Dieses RDD Template stellt eine grobe Struktur des Dokuments dar.

Sie können es nach Belieben verändern.

Der rot-kursive Text soll durch Ihren Text ersetzt werden.

Bitte aktualisieren Sie bei jeder Meilenstein-Abnahme das Inhaltsverzeichnis.

Requirements and Design Documentation

(RDD)

Version 001

SE2P – Praktikum – WS13/14

Mulici, Burim, 2043736, Burim.Mulici@haw-hamburg.de

Mulici, Besnik, 2013406, Besnik.Mulici@haw-hamburg.de

El Bebbili, Nebal, 2093133 , Nebal.elbebbili@haw-hamburg.de

Schmidt, Michael, 1871475, [michael.schmidt1@haw-hamburg.de](mailto:michael.schmidt1@haw-hamburg.de)

Änderungshistorie:

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Datum** | **Anmerkungen** |
| 1.00 | Besnik Mulici  Burim Mulici  Nebal el Bebbili | 01.10.2013 | Motivation, Randbedingungen, Entwicklungsumgebung, Werkzeuge, Sprachen und Anforderung erledigt. |
| 1.01 | Besnik Mulici  Burim Mulici  Nebal el Bebbili | 04.10.2013 | Anwendungsfälle |
| 1.02 | Burim Mulici | 29.10.2013 | Zeitplan, System Architektur, Datenmodell, Verhaltensmodell |
| 1.03 | Burim Mulici | 10.12.2013 | Patterns, Unit Test/Komponenten Test, Integration Test/System Test, Regressionstest |
| 1.04 | Burim Mulici | 06.01.2014 | Abnahmetest, Lesson Learned |
|  |  |  |  |

Inhalt

Motivation 3

Randbedingungen 3

Entwicklungsumgebung 3

Werkzeuge 3

Sprachen 3

Requirements und Use Cases 3

Anforderungen 3

Use-Case-Diagramm 3

Design 4

System Architektur 4

Datenmodell 4

Verhaltensmodell 4

Implementierung 4

Algorithmen 4

Patterns 5

Testen 5

Unit Test/Komponenten Test 5

Integration Test/System Test 5

Regressionstest 5

Abnahmetest 5

Testplan 5

Testprotokolle und Auswertungen 6

Projektplan 6

Verantwortlichkeiten 6

PSP und Zeitplan 6

Lessons Learned 6

Glossar 7

Abkürzungen 7

Anhänge 7

1. Motivation

Im Rahmen des Studienganges „ Technische Informatik“ viertes Semester an der HAW-Hamburg, soll im Rahmen des Kurses „Software Engineering 2“ mit einem Projekt absolviert werden.

Das Projekt besteht aus einer Werkstück-Sortieranlage aus zwei Förderbandmodulen, welche über jeweils eigene GEME-Rechner gesteuert werden. Diese Rechner sind über eine serielle Schnittstelle mit einander verbunden.

Stakeholder:

Kunden, Projektleiter, Entwickler, Tester

1. Randbedingungen
   1. Entwicklungsumgebung

* VM & GEME-Rechner mit QNX-Version 6.5.0
* C++-Projekt in Momentics
* MS Windows 7
  1. Werkzeuge
* GitHub
* Visual Paradigm 10.1
* Microsoft Office
* Notepad+
* Werkstück-Sortieranlage (Labor HAW-Hamburg 7. Stock)
* SourceTree
  1. Sprachen
* C++

1. Requirements und Use Cases
   1. Anforderungen

Es gibt 3 Werkstücke:

* Flache Werkstücke
* Werkstück mit Bohrung und Metalleinsatz
* Werkstück mit Bohrung

Auf beiden Bändern sollen die Werkstücke langsam durch die Höhenmessung transportiert

werden.

Beide Bänder sollen jeweils stoppen, wenn sich kein Werkstück auf ihnen befindet.

Die Weichen sind im stromlosen Zustand geschlossen.

**Band 1:**

* Werkstück Zuführung erfolg durch das Einlegen am Anfang von Band 1, durch das Unterbrechen der Lichtschranke.
* Es dürfen sich mehrere Werkstücke auf Band 1 befinden.

* Erkannte flache Werkstücke werden am Band 1 aussortiert.
* Erkannte Metalwerkstücke werden am Band 1 Ende von Hand umgedreht, dazu wird das Band 1 angehalten und die Ampel blinkt Gelb.
* Werkstück mit Bohrung, die nach unten gerichtet eingelegt wurden, müssen am Band 1 Ende vom Bedienpersonal von Hand aus umgedreht werden. Dazu wird das Band 1 angehalten und die Ampel blinkt Gelb.
* Die von den Bändern gesammelten Werkstückdaten sollen am Band 2 Ende an der Console ausgegeben werden:
  + - ID
    - Typ
    - Höhen-Messwert von Band 1
    - Höhen-Messwert von Band 2

**Band 2:**

* Nur wenn Band 2 frei ist, darf ein neues Werkstück vereinzelt transferiert werden.
* Die an Band 1 ermittelten Werkstückdaten werden über die serielle Schnittstelle an Band 2 übermittelt.
* Am Ende von Band 2 sollen nur Werkstücke mit Bohrung ankommen, Metalwerkstücke sollen nach unten und Werkstücke ohne Metaleinsatz sollen nach oben liegen.
* Metalwerkstücke mit Bohrung nach oben werden an dem Bandanfang zurück gefahren, dazu blinkt die Ampel Gelb und das Metalwerkstück muss von Hand umgedreht werden. Wird das Werkstück erneut mit Bohrung nach oben erkannt, dann wird diese mit der Weiche aussortiert.

**Anwendungsszenario 1**

**Titel**: Akzeptiertes Werkstück

**Akteur**: Förderband Arbeiter

**Ziel**: Das Werkstück erreicht das Förderband 2 Ende

**Auslöser**: legen des Werkstücks auf das Forderband vom Arbeiter

**Vorbedingung**:

* Lauffähiges Förderband und Ampel leuchtet Grün
* Erste Lichtschranke muss frei sein zum Werkstück einlegen

**Erfolgsszenario** **1**: **„Bohrung mit Metalleinsatz“**

1. Einlegen des Werkstücks mit Bohrung auf das erste Förderband durch Arbeiter.
2. Erste Lichtschranke wird unterbrochen und erkennt dadurch das ein Werkstück eingelegt wurde.
3. Erstes Förderband beginnt an zulaufen.
4. Höhenmesser vom ersten Förderband erkennt ein Werkstück mit einer Bohrung nach oben.
5. Metallsensor vom ersten Förderband erkennt ein Metall-Werkstück.
6. Weiche wird geöffnet und das Werkstück kann passieren bis zum ende des ersten Förderbands.
7. Förderband wird gestoppt und Ampel blinkt Gelb.
8. Arbeiter dreht das Werkstück um und legt es auf dem zweiten Förderband falls frei.
9. Erste Lichtschranke erkennt ein Werkstück
10. Zweites Förderband beginnt an zulaufen.
11. Höhenmessung des zweiten Förderbands erkennt keine Bohrung
12. Metallsensor erkennt kein Metall im Werkstück
13. Weiche wird geöffnet und Werkstück kann passieren
14. Werkstück erreicht das Förderband Ende
15. ID, Typ und Höhenmessung vom Förderband 1 und 2 werden gespeichert & ausgegeben.
16. Arbeiter nimmt das Werkstück vom Förderband.

**Erfolgsszenario** **2**: **„Bohrung ohne Metalleinsatz nach oben“**

1. Einlegen des Werkstücks mit Bohrung auf das erste Förderband durch Arbeiter.
2. Erste Lichtschranke wird unterbrochen und erkennt dadurch das ein Werkstück eingelegt wurde.
3. Erstes Förderband beginnt an zulaufen.
4. Höhenmesser vom ersten Förderband erkennt ein Werkstück mit einer Bohrung nach oben.
5. Metallsensor vom ersten Förderband erkennt kein Metall-Werkstück
6. Weiche wird geöffnet und das Werkstück kann passieren bis zum ende des ersten Förderbands.
7. Werkstück wird vom Arbeiter auf das zweite Forderband gelegt, falls frei.
8. Die erste Lichtschranke vom Forderband 2 erkennt das Werkstück
9. Zweites Förderband beginnt an zulaufen.
10. Höhenmessung des zweiten Förderbands erkennt eine Bohrung
11. Metallsensor erkennt kein Metall im Werkstück
12. Weiche wird geöffnet und Werkstück kann passieren
13. Werkstück erreicht das Förderband Ende
14. ID, Typ und Höhenmessung vom Förderband 1 und 2 werden gespeichert & ausgegeben.
15. Arbeiter nimmt das Werkstück vom Förderband.

**Erfolgsszenario** **3**: **„Bohrung ohne Metalleinsatz nach unten“**

1. Einlegen des Werkstücks mit Bohrung auf das erste Förderband durch Arbeiter.
2. Erste Lichtschranke wird unterbrochen und erkennt dadurch das ein Werkstück eingelegt wurde.
3. Erstes Förderband beginnt an zulaufen.
4. Höhenmesser vom ersten Förderband erkennt ein nicht zu flaches Werkstück ohne Bohrung
5. Metallsensor vom ersten Förderband erkennt kein Metall-Werkstück
6. Weiche wird geöffnet und das Werkstück kann passieren bis zum ende des ersten Förderbands.
7. Förderband wird gestoppt und Ampel blinkt Gelb.
8. Werkstück wird vom Arbeiter **umgedreht** und auf das zweite Forderband gelegt, falls frei.
9. Die erste Lichtschranke vom Forderband 2 erkennt das Werkstück
10. Zweites Förderband beginnt an zulaufen.
11. Höhenmessung des zweiten Förderbands erkennt eine Bohrung
12. Metallsensor erkennt kein Metall im Werkstück
13. Weiche wird geöffnet und Werkstück kann passieren
14. Werkstück erreicht das Förderband Ende
15. ID, Typ und Höhenmessung vom Förderband 1 und 2 werden gespeichert & ausgegeben.
16. Arbeiter nimmt das Werkstück vom Förderband.

**Fehlbedienung:**

* Rutsche ist voll
* Verschwinden von Werkstücken
* Hinzufügen von Werkstücken mitten auf dem Band

**Anwendungsszenario 2**

**Titel :** Aussortieren von flachen Werkstücken

**Akteur:** Förderband Arbeiter

**Ziel:** Die flachen Werkstücke werden aussortiert.

**Auslöser:** Höhenmesser erkennt flaches Werkstück

**Vorbedingung**:

* Lauffähiges Förderband und die Ampel leuchtet grün
* Erste Lichtschranke muss frei sein sein zum Werkstück einlegen

**Erfolgsszenario 1:**

**1.** Werkstück wird vom Arbeiter auf das Förderband gelegt

**2.** Lichtschranke des ersten Förderbands erkennt ein Werkstück und Förderband 1 fängt an zulaufen

3. Die Höhenmessung des ersten Förderbands erkennt ein flaches Werkstück

4. Weiche wird nicht geöffnet

5. Werkstück wird aussortiert und landet in der Rutsche

**Fehlbedienung:**

* Rutsche ist voll
* Verschwinden von Werkstücken
* Hinzufügen von Werkstücken mitten auf dem Band

**Anwendungsszenario 3**

**Titel :** Aussortieren von Werkstücken mit falscher Reihenfolge

**Akteur:** Förderband Arbeiter

**Ziel:** Werkstücke mit falscher Reihenfolge werden aussortiert

**Auslöser:** falscher Reihenfolge

**Vorbedingung**:

* Lauffähiges Förderband und die Ampel leuchtet grün
* Erste Lichtschranke muss frei sein zum Werkstück einlegen

**Erfolgsszenario 1:**

**1.** Einlegen des Werkstücks mit Bohrung auf das erste Förderband durch Arbeiter.

**2.** Erste Lichtschranke wird unterbrochen und erkennt dadurch das ein Werkstück eingelegt wurde.

**3.** Erstes Förderband beginnt an zulaufen.

**4.** Höhenmesser vom ersten Förderband erkennt ein Werkstück mit einer Bohrung nach oben.

**5.** Metallsensor vom ersten Förderband erkennt kein Metall-Werkstück

**6.** Weiche wird geöffnet und das Werkstück kann passieren bis zum ende des ersten Förderbands.

**7.** Werkstück wird vom Arbeiter auf das zweite Forderband gelegt, falls frei.

**8.**  Die erste Lichtschranke vom Forderband 2 erkennt das Werkstück

**9.** Zweites Förderband beginnt an zulaufen.

**10.** Höhenmessung des zweiten Förderbands erkennt keine Bohrung

**11.** Metalsensor erkennt kein Metal an dem Werkstück

**12.** Weiche wird nicht geöffnet und Werkstück wird aussortiert

**Fehlbedienung:**

* Rutsche ist voll
* Verschwinden von Werkstücken
* Hinzufügen von Werkstücken mitten auf dem Band

**Anwendungsszenario 4**

**Titel :** Rückkehr des Werkstücks mit Metalleinsatz

**Akteur:** Förderband Arbeiter

**Ziel:** Metall Werkstücke bekommen im zweiten Förderband eine zweite Chance

**Auslöser:** Höhenmesser & Metallsensor

**Vorbedingung**:

* Lauffähiges Förderband und die Ampel leuchtet grün
* Werkstück mit Metall das im ersten Förderband nach unten gezeigt hat.

**Erfolgsszenario 1:**

1. Einlegen des Werkstücks mit Bohrung auf das erste Förderband durch Arbeiter.
2. Erste Lichtschranke wird unterbrochen und erkennt dadurch das ein Werkstück eingelegt wurde.
3. Erstes Förderband beginnt an zulaufen.
4. Höhenmesser vom ersten Förderband erkennt ein Werkstück mit einer Bohrung nach unten.
5. Metallsensor vom ersten Förderband erkennt kein Metall-Werkstück.
6. Weiche wird geöffnet und das Werkstück kann passieren bis zum ende des ersten Förderbands.
7. Förderband wird gestoppt und Ampel blinkt Gelb.
8. Arbeiter dreht das Werkstück um und legt es auf dem zweiten Förderband falls frei.
9. Erste Lichtschranke erkennt ein Werkstück
10. Zweites Förderband beginnt an zulaufen.
11. Höhenmessung des zweiten Förderbands erkennt eine Bohrung
12. Metallsensor erkennt Metall im Werkstück
13. Weiche wird nicht geöffnet und Werkstück wird zum Anfang des Förderbands gefahren.
14. Förderband wird gestoppt und Ampel blinkt Gelb.
15. Arbeiter dreht das Werkstück nochmals um.
16. Weiche wird diesesmal geöffnet und das Werkstück erreicht das Ende des Förderbands.
17. ID, Typ und Höhenmessung vom Förderband 1 und 2 werden gespeichert & ausgegeben.
18. Arbeiter nimmt das Werkstück vom Förderband.

**Nachbedingung:** Metall Werkstück wird nur einmal zurück gefahren, beim zweiten mal mit

Bohrung nach oben, wird diese aussortiert.

**Fehlbedienung:**

* Rutsche ist voll
* Verschwinden von Werkstücken
* Hinzufügen von Werkstücken mitten auf dem Band

**Fehlerszenarien Behebung:**

**Fehlerszenario 1:**

**Titel:** Rutsche Voll

**Akteur:** Förderband Arbeiter

**Ziel:** Fehlerbehebung durch Entleerung der Rutsche

**Auslöser:** Sensor meldet Rutsche voll

**Vorbedingungen:**

* Lauffähiges Förderband und die Ampel leuchtet grün
* Erste Lichtschranke muss frei sein zum Werkstück einlegen
* Rutsche ist voll

**Fehlerbehebung:**

1. Förderband stoppt und die Ampel blinkt schnell Rot ( Fehlerzustand: anstehend unquittiert)
2. Arbeiter am Förderband sieht den Fehler und drück Quittierungstaste
3. Ampel hat hat rotes Dauerlicht ( Fehlerzustand: anstehend quittiert)
4. Arbeiter am Förderband entleert die Werkstücke von der Rutsche
5. Arbeiter am Förderband betätigt die Starttaste
6. Ampel leuchtet nicht mehr rot ( Fehlerzustand: ok)

**Nachbedingung:**

* Förderband wird wieder in Betrieb gesetzt und Ampel leuchtet Grün

**Fehlerszenario 2:**

**Titel:** Werkstück verschwindet

**Akteur:** Förderband Arbeiter

**Ziel:** Erkennung von einem Fehlenden Werkstück mitten im Betrieb

**Auslöser:** Sensor meldet, dass ein Werkstück fehlt

**Vorbedingung:**

* Lauffähiges Förderband und die Ampel leuchtet grün
* Erste Lichtschranke muss frei sein zum Werkstück einlegen
* Ein Werkstück wird vom Förderband entfernt

**Fehlerbehebung:**

1. Förderband stoppt und die Ampel blinkt schnell Rot ( Fehlerzustand: anstehend unquittiert)
2. Arbeiter am Förderband sieht den Fehler und drück Quittierungstaste
3. Ampel hat hat rotes Dauerlicht ( Fehlerzustand: anstehend quittiert)
4. Arbeiter am Förderband legt das entfernte Werkstück an dem Anfang vom Förderband
5. Arbeiter am Förderband betätigt die Starttaste
6. Ampel leuchtet nicht mehr rot ( Fehlerzustand: ok)

**Nachbedingung:**

* Förderband wird wieder in Betrieb gesetzt und Ampel leuchtet Grün

**Fehlerszenario 3:**

**Titel:** Werkstück wird mitten auf dem Förderband eingefügt

**Akteur:** Arbeiter am Förderband

**Ziel:** das eingefügte Werkstück wird vom Förderband genommen

**Auslöser:** Sensor meldet, dass ein Werkstück zu viel auf dem Förderband ist

**Vorbedingung:**

* Lauffähiges Förderband und die Ampel leuchtet grün
* Erste Lichtschranke muss frei sein zum Werkstück einlegen
* Ein Werkstück wird mitten auf dem Förderband hinzugefügt

**Fehlerbehebung:**

1. Förderband stoppt und die Ampel blinkt schnell Rot ( Fehlerzustand: anstehend unquittiert)
2. Arbeiter am Förderband sieht den Fehler und drück Quittierungstaste
3. Ampel hat hat rotes Dauerlicht ( Fehlerzustand: anstehend quittiert)
4. Arbeiter entfernt die zusätzlichen Werkstücke vom Band
5. Arbeiter am Förderband betätigt die Starttaste
6. Ampel leuchtet nicht mehr rot ( Fehlerzustand: ok)

**Nachbedingung:**

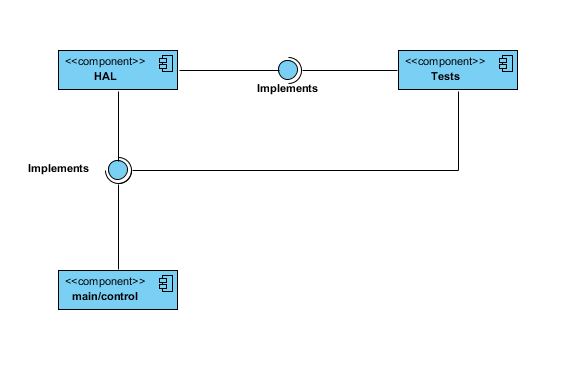
* Förderband wird wieder in Betrieb gesetzt und Ampel leuchtet Grün

1. Design

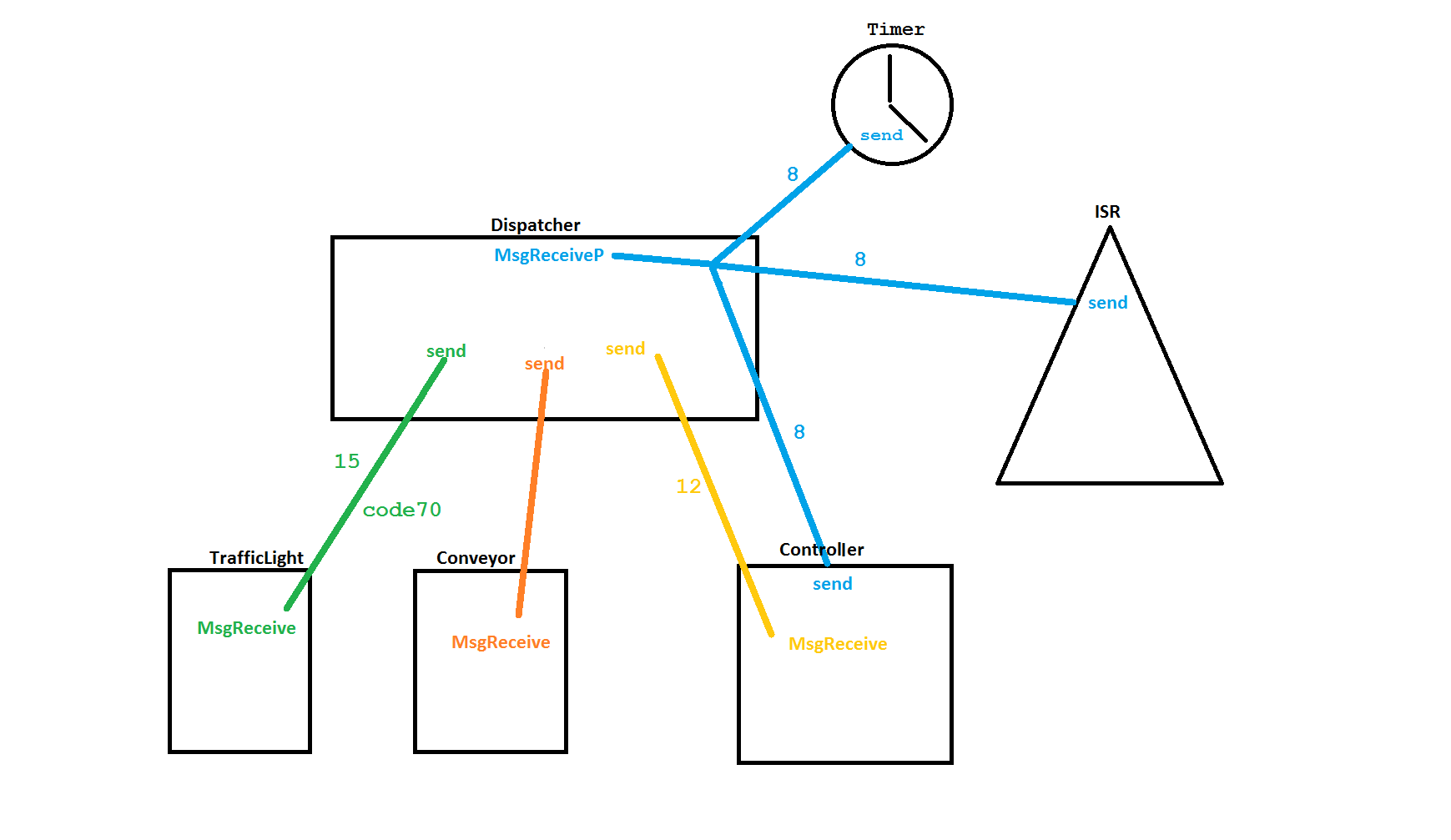
Anmerkung: Die Implementierung MUSS mit Ihrem Design-Modell korrespondieren. Daher ist ein wohlüberlegtes Design wichtig.

* 1. System Architektur

Erstellung der System-Architektur. Geben Sie eine kurze Beschreibung Ihrer Architektur mit den dazugehörenden Komponenten und Schnittstellen.

Spezifikation der Architektur und Definition der System-Schnittstellen in einem UML Komponentendiagramm.

* 1. Datenmodell

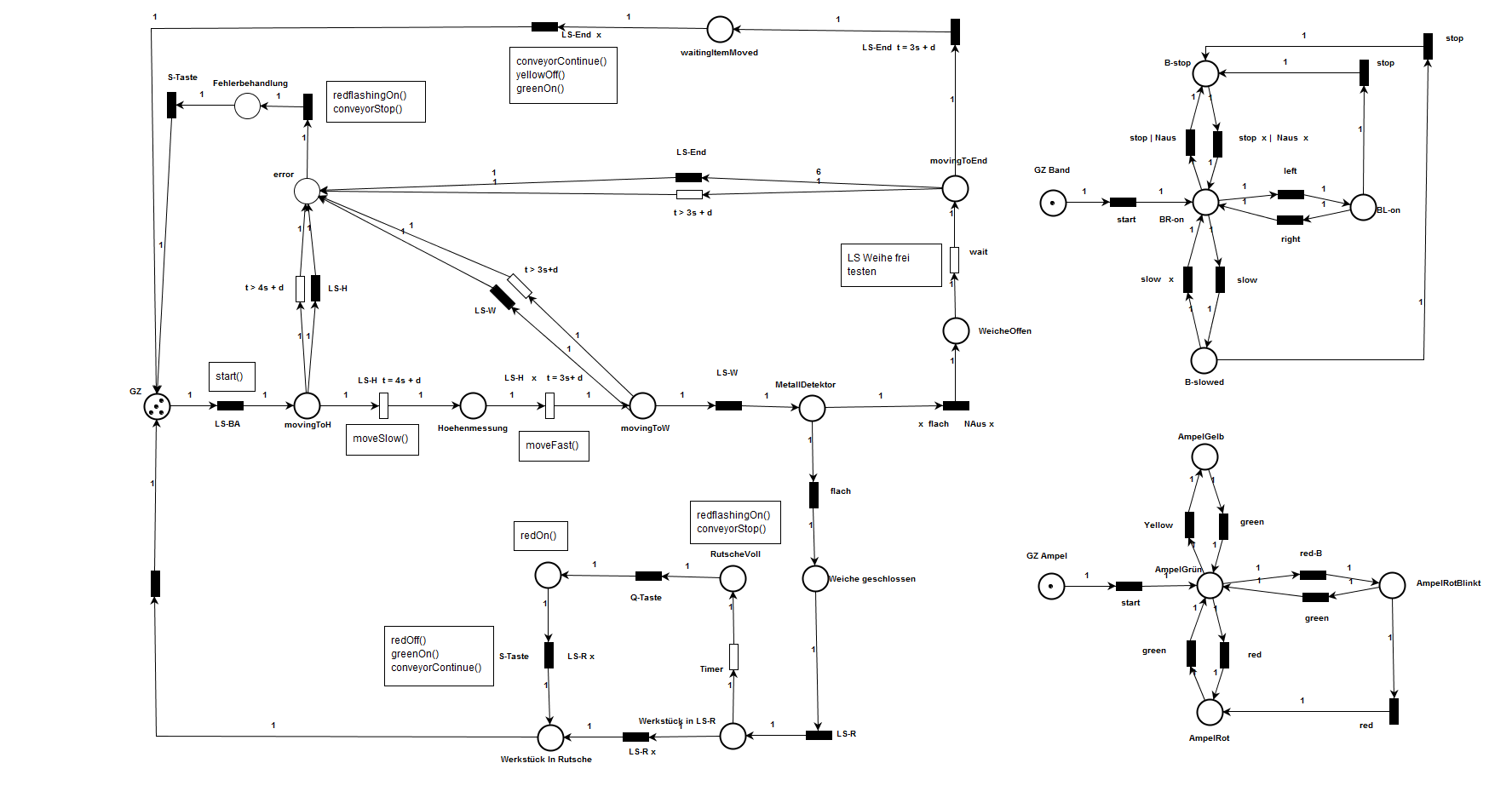


* 1. Verhaltensmodell

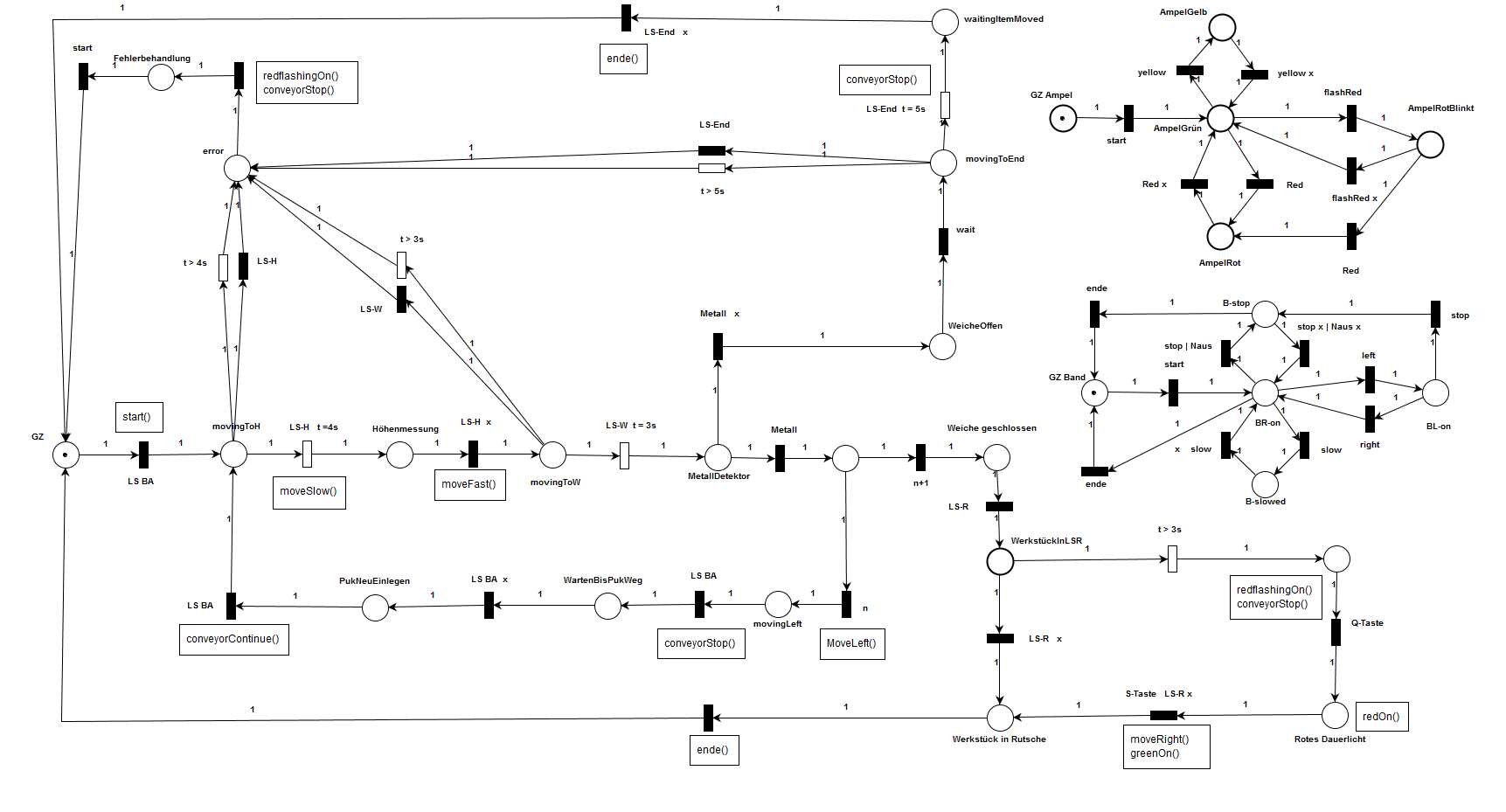
Spezifikation der wichtigsten System-Szenarien anhand von Verhaltensdiagrammen.

Sie können für die Spezifikation der Prozess-Lenkung entweder Petri-Netze oder hierarchische Automaten nehmen.

Band 1 Modelierung



Band 2 Modelierung



**Legende:**

**Petrinetz Bandsteuerung:**

GZ Band - Grundzustand Band

BR-on - Bandrechts on

BL-on - Bandlinks on

Signalname - signal gesetzt

Signalname x - Signal nicht gesetzt

Naus - Notaus

**Petrinetz Band:**

LS-BA – Lichtschranke Bandanfang

LS-H – Lichtschranke Höhenmesser

LS-W – Lichtschranke Weiche

LS-R – Lichtschranke Rutsche

Q-Taste – Quitiertaste

S-Taste – Starttaste

flach – Metalwerstück flach

LS-End – Lichtschranke Bandende

1. Implementierung

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 und bei 0 ist dementsprechend ein Timeout vorhanden.

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.

* 1. Patterns

- Singleton-Pattern

- Reactor-Pattern

1. Testen

Machen Sie sich Gedanken über Unit-Test, Komponententest, Integrationtest, Systemtest, Regressionstest und Abnahmetest.

* 1. Unit Test/Komponenten Test

Band 1:

Sonderfall Rutsche:

- Timeout bei Rutsche voll nach 3 sec wenn Lichtschranke nicht frei gegeben wird

Sonderfall Notaus mit einem Laufband:

- Band hält an und fängt von der letzen Transition wieder an

Höhenmesser:

- erkennen der Puk Typen über die höhe

Aussortieren:

- Flache Puks aussortieren

Akzeptieren:

- Puk mit Loch wird ohne weiteres durchgelassen

- Puk mit Metal muss am ende des Bandes umgedreht werden

- Puk mit Loch nach unten muss am ende des Bandes umgedreht werden

Band 2:

Sonderfall Puk Verschwindet:

- bereich von Lichtschranke Anfang bis Lichtschranke Hoehenmesser

- bereich von Lichtschranke Hoehenmesser bis Lichtschranke Weiche

- bereich von Lichtschranke Weiche von bis Lichtschranke Ende

Sonderfall Puk wurde Hinzugefügt:

- bereich von Lichtschranke Anfang bis Lichtschranke Hoehenmesser

- bereich von Lichtschranke Hoehenmesser bis Lichtschranke Weiche

- bereich von Lichtschranke Weiche von bis Lichtschranke Ende

Sonderfall Rutsche:

- Timeout bei Rutsche voll nach 3 sec wenn Lichtschranke nicht frei gegeben wird

Sonderfall Notaus mit einem Laufband:

- Band hält an und fängt von der letzen Transition wieder an

Höhenmesser:

- erkennen der Puk Typen über die höhe

Aussortieren:

- Puk ohne Loch wird aussortiert.

- Puk mit Metal wird nach der Second Chance aussortiert.

Akzeptieren:

- Puk mit Loch wird ohne weiteres durchgelassen

- Puk mit Metal und loch nach unten

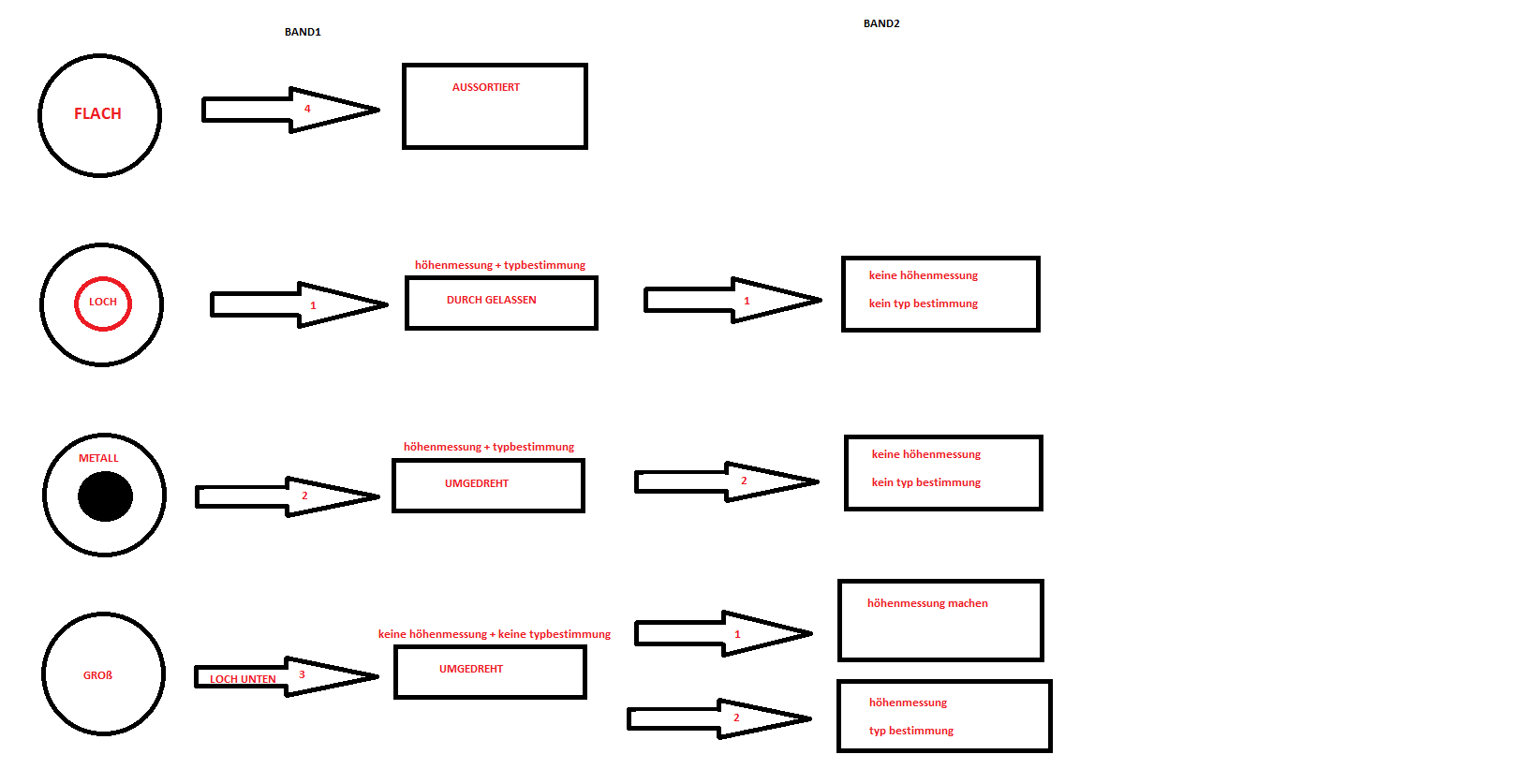
* 1. Integration Test/System Test

Serielle Schnittstelle:

Notaus:

- anhalten beider Bänder beim betätigen des Notaus Tasters

Weiter Senden der Puk Daten für weitere funktionen beider Bänder:



* 1. Regressionstest

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.

- Notaus test auf beiden Bändern

- Puk Typ erkennung auf Band 1 und 2 für alle Akzeptierenden und nicht Akzeptierenden Puks.

- Timer für Rutsche, Puk verschwindet und kommt Hinzu

* 1. Abnahmetest

Erfolgreich Implementiert:

Band 1:

- Puk erkennung und szenarios wie in der Aufgabe gestellt.

- Sonderfall Rutsche voll

- Serielle Schnittstelle zu Band 2 und das weiter reichen der Pukdaten

- Notaus Taster für Band 1 und Band 2 via serielle Schnittstelle

Band 2 :

- Puk erkennung und szenarios wie in der Aufgabe gestellt

- Sonderfall Puk verschwindet

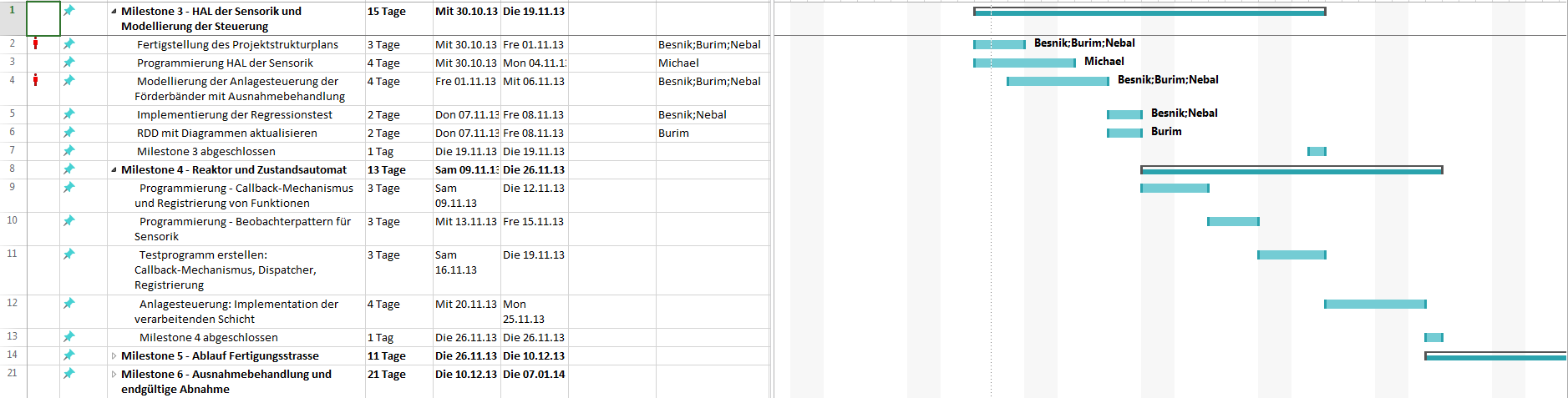
- Sonderfall Puk wird hinzugefügt

- Sonderfall Rutsche voll

- Serielle Schnittstelle zu Band 1 und erkennen der Puk Daten mit dennen weiter gearbeitet werden soll.

- Notaus Taster für Band 2 und Band 1 via Serielle Schnittstelle

1. Projektplan
   1. PSP und Zeitplan



1. Lessons Learned

Die Aufgabe stellte sich für recht groß und das organisieren war für uns am anfang recht anstrengend. Das Zerteilen der Aufgabe in den Meilensteinen hat uns sehr geholfen, da wir so immer einen überblick hatten was getan werden musste und wielange zeit wir noch zur verfügung haben. Das eigentliche Problem kam erst nachdem wir sozusagen die Petrinetze für Band 1 und Band 2 programmieren und Implementieren mussten. Für den Meilenstein 4 wurde dies zur großen aufgabe und wir waren gezwungen unser eigentliches Petrinetz mit einem Simplen Petrinetz zu ersetzen damit wir die funktionalitäten präsentieren konnten. Dieser Meilenstein war der erste der uns in zeitlichen verzug brachte, da wir die organisation nicht streng eigenhalten haben und die Programmierung bzw implementierung noch recht unsauber stattfand.

Ein weiteres Problem wurde deutlich sichtbar beim Meilenstein 5 nachdem wir bereits beide Bänder implementiert hatten und zwar wurden kleine Fehler die nicht sofort sichtbar waren uns zum verhängnis für entliche stunden an arbeit. Diese Fehler kosten uns sehr viel zeit und wir konnten da durch den Meilenstein nicht abgeben und mussten den extra Termin zur kenntnis nehmen.

Diese Probleme kamen durch zu wenig testen zustande. Wir haben das testen der einzelnen änderungen nicht so stark wahrgenommen und mussten dann dies nachhollen um die kleinen fehler auszumetzen.

Am ende des Projektes waren wir sehr gestresst, da wir selbst im letzen Termin nicht vollständig fertig werden konnten und wir waren gezwungen kleine funktionen auszulassen. Mit absprache mit dem Kunden Herrn Fohl ließen wir die Timer für Puk verschwindet und kommt hinzu für Band 1 liegen. Dies ließ uns erkennen das selbst im realen Leben bei großen Projekten nicht immer alles vollständig werden konnte und evtl. mit absprache des Kunden funktionen ausgelassen werden müssen.

Im Großen und ganzen war dies eine große erfahrung, die unter anderem sehr stressig und anstrengen wurde mit dem zeitverzug im rücken, die sehr viel Spaß bereitet hat. Wir haben sehr viel dazu gelernt.

Wenn wir ein Projekt dieser größe erneut hätten würden wir aber sehr viel anders machen was Organisation, Codingstyle und test angeht um nicht nochmal in diesen Zeitverzugen zu geraten.

Glossar

Anhänge

1. Alle Modell-Dateien (Visual Paradigm, Petri-Netze etc.)
2. Source Code und Code Dokumentationen (z.B. Doxygen)
3. Test Protokolle
4. Meeting Protokolle
5. Projektplan
6. etc.