. Die Pucks werden einzeln auf das Band 1 gelegt und müssen die gesamte Strecke auf dem Band durchlaufen. Mit Hilfe von Debug-Ausgaben kann zusätzlich analysiert werden, ob es sich um einen erfolgreichen Durchlauf handelt oder ob ein fehlerhaftes Verhalten bezüglich der Puck Positionen und deren Auswertung auftritt.

Auf dem Band wird ebenfalls getestet ob sich die Anlage bei jeder Art von Pucks gemäß den Anforderungen verhält.

Folgende Werkstücke wurden auf dem Band 1 und 2 getestet:

Flat  
Hoch mit Loch  
Hoch Loch unten  
Metall oben  
Metall unten  
Metall unten und auf dem Band 2 nicht umgedreht

#### Auf dem Band 1 kamen beim Test folgende Ausgaben:

Werkstück Nr. : 0  
ConveyerEmpty  
ConveyorBeginnind  
InHeightMeasure  
FlatPukHM  
Werkstück Nr. : 1  
ConveyerEmpty  
ConveyorBeginnind  
InHeightMeasure  
HolePukHM  
InSwitchNonMetal  
InSwitchNonMetal  
StartExitPartHole  
    send puk  
    send puk  
WaitForConveyor2  
Reset pusht! Machine is ready  
Werkstück Nr. : 2  
ConveyerEmpty  
ConveyorBeginnind  
InHeightMeasure  
NonHolePukHM  
inSwitchNonHoleMetal  
StartExitNonPartHoleMetal  
Turnplace 10  
    send puk  
    send puk  
WaitForConveyor2  
Reset pusht! Machine is ready  
Werkstück Nr. : 3  
ConveyerEmpty  
ConveyorBeginnind  
InHeightMeasure  
HolePukHM  
inSwitchNonHoleMetal  
StartExitNonPartHoleMetal  
Turnplace 10  
    send puk  
    send puk

```

WaitForConveyor2
Reset pusht! Machine is ready
Werkstück Nr.: 4
ConveyerEmpty
ConveyorBeginni nd
InHeightMeasure
NonHolePukHM
inSwitchNonHoleMetal
StartExitNonPartHoleMetal
Turnplace 10
  send puk
  send puk
WaitForConveyor2
Reset pusht! Machine is ready
Werkstück Nr.: 5
ConveyerEmpty
ConveyorBeginni nd
InHeightMeasure
NonHolePukHM
inSwitchNonHoleMetal
StartExitNonPartHoleMetal
Turnplace 10
  send puk
  send puk
WaitForConveyor2
Reset pusht! Machine is ready

```

### Ausgaben vom Band 2:

```

send ack
Werkstuecknummer: 1
ConveyerEmpty
ConveyorBeginni nd
HolePukHM
InSwitchHole
StartExitPart
Puk:
PukID: 1
Hohe1: 3543
Hohe2: 3615
Typ: with Hole
Exit
ConveyerEmpty

```

```

  send ack
Werkstuecknummer: 2
ConveyerEmpty
ConveyorBeginni nd
HolePukHM
InSwitchHole
StartExitPart
Puk:
PukID: 0
Hohe1: 2534
Hohe2: 3609
Typ: with Hole
Exit
ConveyerEmpty

```

```

  send ack
Werkstuecknummer: 3
ConveyerEmpty
ConveyorBeginni nd
NonHolePukHM
InSwitchHole
StartExitPart
Puk:
PukID: 0
Hohe1: 3622
Hohe2: 2526
Typ: with Metal

```



Exit  
ConveyerEmpty

send ack  
Werkstuecknummer: 4  
ConveyerEmpty  
ConveyorBeginnind  
HolePuckHM  
MetallNotOk  
ConveyerEmpty  
ConveyorBeginnind  
NonHolePuckHM  
InSwitchHole  
StartExitPart  
Puk:  
PukID: 0  
Hohe1: 2529  
Hohe2: 2548  
Typ: with Metal  
Exit  
ConveyerEmpty

send ack  
Werkstuecknummer: 5  
ConveyerEmpty  
ConveyorBeginnind  
HolePuckHM  
MetallNotOk  
ConveyerEmpty  
ConveyorBeginnind  
HolePuckHM  
ConveyerEmpty

## 6.2. Integration Test/System Test

### 6.2.1. Puck Variationen

PF	flacher Puck
PMO	Puck mit Metall, Bohrung oben
PMU	Puck mit Metall, Bohrung unten
PBO	Puck ohne Metall, Bohrung oben
PBU	Puck ohne Metall, Bohrung unten

### 6.2.2. Testszzenarien für beider Laufbänder

Im folgenden Abschnitt werden einige Testszzenarien vorgestellt, die die Arbeitsweise der Sortieranlage mit dem zu erwarteten Ergebnis darstellen.

Szenario	Erwartetes Ergebnis
PF	Der Flache Puck wird auf das Band 1 gelegt, in der Höhenmessung als zu flach erkannt und in die Rutsche aussortiert



PMO	<p>Der Puck wird am Anfang auf das Band 1 gelegt, in der Höhenmessung wird die Höhe akzeptiert und eine Bohrung erkannt. Metallsensor erkennt Metall, Weiche wird geöffnet und im Auslauf wird das band gestoppt und der Puck muss umgedreht werden. Anschließend werden die Daten über die serielle Schnittstelle an das Band 2 weiter gereicht. Wenn das Band 2 frei ist, wird der Puck weiter transferiert ohne weitere Aktionen auf Band 2 vorzunehmen, erreicht der Puck das Ende vom Band 2. Band 2 ist wieder frei.</p>
PMU	<p>Der Puck wird am Anfang auf das Band 1 gelegt, in der Höhenmessung wird die Höhe akzeptiert, Bohrung wird nicht erkannt. Metallsensor erkennt kein Metall, Weiche wird geöffnet und im Auslauf wird das band gestoppt und der Puck muss umgedreht werden. Anschließend werden die Daten über die serielle Schnittstelle an das Band 2 weiter gereicht. Wenn das Band 2 frei ist, wird der Puck weiter transferiert. Metallsensor erkennt Metall und der Puck wird zum Anfang des Bands 2 gefahren, Band stoppt und der Puck wird umgedreht. Bohrung wird nicht erkannt, Metallsensor erkennt auch kein Metall. Weiche wird geöffnet und der Puck erreich das Ende vom band 2. Band 2 ist wieder frei.</p>
PBO	<p>Der Puck wird am Anfang auf das Band 1 gelegt, in der Höhenmessung wird die Höhe akzeptiert und eine Bohrung erkannt. Metallsensor erkennt kein Metall, Weiche wird geöffnet und der Puck weiter befördert. Anschließend im Auslauf werden die Daten über die serielle Schnittstelle an das Band 2 weiter gereicht. Wenn das Band 2 frei ist, wird der Puck aufs Band 2 transferiert und ohne weitere Aktionen auf dem Band 2 vorzunehmen, erreicht der Puck das Ende vom Band 2. Band 2 ist wieder frei.</p>

PBU	<p>Der Puck wird am Anfang auf das Band 1 gelegt, in der Höhenmessung wird die Höhe akzeptiert, eine Bohrung wird nicht erkannt. Metallsensor erkennt kein Metall, Weiche wird geöffnet und der Puck weiter befördert. Anschließend im Auslauf werden die Daten über die serielle Schnittstelle an das Band 2 weiter gereicht. Wenn das Band 2 frei ist, wird der Puck aufs Band 2 transferiert und ohne weitere Aktionen auf dem Band 2 vorzunehmen, erreicht der Puck das Ende vom Band 2. Band 2 ist wieder frei.</p>
PF - PF - PF - PF	<p>Die flachen Pucks werden nacheinander auf Band 1 gelegt. Höhenmessung werden alle vier als zu flach erkannt und vor der Weiche in die Rutsche aussortiert. Weiche bleibt die ganze Zeit zu. Der vierte Puck unterbricht für längere Zeit die Lichtschranke auf der Rutsche und der Fehler "Rutsche voll" tritt auf. Der Fehler wird quittiert. Rutsche wird geräumt und danach der normale Betrieb wieder aufgenommen</p>
PBO - PF - PBO	<p>Am Anfang von Band 1 wird der Puck aufs Band gelegt, gefolgt vom flachen Puck und dann kommt wieder normaler Puck mit Bohrung oben. Der erste Puck wird von der Höhenmessung als normaler Puck mit Loch erkannt, Metallsensor erkennt kein Metall, Weiche wird geöffnet und er fährt durch. Muss nicht umgedreht werden und wird sofort auf das Band 2 transferiert, wenn dieses frei ist. Und erreicht so in diesem Zustand das Ende vom Band 2. Der flache Puck wird in der Höhenmessung als zu flach erkannt und in die Rutsche vom Band 1 aussortiert. Der letzte Puck mit Bohrung oben geht den gleichen Weg wie der erste Puck und erreicht ohne weiteren Ereignisse das Ende vom Band 2.</p>
PMU (wird auf dem Band zwei nicht umgedreht) – PBO – PMU (wird auf dem Band 2 umgedreht)	<p>Der erste Puck wird auf Band eins gelegt die Höhenmessung erkennt kein Loch, Höhe ist ok, die Weiche öffnet sich und sobald der Puck im Auslauf ist wird das Band getoppt und muss umgedreht werden. Das Laufband bleibt solange stehen bis das erste Werkstück umgedreht wurde.</p> <p>Wenn Band zwei frei ist wird der erste Puck auf Band zwei befördert und fährt bis zum Metallsensor vor, dieser Erkennt die Bohrung mit Metall und der Puck wird zurück zum Eingangslichtsensor gefahren vom Band genommen jedoch nicht umgedreht. Dann wird er wieder zum Metallsensor gefahren und von der Weiche aussortiert. Band zwei ist somit wieder frei und Band eins kann seinen</p>

	<p>Betrieb fortsetzen.</p> <p>Der zweite Puck erreicht ohne Probleme das ende von Band zwei.</p> <p>Der Dritte Puck wird am Auslauf von Band eins umgedreht und fährt dann auf Band 2 weiter bis zum Metallsensor, dann wird dieser wieder zum Einlauf gefahren, weil Metall erkannt wurde. Dieser wird dann umgedreht und läuft bis zum Auslauf von Band zwei durch.</p>
Zuviele Pucks hintereinander	<p>Die Pucks werden mit sehr kurzem Abstand hintereinander auf das Band eins gelegt, dadurch wird das Band eins in einen Fehlerzustand geleitet, das Band stoppt, der Fehler wird quittiert und das Band geräumt nach erneuter quittierung kann das Band den Normalbetrieb wieder aufnehmen.</p>
FP – (ein weiterer Puck wird kurz vor der Höhenmessung aufs Band 1 gelegt)	<p>Der erste Puck wird an Anfang von Band 1 gelegt. Sobald dieser in Höhenmessung ist, wird direkt dahinter ein anderer Puck aufs Band gelegt. Der erste Puck wird als flach erkannt, erreicht jedoch die Rutsche nicht. Das System meldet Fehler. Das Band wird gestoppt und der Fehler quittiert. Danach werden die Pucks vom Band geräumt und es wird erneut quittiert. Danach kann das Band in das Normalbetrieb übergehen.</p>
PMO – PMO – (verschwindet im Auslauf von band 1) - PMO	<p>Der erste Puck wird aufs Band 1 gelegt. Die Höhenmessung und Metallsensor erkennen eine Bohrung mit Metall, Weiche geht auf und im Auslauf muss dieser Puck umgedreht werden. Sobald dies geschehen ist und das Band 2 frei ist, wird der erste Puck auf das Band 2 befördert und erreicht das Ende vom band 2. Währenddessen wird beim zweiten und dritten Puck ebenfalls die Bohrung mit Metall erkannt und nachdem der zweite Puck die Weiche passiert ist, verschwindet dieser vom Band ohne die Auslauflichtschranke zu erreichen. Das System meldet ein Fehler und dieser wird quittiert. Der dritte Puck wird vom Band geräumt und es wird erneut quittiert. Das Band 1 ist leer und kann den Betrieb wieder aufnehmen. Dieser Fehler betrifft den ersten Puck nicht und nach dem dieser das Ende vom Band eins erreicht hat, wird auch Band 2 freigegeben.</p>
PBO (verschwindet vor der Höhenmessung auf dem Band 2) – PBO – PBO - PBO	<p>Die 4 Pucks werden nacheinander auf das Band 1 gelegt. Der erste Puck erreicht das Band 2 und verschwindet vor der Höhenmessung. Die Ampel leuchtet langsam rot und nach 1 Sekunde nimmt das Band 2 normalen Betrieb auf. Die weiteren 3 Pucks erreichen einzeln das Ende vom Band 2.</p>

PBO – Ein Puck wird aufs Band 2 vor dem ersten Puck hinzugefügt	Der erste Puck wird auf das Band 1 gelegt und nimmt seinen Durchlauf auf. Währenddessen wird ein Puck am Anfang vom Band 2 gelegt. Band 2 erkennt, dass zu diesem Puck noch keine Daten vorliegen und meldet den Fehler. Fehler wird quittiert. Der Puck wird entfernt. Nach erneuter Quittierung ist das Band 2 wieder Betriebsbereit. Der erste Puck wird auf das zweite Band befördert und das Ende vom Band 2 erreichen, sobald der Fehler auf dem Band 2 behoben wurde. Solange der Fehler noch vorliegt, wartet der Puck auf dem Band 1 im Auslauf
---	--

### 6.3. Regressionstest

Regressionstests dienen dazu, die grundlegenden Funktionalitäten der Anlage zu testen bzw. sicher zu stellen, dass nach weiteren Implementierungsschritten diese Funktionalitäten immer noch erhalten bleiben.

LED: - LED – Q1 ist an  
- LED – Q1 ist aus  
- LED – Q2 ist an  
- LED – Q2 ist aus  
- LED – Start ist an  
- LED – Start ist aus  
- LED – Reset ist an  
- LED – Reset ist aus

Ampel: - grüne Leuchte geht an und aus  
- gelbe Leuchte geht an und aus  
- rote Leuchte geht an und aus

Weiche: - Weiche öffnen  
- Weiche schließen

Laufband: - Band läuft normal nach rechts  
- Band stoppt  
- Band läuft normal nach links  
- Band stoppt  
- Band läuft langsam nach rechts  
- Band stoppt  
- Band läuft langsam nach links  
- Band stoppt

Sensorik: Wenn ein Sensor ausgelöst wird, gibt es eine entsprechende Info in der Konsole um welchen Sensor es sich handelt.

Dieser Test liefert folgende Ausgaben:

```
LED Q1 on
LED Q1 off
LED Q2 on
LED Q2 off
LED Start on
LED Start off
LED Reset on
LED Reset off
green on
```

green off  
yellow on  
yellow off  
red on  
red off  
switch open  
switch close  
engine right  
engine stop  
engine start  
engine left  
engine stop  
engine start  
engine slow on  
engine slow off  
reset the machine  
Test is finished

## 6.4. Abnahmetest

Für den Abnahmetest sollen die Szenarien durchgespielt werden, die im 6.2.2. vorgestellt sind.

## 6.5. Testplan

06.10.2013 Meilenstone 1

- Planung von Regressionstests

14/21.10.2013 Meilenstone 2

- Testklasse implementieren und die ersten Tests (Regressionstests) durchführen
- Serielle Schnittstelle testen

18.11.2013 Meilenstone 3

- Regressionstest durchführen
- 

25.11.2013 Meilenstone 4

- Regressionstest durchführen
- Callback-Mechanismus, Dispatcher, Registrierung

09.12.2013 Meilenstone 5 und 6

- Regressionstests durchführen
- Beide Laufbänder einzeln und dann zusammen testen
- Funktionalität des gesamten Anlage testen

## 7. Projektplan

Im Rahmen des Projekts wurde für die agile Arbeitsweise entschieden.  
Der ausführliche Projektplan inklusive Zeitplan befindet sich im Anhang.

### 7.1. Verantwortlichkeiten

Philipp Kloth – Konfigurationsmanager, Entwickler, Architekt, Requirementsengineer  
Jannik Schick – Entwickler, Architekt, Requirementsengineer  
Natalia Duske – Projektmanagerin, Dokumentiererin, Entwicklerin, Requirementsengineer  
Rutkay Küpelikilinc – Qualitätsmanager, Entwickler, Requirementsengineer

## 8. Lessons Learned

Zu Beginn des Projekts schien die Aufgabe recht groß zu sein, was sich dann auch im Laufe des Semesters mehrmals bestätigt hatte. Hilfreich war für uns die grobe Strukturierung der Meilensteine die schon vorgegeben war, so konnten wir uns zumindest am Anfang besser orientieren.

Gleich am Anfang haben wir uns für die agile Arbeitsweise entschieden, da aus unserer Sicht uns dieses Model bei der Auslastung während des Semesters die größte Flexibilität bieten konnte und wir dadurch auch nicht gezwungen wurde die Ideen, die aus Unwissenheit entstanden sind, auch so umzusetzen, hätten wir ein anderes Modell gewählt. Diese Entscheidung war aus unserer Sicht richtig. Im Laufe der Zeit haben wir bemerkt, dass am Anfang des Projekts noch recht schwierig ist eigene Stärken und die Stärken der anderer einzuschätzen und Projektfortschritt im Projektplan mit festen Daten festzuhalten. Dies liegt aber mit großer Sicherheit daran, dass dies unser erstes großes Projekt an der Hochschule war und mit dessen Hilfe wir schon ein Gefühl bekommen haben wie schwierig solche Planung zu machen ist.

Bei der Realisierung der einzelnen Meilensteine haben wir auch festgestellt, dass einige Anforderungen unterspezifiziert waren, welche dann durch die Kommunikation mit dem Kunden aber auch mit dem Senior Berater geklärt wurden. Ohne die Vorgaben in Meilensteinen wäre die Aufgabe sicherlich schwieriger umzusetzen, da wir jeden Schritt und den damit verbundenen logischen Zusammenhang selbst ausdenken müssten. Dies hätte uns auf jeden Fall zusätzliche Zeit in Anspruch genommen, die wir nicht wirklich hatten.

Schon am Anfang unserer Arbeit haben wir in unserem Team die Rollen verteilt und diese auch konsequent ausgeführt. So war jedem Teammitglied klar, welche Aufgaben er oder sie zu bewältigen hatte. Kommunikation in unserem Team hat natürlich auch eine große Rolle gespielt, denn um jede kleine Aufgabe lösen zu können, mussten wir die Anforderungen zusammen besprechen und festlegen welche Ergebnisse wir haben wollen und wie wir diese auch erreichen können. Dadurch war die Aufteilung der einzelnen Aufgaben schon wesentlich leichter. Bei der Umsetzung der Aufgaben haben wir auch stets Kontakt zueinander gehabt, da es während des Projekts die eine oder andere Frage unklar war und die wir aber zusammen im Team dann doch lösen konnten. Daher können wir unsere Teamarbeit als erfolgreich bezeichnen. Auch lag unser Erfolg als Team sicherlich daran, dass wir jedem Teammitglied auf seine Fähigkeiten und sein Können vertrauen schenken und haben uns gegenseitig bei der auftauchenden Fragen stets

unterstützt. Tauchte ein Problem auf oder konnte eine Aufgabe nicht bewältigt können, so haben wir in einer Diskussionsrunde gemeinsam nach einer Lösung gesucht und bei der Umsetzung aber uns auch nicht gegenseitig gestört.

Durch diese Arbeitsweise und gute Absprache miteinander entsprachen die Ergebnisse auch meistens unseren Erwartungen.

Die Modellierung stellte in unserem Projekt eine neue Herausforderung dar. Wir haben uns schon ziemlich früh dazu entschieden mit Petrinetzen zu arbeiten, was aus Erfahrungsberichten der vorherigen Semestern meist nicht der Fall war, da bis jetzt meistens nur mit Zustandsautomaten gearbeitet wurde. Diese Entscheidung haben wir auch nicht bereut, denn es stellte sich für uns heraus, dass Petrinetze bei wohl durchdachter Umsetzung eine gute Grundlage für die Implementierung und die Logik der Anlage bildeten. Schwierig war jedoch die Implementierung der Petrinetzen am Anfang. Denn außer einiger Beispiele aus der Vorlesung konnten wir kaum Informationen finden, die uns weiter helfen konnten. So mussten wir erneut auf die Kenntnisse des betreuenden Professors als Senior Berater zugreifen um die Implementierung erfolgreich abzuschließen.

Auch das Erlernen einer neuen Programmiersprache für dieses Projekt stellte keine großen Schwierigkeiten dar, wie am Anfang angenommen. Und wir konnten schon recht schnell und vertraut mit der Sprache umgehen.

Der Zeitmangel war unser größtes Problem, denn neben diesem Projekt hatte jeder von unserem Team noch andere Verpflichtungen an der Hochschule gehabt und die eine oder andere Aufgabe schon für etwas Stress gesorgt hat. Dies führte auch dazu, dass wir oft gezwungen waren längere Zeit an der Hochschule zu verbringen, damit wir die Termine für die Abgabe der Meilensteine einhalten könnten. Und ein großer Minuspunkt, der sich daraus ergeben hat, ist das die eine oder andere Veranstaltung darunter leiden musste, da man sonst mit dem Projekt nicht weiter kam. Dies wäre eventuell ein Punkt, wo man überlegen konnte den Schwierigkeitsgrad aller angebotenen Veranstaltungen in dem Semester die parallel laufen, aneinander anzupassen, so dass frustfreies Lernen möglich gemacht wird.

Im Großen und Ganzen sind wir jedoch mit der Arbeit unseres Teams mehr als zufrieden und können diesen Erfolg auch genießen, da jeder dazu beigetragen hat, dass dieses Projekt für uns möglich ist und wir die Aufgaben zusammen bewältigen können!

## Glossar

Werkstück – zylinderförmige Kunststoffpucks in drei Ausführungen  
Sensoren – Messinstrumente der Anlage  
Förderband, Band – Laufband des Festo Transfersystems  
Weiche – Aktor zur Leitung der Werkstücke  
Rutsche – Lagerplatz für aussortierte Werkstücke  
Bohrung – Aussparung auf der Grundfläche der Werkstücke  
Metalleinsatz – Metallring in der Bohrung der Werkstücke  
Ampel, Signal, Signalanlage – Dreifarbige Lichtanzeige der Anlage  
Stakeholder – Gruppen die mit der Anlage interagieren  
Steuerung – Software zur Koordination der Anlage  
Bedienpersonal – auf Bedienung der Anlage geschultes Personal  
Interrupts – asynchrone Nachricht einer Systemkomponente  
Singleton – Modell zur Sicherstellung einer singulären Instanz  
Observer – Modell zur Überwachung und Weiterleitung von Signalen  
Pattern – Modelle oder Muster aus dem Software Engineering  
FIFOPrinzip – Warteschlangenprinzip, wer zu erst kommt wird zuerst bedient  
Hardware – Physikalische Komponenten eines Systems  
Initiallauf – Erster Durchlauf zur Einmessung der Anlage und Timer  
Einlauf – Anfang eines Förderbandes

## Abkürzungen

RDD – Requirements and Design Dokumentation  
SE2P – Software Engineering 2 Praktikum  
WS – Wintersemester  
UML – Unified Modelling Language  
MS – Microsoft  
ID – Identifikationsnummer  
bzw. – beziehungsweise  
HAL – Hardware Abstraction Layer  
LED – Light Emitting Diod  
FIFO – first in first out  
z.B. – zum Beispiel

## Anhänge

- Projektstrukturplan: 2.Entwurf