

Testspezifikation Leitzentrale

Für das studentische Projekt *Sichere Eisenbahnsteuerung*

Datum	18.01.2011
Quelle	Dokumente → 04_Test → 04.01_Testspezifikation
Autoren	Norman Nieß Kai Dziembala
Version	1.0
Status	freigegeben

Testspezifikation Leitzentrale

Copyright (C) 2011 Hochschule Bremen.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

1 Historie

Version	Datum	Autor	Bemerkung
0.0	26.05.2010	Kai Dziembala Norman Nieß	Initialisierung der Testspezifikation
0.1	27.05.2010	Kai Dziembala Norman Nieß	Erstellung der Testfälle 2, 3 (Kapitel 6, 7)
0.2	01.06.2010	Kai Dziembala Norman Nieß	Erstellung des Testfalls 4 (Kapitel 8)
0.3	14.06.2010	Kai Dziembala Norman Nieß	Anpassungen an Review: <ul style="list-style-type: none"> • Umformulierung in Kap. 4 • Umformulierung in Kap 5.3 • Ergänzung in Kap. 6.3 • Umformulierung in Kap. 6.4 • neuer Dokumentenstatus: freigegeben
1.0	17.06.2010	Kai Dziembala	Freigabe der Testspezifikation 'Leitzentrale'

2 Inhaltsverzeichnis

1 Historie.....	2
2 Inhaltsverzeichnis.....	3
3 Identifikation des Testobjekts.....	5
4 Testziele.....	6
5 Testfall 1 „Fahranweisung holen und wiederholen“.....	7
5.1 Identifikation des Testobjektes.....	7
5.2 Test-Identifikation.....	7
5.3 Testfallbeschreibung.....	7
5.4 Testskript.....	7
5.5 Testreferenz.....	9
5.6 Test-Protokoll.....	10
6 Testfall 2 „Befahrbarkeit sicherstellen“.....	11
6.1 Identifikation des Testobjektes.....	11
6.2 Test-Identifikation.....	11
6.3 Testfallbeschreibung.....	11
6.4 Testskript.....	11
6.5 Testreferenz.....	13
6.6 Test-Protokoll.....	13
7 Testfall 3 „Senden einer Nachricht an das Auditing System“.....	14
7.1 Identifikation des Testobjektes.....	14
7.2 Test-Identifikation.....	14
7.3 Testfallbeschreibung.....	14
7.4 Testskript.....	14
7.5 Testreferenz.....	15
7.6 Test-Protokoll.....	16
8 Testfall 4 „Schreiben eines Streckenbefehls“.....	17
8.1 Identifikation des Testobjektes.....	17
8.2 Test-Identifikation.....	17
8.3 Testfallbeschreibung.....	17

8.4 Testskript.....	17
8.5 Testreferenz.....	18
8.6 Test-Protokoll.....	18
9 Auswertung.....	19

3 Identifikation des Testobjekts

Es wird der Programmcode zum Softwaremodul „Leitzentrale“ getestet:

- Leitzentrale.c (Version 1.2, Repository-Nr. 181)
- Leitzentrale.h (Version 1.2, Repository-Nr. 181)

Das Modul 'Leitzentrale' ist Teil der Anwendungsschicht und erzeugt aus den von den Fahrprogrammen abgerufenen Fahranweisungen Streckenbefehle, die jede der beiden Loks ihre im Pflichtenheft vorgegebene Fahraufgabe kollisionsfrei erledigen lassen sollen.

Die erzeugten Streckenbefehle werden an das Modul Befehlsvalidierung der Sicherheitsschicht zur Überprüfung weitergegeben. Von diesem Modul werden auch die Sensordaten empfangen, die Aufschluss über die derzeitige Position der Loks und Wagons auf der Modelleisenbahnstrecke geben. Zur Vermeidung von Redundanzen wird die Streckentopologie sowie Gleisabschnitt-/ Weichenbelegung und Zugposition über das Shared Memory von der Befehlsvalidierung zur Verfügung gestellt.

Die Leitzentrale soll nicht nur Kollisionen vermeiden, sondern auch mögliche Fehler beim An- oder Abkoppeln erkennen und diese durch Wiederholung des Vorgangs beheben. Kritische Zustände, wie zwei Loks auf einem Gleisabschnitt, sollen durch vorausschauendes Sperren von Gleisabschnitten für den jeweils anderen Zug vermieden werden.

4 Testziele

Der Test des Software-Moduls 'Leitzentrale' soll sicherstellen, dass die Streckenbefehle für die Lokomotiven, Weichen und Entkoppler laut dem Modul-Design 'Leitzentrale' und entsprechend dem jeweils aktuellen Streckenabbild erstellt werden. Dies dient dem Gesamtziel, die Fahraufgabe gemäß Pflichtenheft (Kapitel 6) auszuführen.

5 Testfall 1 „Fahranweisung holen und wiederholen“

5.1 Identifikation des Testobjektes

siehe Kapitel 3

5.2 Test-Identifikation

Testname: Test_Fahranweisung_holen+wiederholen

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript → 04.02.05_Leitzentrale

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale

5.3 Testfallbeschreibung

Es wird getestet, ob die Funktion 'getFahranweisung()' wiederholt für die Lokomotive 1 und 2 die korrekten Fahranweisungen zurückgibt. Des Weiteren wird getestet, ob ein Wiederholen von Fahranweisungen, im Fall von misslungenen An- oder Abkoppelmanövern möglich ist.

5.4 Testskript

Notizen:

vor Test der Leitzentrale muss das Modul Fahrprogramm getestet sein!

- Modul 'Fahrprogramm' initialisieren
 - Modul 'Leitzentrale' initialisieren
 - for-Schleife mit drei Durchläufen (→ Testfall dreimal durchlaufen) mit folgendem Inhalt:
 - for-Schleife mit zwei Durchläufen
 - *Schleifenindex* entspricht *Lok*
 - das Flag *wiederholen[Lok]* auf *FALSE* setzen
 - for-Schleife mit drei Durchläufen (drei Fahranweisungen holen)
 - *getFahranweisung()*
 - Vergleich der Rückgabe mit der erwarteten Fahranweisung
-
- *das Flag wiederholen[Lok]* auf *TRUE* setzen
 - for-Schleife mit zwei Durchläufen

-
- *getFahranweisung()*
 - Vergleich der Rückgabe mit der erwarteten Fahranweisung
 - *das Flag wiederholen[Lok]* auf *FALSE* setzen
 - Testergebnis in der Konsole ausgeben

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.05_Leitzentrale'

5.5 Testreferenz

Während des Testdurchlaufs werden die in Tabelle 1 und 2 aufgelisteten Rückgabewerte erwartet.

Für die Schleifendurchläufe mit dem Flag *wiederholen[Lok]=FALSE* gilt *Tabelle 1*:

Schleifen-durchlauf 1. Schleife (Testfalldurchführung)	Schleifen-durchlauf 2. Schleife (Lokomotive)	Schleifendurchlauf 3. Schleife (Fahranweisung)	Zurückzugebende Fahranweisung
1	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]
2	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]
3	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]

Tabelle 1: Erwartete Rückgabewerte für den Testfall 1 mit wiederholen[Lok] = FALSE

Für die Schleifendurchläufe mit dem Flag *wiederholen[Lok]=TRUE* gilt *Tabelle 2*:

Schleifen-durchlauf 1. Schleife (Testfalldurchführung)	Schleifen-durchlauf 2. Schleife (Lokomotive)	Schleifendurchlauf 3. Schleife (Fahranweisung)	Zurückzugebende Fahranweisung
1	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]
2	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]
3	1	1	[0x0] [0x4]
		2	[0x0] [0x5]
		3	[0x0] [0x6]
	2	1	[0x1] [0x1]
		2	[0x1] [0x7]
		3	[0x1] [0x4]

Tabelle 2: Erwartete Rückgabewerte für den Testfall 1 mit wiederholen[Lok] = TRUE

5.6 Test-Protokoll

Das Konsolen-Ergebnis wird in das Dokument 'Protokoll_Test_Leitzentrale' kopiert und diese Datei im Ordner 'Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale' abgelegt.

6 Testfall 2 „Befahrbarkeit sicherstellen“

6.1 Identifikation des Testobjektes

siehe Kapitel 3

6.2 Test-Identifikation

Testname: Test_Befahrbarkeit

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript →
04.02.05_Leitzentrale

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle →
04.03.05_Leitzentrale

6.3 Testfallbeschreibung

Es wird getestet, ob ein wiederholter Aufruf der Funktion *checkfahrbarkeit(byte gleisabschnittsNr)* mit einem identischem Gleisabschnitt, ab dem zweiten Aufruf, die Information *nicht befahrbar (FALSE)* liefert.

In diesem Testfall werden Implizit die internen Funktionen 'checkBelegt(byte gleisabschnittNr)', 'checkGesperrt(byte gleisabschnittNr)', 'setGleisabschnittGesperrt(byte gleisabschnittNr, boolean gesperrt)' und 'getNextWeiche(byte zielNr)' getestet.

6.4 Testskript

Notizen:

- Modul 'Leitzentrale' initialisieren
- Variablen BV_Streckentopologie.prevSwitch, BV_Streckentopologie.nextSwitch, BV_weichenBelegung, BV_gleisBelegung erstellen, fahranweisung.fahrbehl=fahrend setzen
- Variable *gleisabschnittsNr* = 0x4 definieren
- for-Schleife mit drei Durchläufen (→ Testfall dreimal durchlaufen) mit folgendem Inhalt:
 - die Funktion *checkBefahrbarkeit(byte gleisabschnittNr)* wird zweimal hintereinander mit dem Argument *gleisabschnittsNr* aufgerufen
 - die Funktion *checkBefahrbarkeit(byte gleisabschnittNr)* wird erneut mit dem Argument *gleisabschnittsNr* aufgerufen
- Testergebnis wird in der Konsole ausgegeben

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.05_Leitzentrale'

6.5 Testreferenz

Während des Testdurchlaufs werden die in Tabelle 3 dargestellten Rückgabewerte erwartet:

Schleifendurchlauf	Rückgabewert vom 1. Aufruf <i>checkBefahrbarkeit(0x4)</i>	Rückgabewert vom 2. Aufruf <i>checkBefahrbarkeit(0x4)</i>
1	TRUE	FALSE
2	TRUE	FALSE
3	TRUE	FALSE

Tabelle 3: Erwartete Rückgabewerte für den Testfall 2

6.6 Test-Protokoll

Das Konsolen-Ergebnis wird in das Dokument 'Protokoll_Test_Leitzentrale' kopiert und diese Datei im Ordner 'Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale' abgelegt.

7 Testfall 3 „Senden einer Nachricht an das Auditing System“

7.1 Identifikation des Testobjektes

siehe Kapitel 3

7.2 Test-Identifikation

Testname: Test_sendNachricht

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript → 04.02.05_Leitzentrale

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale

7.3 Testfallbeschreibung

Es wird getestet, ob der Aufruf der Funktion *sendNachricht(FehlerCode fehlerCode)* mit einem definiert übergebenen *fehlerCode* eine korrekt zusammengesetzte Auditing-Meldung erzeugt. Die Überprüfung umfasst alle möglichen Fehlercodes, die auftreten können.

7.4 Testskript

Notizen:

- Array mit den acht möglichen Fehler-Codes erzeugen
- Variablen *BV_zugPosition*, *fahranweisung* definieren ([0x0] [0x05])
- globale Methode *sendMsg(MODULE_ID, nachricht)*, welche die Nachricht dem Testprogramm zur Verfügung stellt, erzeugen
- Modul 'Leitzentrale' wird initialisiert
- for-Schleife mit drei Durchläufen (→ Testfall dreimal durchlaufen) mit folgendem Inhalt:
 - for-Schleife mit acht Durchläufen (i=0 bis 7)
 - *sendNachricht(FehlerCode fehlerCode)* mit dem Argument *Array(i)*
 - die Nachricht aus der Methode *sendMsg(MODULE_ID, nachricht)* mit der in dem Array vergleichen
- Testergebnis in der Konsole ausgegeben

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.05_Leitzentrale'

7.5 Testreferenz

Während des Testdurchlaufs werden die in Tabelle 4 dargestellten Rückgabewerte erwartet:

Schleifen- durchlauf 1. Schleife	Schleifen- durchlauf 2. Schleife	in Funktion 'sendNachricht' als Argument übergebener Fehler-Code	nachricht[0]	nachricht[5]
1	1	FEHLER_ZUSTAND [0x00]	[0x00]	[0x0]
	2	FEHLER_FAHRBEFEHL [0x01]	[0x01]	[0x0]
	3	FEHLER_WEICHENSTELLUNG [0x02]	[0x02]	[0x0]
	4	ZIEL_ERREICHT [0x03]	[0x03]	[0x0]
	5	NICHT_BEFAHRBAR [0x04]	[0x04]	[0x5]
	6	KUPPELN_FEHLGESCHLAGEN [0x05]	[0x05]	[0x0]
	7	ANKUPPEL_VERSUCH [0x06]	[0x06]	[0x0]
	8	ABKUPPEL_VERSUCH [0x07]	[0x07]	[0x0]
2	1	FEHLER_ZUSTAND [0x00]	[0x00]	[0x0]
	2	FEHLER_FAHRBEFEHL [0x01]	[0x01]	[0x0]
	3	FEHLER_WEICHENSTELLUNG [0x02]	[0x02]	[0x0]
	4	ZIEL_ERREICHT [0x03]	[0x03]	[0x0]
	5	NICHT_BEFAHRBAR [0x04]	[0x04]	[0x5]
	6	KUPPELN_FEHLGESCHLAGEN [0x05]	[0x05]	[0x0]
	7	ANKUPPEL_VERSUCH [0x06]	[0x06]	[0x0]
	8	ABKUPPEL_VERSUCH [0x07]	[0x07]	[0x0]
3	1	FEHLER_ZUSTAND [0x00]	[0x00]	[0x0]
	2	FEHLER_FAHRBEFEHL [0x01]	[0x01]	[0x0]
	3	FEHLER_WEICHENSTELLUNG [0x02]	[0x02]	[0x0]
	4	ZIEL_ERREICHT [0x03]	[0x03]	[0x0]
	5	NICHT_BEFAHRBAR [0x04]	[0x04]	[0x5]
	6	KUPPELN_FEHLGESCHLAGEN [0x05]	[0x05]	[0x0]
	7	ANKUPPEL_VERSUCH [0x06]	[0x06]	[0x0]
	8	ABKUPPEL_VERSUCH [0x07]	[0x07]	[0x0]

Tabelle 4: Erwartete Rückgabewerte für den Testfall 3

7.6 Test-Protokoll

Das Konsolen-Ergebnis wird in das Dokument 'Protokoll_Test_Leitzentrale' kopiert und diese Datei im Ordner 'Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale' abgelegt.

8 Testfall 4 „Schreiben eines Streckenbefehls“

8.1 Identifikation des Testobjektes

siehe Kapitel 3

8.2 Test-Identifikation

Testname: Test_Streckenbefehl-Schreiben

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript →
04.02.05_Leitzentrale

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle →
04.03.05_Leitzentrale

8.3 Testfallbeschreibung

Es wird getestet, ob die interne Funktion '*setStreckenbefehl(Streckenbefehl streckenbefehl)*' dem Modul 'Befehlsvalidierung' einen Streckenbefehl über den Shared Memory, entsprechend Modul-Design 'Leitzentrale' zur Verfügung stellt.

8.4 Testskript

Notizen:

- LZ_BV_streckenbefehl global definieren
- for-Schleife mit drei Durchläufen (→ Testfall dreimal durchlaufen) mit folgendem Inhalt:
 - for-Schleife mit drei Durchläufen (i=0 bis 2)
 - Streckenbefehl definieren, mit Fahransweisung '0x0' und '0x(i+3)' als Gleisabschnitt
 - Vergleich des abgelegten Streckenbefehls auf dem Shared Memory mit dem definierten Streckenbefehl.
- Testergebnis wird in der Konsole ausgegeben

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.05_Leitzentrale'

8.5 Testreferenz

Während des Testdurchlaufs werden die in Tabelle 4 dargestellten Rückgabewerte erwartet:

Schleifen- durchlauf 1. Schleife	Schleifen- durchlauf 2. Schleife	in Funktion 'setStreckenbefehl' als Argument übergebener Streckenbefehl	Eintrag in den Shared Memory 'LZ_BV_streckenbefehl'
1	1	[0x0] [0x3]	[0x0] [0x3]
	2	[0x0] [0x4]	[0x0] [0x4]
	3	[0x0] [0x5]	[0x0] [0x5]
2	1	[0x0] [0x3]	[0x0] [0x3]
	2	[0x0] [0x4]	[0x0] [0x4]
	3	[0x0] [0x5]	[0x0] [0x5]
3	1	[0x0] [0x3]	[0x0] [0x3]
	2	[0x0] [0x4]	[0x0] [0x4]
	3	[0x0] [0x5]	[0x0] [0x5]

Tabelle 5: Erwartete Shared Memory-Eintragen für den Testfall 4

8.6 Test-Protokoll

Das Konsolen-Ergebnis wird in das Dokument 'Protokoll_Test_Leitzentrale' kopiert und diese Datei im Ordner 'Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle → 04.03.05_Leitzentrale' abgelegt.

9 Auswertung

wird nach Testdurchführung erstellt