

Requirements and Design Documentation (RDD)

Version: 0.2

SE2P - Praktikum - 2012/13

Hepke, Martin, 2024487, martin.hepke@gmail.com
Sakhri, Mohamed, 1991287, mohamed.sakhri@gmail.com
Dariti, Mahmoud, 1991840, aksilos001@gmail.com
Haleem, Kashif, 1924691, kashoojii@gmail.com

Änderungshistorie:					
Nr.	Datum	Version	Geänderte	Änderung	Autor
1	15.10.20	0.1	Alle	Erstellung	Denis Rycka
2	16.10.20	0.2	-	Gliederung	Martin Hepke
3	16.10.20 12	0.3	3, 6.3	Use Cases, Testfälle, Use	Martin Hepke

Inhalt

1.	Motivation.....	3
2.	Randbedingungen.....	3
2.1.	Entwicklungsumgebung.....	3
2.2.	Werkzeuge.....	3
2.3.	Sprachen.....	3
3.	Requirements und Use Cases.....	3
3.1.	Anforderungen.....	3
3.2.	Use-Case-Diagramm.....	5
4.	Design.....	5
4.1.	System Architektur.....	5
4.2.	Datenmodell.....	5
4.3.	Verhaltensmodell.....	5
5.	Implementierung.....	6
5.1.	Algorithmen.....	6
5.2.	Patterns.....	6
5.3.	Mapping Rules.....	6
6.	Testen.....	6
6.1.	Unit Test/Komponenten Test.....	6
6.2.	Integration Test/System Test.....	6
6.3.	Regressionstest.....	7
6.4.	Abnahmetest.....	7
6.5.	Testplan.....	8
6.6.	Testprotokolle und Auswertungen.....	8
7.	Projektplan.....	8
7.1.	Verantwortlichkeiten.....	8
7.2.	PSP und Zeitplan.....	8
8.	Lessons Learned.....	8
	Glossar.....	9
	Abkürzungen.....	9
	Anhänge.....	9

1. Motivation

Zwei Förderbandmodulen eine Werkstück-Sortieranlage bauen. Jedes Förderbandmodul wird durch einen eigenen GEME-Rechner gesteuert. Die beiden Rechner sind über eine serielle Schnittstelle gekoppelt.

Stakeholder ermitteln.

2. Randbedingungen

2.1. Entwicklungsumgebung

QNX Momentics

2.2. Werkzeuge

2.3. Sprachen

Auflistung der Programmiersprachen und Bibliotheken

3. Requirements und Use Cases

3.1. Anforderungen

Use Case 1	Anlage starten
Akteur	Benutzer
Ziel	Anlage starten oder Betrieb fortsetzen
Auslöser	Benutzer drückt Taster
Vorbedingungen	Anlage vollständig angeschlossen
Nachbedingungen	Anlage muss auf weitere Befehle reagieren können
Erfolgsszenario	S1. Anlage wird gestartet S2. Anlagenstatus und Zustandsanzeige werden aktualisiert S3. Anlage kann weitere Befehle entgegen nehmen
Erweiterungen	-
Alternativfälle	-
Fehlerfälle	F1. Anlage reagiert nicht auf Eingaben
Häufigkeit	Regelmäßig

Use Case 2	Werkstück einlegen
Akteur	Benutzer
Ziel	Werkstück zur Sortierung bereitstellen
Auslöser	Benutzer legt Werkstück auf das Laufband
Vorbedingungen	Anlage gestartet
Nachbedingungen	Werkstück muss bei Sensoren ankommen

Erfolgszenario	S1. Werkstück wird eingelegt S2. Band wird gestartet S3. Intervall x läuft ab S4. Weiteres Werkstück kann eingelegt werden S5. Werkstück wird geprüft S6. Weiche wird gestellt
Erweiterungen	E1. Anlage stoppen E2. Anlagenstatus aktualisieren
Alternativfälle	A5.1. Fehler bei Prüfung A5.2. Anlage stoppen A5.3. Anlagenzustand wird aktualisiert A6.1. Rutsche voll A6.2. Anlage stoppen A6.3. Anlagenzustand wird aktualisiert
Fehlerfälle	F1.1. Werkstück nicht am Anfang abgelegt F3.1. Werkstück zu früh eingelegt F3.2. Werkstück verloren gegangen
Häufigkeit	Regelmäßig

Use Case 3	Rutsche leeren
Akteur	Benutzer
Ziel	Werkstück zur Sortierung bereitstellen
Auslöser	Benutzer sieht das Rutsche voll ist
Vorbedingungen	Ein oder mehrere Werkstücke in Rutsche
Nachbedingungen	Rutsche muss Platz haben für neue Werkstücke
Erfolgszenario	S1. Werkstücke entnehmen
Erweiterungen	-
Alternativfälle	-
Fehlerfälle	-
Häufigkeit	Regelmäßig

Use Case 4	Anlagenzustand aktualisieren
Akteur	Benutzer
Ziel	Fehler quittieren
Auslöser	Benutzer sieht das Fehler aufgetreten ist
Vorbedingungen	Anlagenstatus ändert sich (Fehler tritt auf, Fehler wurde beseitigt, Fehler wurde quittiert, Warnung nach Messung)
Nachbedingungen	Zustandsanzeige muss entsprechend des Anlagenzustands geändert werden
Erfolgszenario	S1. Status ändert sich S2. Zustandsanzeige aktualisieren
Erweiterungen	-
Alternativfälle	-
Fehlerfälle	-
Häufigkeit	Regelmäßig

3.2.Use-Case-Diagramm



4. Design

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

4.1.System Architektur

Erstellung der System-Architektur. Geben Sie eine kurze Beschreibung Ihrer Architektur mit den dazugehörigen Komponenten und Schnittstellen. Spezifikation der Architektur und Definition der System-Schnittstellen in einem UML Komponentendiagramm.

4.2.Datenmodell

Bestimmung des Datenmodells mit Hilfe von UML Klassendiagrammen unter Beachtung der Designprinzipien. Kurze textuelle Beschreibung des Datenmodells und deren wichtigsten Klassen und Methoden.

4.3.Verhaltensmodell

Spezifikation der wichtigsten System-Szenarien anhand von Verhaltensdiagrammen.

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

5. Implementierung

Anmerkung: Wichtige Implementierungsdetails sollen hier erklärt werden.

Code-Beispiele

(snippets) können hier aufgelistet werden, um der Erklärung zu dienen.

Anmerkung: Bitte KEINE ganze Programme hierhin kopieren!

5.1.Algorithmen

Wichtige Algorithmen, die Sie hier benutzt haben.

5.2.Patterns

Wichtige Patterns, die Sie implementiert haben.

5.3.Mapping Rules

Wichtige Mapping Rules, die Sie benutzt haben, z.B. um aus Ihrem Design entsprechenden Code zu erstellen.

6. Testen

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

6.1.Unit Test/Komponenten Test

Test Szenario eines Laufbands.

6.2.Integration Test/System Test

Test Szenarien mit beiden Laufbändern

6.3.Regressionstest

Testfall	Beschreibung	Vorbedingung	Erfolgsfall
1	Ersten Werkstück einlegen	- Band1 gestoppt	- Band startet - Leuchte : grün
2	Werkstück in der Mitte einlegen	- Band1 gestoppt	- Band startet nicht
3	Werkstück in der Mitte einlegen	- Band1 läuft	- Band stoppt - Leuchte : rot - Fehlerzustand aktualisiert
4	Werkstück einlegen	- Band1 läuft - Rutsche voll	- Band stoppt - Leuchte : rot - Fehlerzustand aktualisiert
5	Werkstück aus dem Band entfernen	- Band läuft	- Nach einer bestimmten Zeit Band stoppen - Fehlerzustand aktualisiert
6	Flachen Werkstück einlegen	- Band1 läuft - Rutsche nicht voll	- Werkstück wird aussortiert
7	Werkstück mit Bohrung nach oben und Metalleinsatz einlegen	- Band1 läuft - Rutsche nicht voll - Band2 frei	- Weiche von Band1 öffnen - Band2 startet - Werkstück wird aussortiert
8	Werkstück mit Bohrung nach oben und Metall einsatz einlegen	- Band1 läuft - Rutsche nicht voll - Band2 nicht frei	- Weiche von Band1 öffnen - Band1 stoppt bis Band2 frei ist
9	Werkstück mit Bohrung nach oben ohne Metalleinsatz einlegen	- Band1 läuft - Rutsche nicht voll - Band2 frei	- Weiche von Band1 öffnen - Band2 startet - Weiche von Band2 öffnen

10	Werkstück mit Bohrung nach unten und Metalleinsatz einlegen	<ul style="list-style-type: none"> - Band1 läuft - Rutsche nicht voll - Band2 frei - Bediener dreht den Werkstück um wenn er am Ende von Band1 ist 	<ul style="list-style-type: none"> - Weiche von Band1 öffnen - Band2 startet - Band1 wird angehalten - Leuchte : gelb - Werkstück wird aussortiert
11	Werkstück mit Bohrung nach unten und Metalleinsatz einlegen	<ul style="list-style-type: none"> - Band1 läuft - Rutsche nicht voll - Band2 frei - Werkstück wird am Ende von Band1 nicht umgedreht 	<ul style="list-style-type: none"> - Weiche von Band1 öffnen - Band2 startet - Nach Höhemessung Anlage wird gestoppt - Fehler signalisieren
12	Werkstück mit Bohrung nach unten und Metalleinsatz einlegen	<ul style="list-style-type: none"> - Band1 läuft - Rutsche nicht voll - Band2 frei - Werkstück wird nach einer bestimmten Zeit nicht ans Bandende wieder gelegt 	<ul style="list-style-type: none"> - Weiche von Band1 öffnen - Band2 startet - Fehler signalisieren
Xx

1.1.Abnahmetest

Leiten Sie die Abnahmebedingungen aus den Kunden-Anforderungen her. Geben Sie an, welche Anforderungen erfolgreich und eventuell nicht erfolgreich implementiert sind.

1.2.Testplan

Zeitpunkte für die jeweiligen Teststufen in Ihrer Projektplanung setzen. Dazu können Sie die Meilensteine zu Hilfe nehmen.

1.3.Testprotokolle und Auswertungen

Hier fügen Sie die Test Protokolle bei, auch wenn Fehler bereits beseitigt worden sind, ist es schön zu wissen, welche Fehler einst aufgetaucht sind. Eventuelle Anmerkung zur Fehlerbehandlung kann für weitere Entwicklungen hilfreich sein. Das letzte Testprotokoll ist das Abnahmeprotokoll, das bei der abschließenden Vorführung erstellt wird. Es enthält eine Auflistung der erfolgreich vorgeführten Funktionen des Systems sowie eine Mängelliste mit Erklärungen der Ursachen der Fehlfunktionen und Vorschlägen zur Abhilfe

2. Projektplan

2.1. Verantwortlichkeiten

Verantwortliche innerhalb des Projekts (Projektleiter, Tester, Implementierer, etc.) benennen.

2.2. PSP und Zeitplan

Projektstrukturplan, Ressourcenplan, Zeitplan, Abhängigkeiten von Arbeitspaketen, Eventueller Zeitverzug, etc.

3. Lessons Learned

Was lief gut, was lief schlecht in diesem Projekt (technisch und organisatorisch)?

Was haben Sie gelernt?

Weitere Anregungen und Erkenntnisse durch das Projekt.

Glossar

Eindeutige Begriffserklärungen

Abkürzungen

Listen Sie alle Abkürzungen auf, die Sie in diesem Dokument benutzt haben.

Anhänge

Auflistung aller Artefakten dieses Projekts

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