

Unit Test Report

for Coffee Machine System

- Test Cases Specification
- Test Summary Report

Project Team

Class A Team 7

Latest update on:

2016-11-06

Team Information

201414134 오세욱

201414136 임현유

201211938 황준익

Table of Contents

1	Introduction	3
1.1	Objectives	3
1.2	References	3
2	Unit test case specification	3
2.1	Test case specification identifier	3
2.2	Test items	3
2.3	Input specifications	3
2.4	Output specifications	3
3	Environmental needs	3
4	Unit test summary report	3
4.1	Test summary report identifier	3
4.2	Evaluation	3

1 Introduction

1.1 Objectives

본 문서는 2016년 건국대학교 소프트웨어 공학개론 강의의 실습과제를 설명한다. 실습과제는 Coffee Machine을 소프트웨어로 구성된 가상의 시스템으로 구현하는 것이다.

1.2 References

SRA_Team7, SDS_Team7

2 Unit test case specification

2.1 Test case specification identifier

Identifier	Input Specification	Output Specification
CMS_UTP_000_000	1.1 Button Sensor Interface	정상 범위 내 (0 ~ 5) 입력 - 입력값 반환 (입력 ; 3)
CMS_UTP_000_001	1.1 Button Sensor Interface	정상 범위 외 !(0 ~ 5) 입력 - 입력값 변환 (-1) 후 반환 (입력 ; 8)
CMS_UTP_001_000	1.2 Setting / Reservation Sensor Interface	flag == 3 첫 번째 입력 (1 ~ 2) 두 번째 입력 (1 ~ 3) - 입력값 배열로 반환 (입력 ; 1, 2)
CMS_UTP_001_001	1.2 Setting / Reservation Sensor Interface	flag == 3 첫 번째 입력 !(1 ~ 2) 두 번째 입력 (1 ~ 3) - 반환 배열의 첫 번째 원소를 0으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; 5, 3)
CMS_UTP_001_002	1.2 Setting / Reservation Sensor Interface	flag == 3 첫 번째 입력 (1 ~ 2) 두 번째 입력 !(1 ~ 3) - 반환 배열의 두 번째 원소를 0으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; 2, 4)
CMS_UTP_001_003	1.2 Setting / Reservation	flag == 3

	Sensor Interface	<p>첫 번째 입력 !(1 ~ 2) 두 번째 입력 !(1 ~ 3) - 반환 배열의 첫 번째, 두 번째 원소를 0 으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; 8, 20)</p>
CMS_UTP_001_004	1.2 Setting / Reservation Sensor Interface	<p>flag == 4 첫 번째 입력 (1 ~ 2) 두 번째 입력 (0000 ~ 2359) - 입력값 배열로 반환 (입력 ; 1, 1823)</p>
CMS_UTP_001_005	1.2 Setting / Reservation Sensor Interface	<p>flag == 4 첫 번째 입력 !(1 ~ 2) 두 번째 입력 (0000 ~ 2359) - 반환 배열의 첫 번째 원소를 0 으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; -1 0824)</p>
CMS_UTP_001_006	1.2 Setting / Reservation Sensor Interface	<p>flag == 4 첫 번째 입력 (1 ~ 2) 두 번째 입력 !(0000 ~ 2359) - 반환 배열의 두 번째 원소를 -1 으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; 2, 2580)</p>
CMS_UTP_001_007	1.2 Setting / Reservation Sensor Interface	<p>flag == 4 첫 번째 입력 !(1 ~ 2) 두 번째 입력 !(1 ~ 3) - 반환 배열의 첫 번째 원소를 0 으로, 두 번째 원소를 -1 으로 변환 후 반환, warning_flag.txt 파일 생성해 4 입력 (입력 ; 5, 8888)</p>
CMS_UTP_002_000	1.3 User Input Detection	<p>iflag == 3 flag == 4 1.2 프로세스 호출 후 해당 반환값 반환 (입력 ; 2, 1)</p>
CMS_UTP_002_001	1.3 User Input Detection	<p>flag != 3 && flag != 4 1.1 프로세스 호출 후 해당 반환값 반환 (입력 ; 3)</p>

CMS_UTP_003_000	1.4 Ingredient Sensor Interface	Ingredient.txt 파일을 열어서 저장된 3개의 정수값 반환 (txt ; 200 30 1)
CMS_UTP_004_000	1.5 User Interface	powder == 0 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 입력값 배열로 반환, powder 값 1로 변경 (입력 ; 3, 2)
CMS_UTP_004_001	1.5 User Interface	powder == 0 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'n' 'N' - 입력값 배열로 반환, powder 값 유지 (입력 ; 9, 1)
CMS_UTP_004_002	1.5 User Interface	powder == 0 첫 번째 입력 !(1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 첫 번째 원소 0으로 변환 후 배열 반환, powder 값 1로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; -3, 6)
CMS_UTP_004_003	1.5 User Interface	Powder == 0 첫 번째 입력 (1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 두 번째 원소 0으로 변환 후 배열 반환, powder 값 1로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; 8 14)
CMS_UTP_004_004	1.5 User Interface	powder == 0 첫 번째 입력 !(1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'n' 'N' - 반환 배열의 첫 번째, 두 번째 원소 0으로 변환 후 배열 반환, powder 값

		<p>유지, warning_flag.txt 파일 생성해 4 입력 (입력 ; 21, -88)</p>
CMS_UTP_004_005	1.5 User Interface	<p>powder == 1 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 입력값 배열로 반환, powder 값 0 으로 변경 (입력 ; 9 1)</p>
CMS_UTP_004_006	1.5 User Interface	<p>powder == 1 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'n' 'N' - 입력값 배열로 반환, powder 값 유 지 (입력 ; 2, 2)</p>
CMS_UTP_004_007	1.5 User Interface	<p>powder == 1 첫 번째 입력 !(1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 첫 번째 원소 0으로 변환 후 배열 반환, powder 값 0 으로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; 89, 10)</p>
CMS_UTP_004_008	1.5 User Interface	<p>powder == 1 첫 번째 입력 (1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 두 번째 원소 0으로 변환 후 배열 반환, powder 값 0 으로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; 3, 22)</p>
CMS_UTP_004_009	1.5 User Interface	<p>powder == 1 첫 번째 입력 !(1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'n' 'N' - 반환 배열의 첫 번째, 두 번째 원소</p>

		0으로 변환 후 배열 반환, powder 값 유지, warning_flag.txt 파일 생성해 4 입력 (입력 ; 100, -100)
CMS_UTP_004_010	1.5 User Interface	powder == 2 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 입력값 배열로 반환, powder 값 0으로 변경 (입력 ; 1 1)
CMS_UTP_004_011	1.5 User Interface	powder == 2 첫 번째 입력 (1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'n' 'N' - 입력값 배열로 반환, powder 값 유지 (입력 ; 3 3)
CMS_UTP_004_012	1.5 User Interface	powder == 2 첫 번째 입력 !(1 ~ 10) 두 번째 입력 (1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 첫 번째 원소 0으로 변환 후 배열 반환, powder 값 0으로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; 99 10)
CMS_UTP_004_013	1.5 User Interface	powder == 2 첫 번째 입력 (1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'y' 'Y' - 반환 배열의 두 번째 원소 0으로 변환 후 배열 반환, powder 값 0으로 변경, warning_flag.txt 파일 생성해 4 입력 (입력 ; 8 14)
CMS_UTP_004_014	1.5 User Interface	powder == 2 첫 번째 입력 !(1 ~ 10) 두 번째 입력 !(1 ~ 10) 세 번째 입력 'n' 'N'

		- 반환 배열의 첫 번째, 두 번째 원소 0으로 변환 후 배열 반환, powder 값 유지, warning_flag.txt 파일 생성해 4 입력 (입력 ; 15 15)
CMS_UTP_005_000	1.6 Ingredients Detection	flag == 1 1.4 호출 후 반환 배열의 세 번째 원소를 인자로 1.5 호출, ret[0] <= 1000 && ret[1] <= 100 - ingredient.txt 파일 생성해 배열 값 저장, 배열 반환 (txt ; 300 20 1 / 입력 ; 2 3 'n')
CMS_UTP_005_001	1.6 Ingredients Detection	flag == 1 1.4 호출 후 반환 배열의 세 번째 원소를 인자로 1.5 호출, ret[0] <= 1000 && ret[1] > 100 - ret[1] 값 100으로 변환 후 ingredient.txt 파일 생성해 배열 값 저장, 배열 반환, warning_plag.txt 파일 생성해 4 저장 (txt ; 300 20 1 / 입력 ; 3 9 'n')
CMS_UTP_005_002	1.6 Ingredients ;Detection	flag == 1 1.4 호출 후 반환 배열의 세 번째 원소를 인자로 1.5 호출, ret[0] > 1000 && ret[1] <= 100 - ret[0] 값 1000 으로 변환 후 ingredient.txt 파일 생성해 배열 값 저장, 배열 반환, warning_plag.txt 파일 생성해 4 저장 (txt ; 300 20 1 / 입력 ; 8 4 'n')
CMS_UTP_005_003	1.6 Ingredients Detection	flag == 0 1.4 호출

		- 배열 반환 (txt ; 300 20 1)
CMS_UTP_006_000	1.7 Setting Interface	setting.txt 파일 열어서 저장된 정수 2개 반환 (txt ; 1 2)
CMS_UTP_007_000	1.8 Current State Interface	state.txt 파일 열어서 저장된 정수 2개 저장 ret[0] == 0! && ret[1] == 0 && progress != empty Queue 의 첫 번째 값 ret[2] 에 저장 후 배열 반환 (txt ; 0 0 / Node ; 1)
CMS_UTP_007_001	1.8 Current State Interface	state.txt 파일 열어서 저장된 정수 2개 저장 ret[0] ==; 0 && ret[1] == 0 && progress == empty ret[2] 에 0 저장 후 배열 반환 (txt ; 0 0 / Node ; empty)
CMS_UTP_007_002	1.8 Current State Interface	state.txt 파일 열어서 저장된 정수 2개 저장 ret[0] != 0 && ret[1] == 0 ret[2] 에 0 저장 후 배열 반환 (txt ; 3 0)
CMS_UTP_007_003	1.8 Current State Interface	state.txt 파일 열어서 저장된 정수 2개 저장 ret[0] == 0 && ret[1] != 0 ret[2] 에 0 저장 후 배열 반환 (txt ; 0 2)
CMS_UTP_007_004	1.8 Current State Interface	state.txt 파일 열어서 저장된 정수 2개 저장 ret[0] != 0 && ret[1] != 0 ret[2] 에 0 저장 후 배열 반환 (txt ; 5 1)
CMS_UTP_008_000	1.9 Reservation Interface	reservation != empty && reservation time 예약 작업 반환 (Node ; 1-1820 / time ; 1820)

CMS_UTP_008_001	1.9 Reservation Interface	reservation != empty && ! reservation time 0 반환 (Node ; 2-1740 / time ; 0800)
CMS_UTP_008_002	1.9 Reservation Interface	reservation == empty 0 반환 (Node ; empty / time ; 0720)
CMS_UTP_009_000	1.10 Storage Data Detection	1.7 프로세스 호출 후 2개의 정수 배열에 저장 1.8 프로세스 호출 후 3개의 정수 배열에 저장 ret[2] != 0 ret[4] != 0 && reservation != 0 - 1.9 프로세스 호출 후 반환 값 progress queue 에 삽입, ret[5] 에 0 저장 후 배열 반환 (setting.txt ; 1 2 / state.txt ; 1 1 / reservation ; 2)
CMS_UTP_009_001	1.10 Storage Data Detection	1.7 프로세스 호출 후 2개의 정수 배열에 저장 1.8 프로세스 호출 후 3개의 정수 배열에 저장 ret[2] != 0 ret[4] != 0 && reservation != 0 - ret[5] 에 0 저장 후 배열 반환 (setting.txt ; 1 2 / state.txt ; 1 1 / reservation ; 1)
CMS_UTP_009_002	1.10 Storage Data Detection	1.7 프로세스 호출 후 2개의 정수 배열에 저장 1.8 프로세스 호출 후 3개의 정수 배열에 저장 ret[2] == 0 && ret[4] == 0 && reservation == 0 - 1.9 프로세스 호출 후 반환값 ret[5] 에 저장, 배열 반환 (setting.txt ; 2 3 / state.txt ; 0 0 / Reservation ; 0)
CMS_UTP_009_003	1.10 Storage Data	1.7 프로세스 호출 후 2개의

	Detection	정수 배열에 저장 1.8 프로세스 호출 후 3개의 정수 배열에 저장 ret[2] == 0 && ret[4] == 0 && reservation != 0 - 1.9 프로세스 호출 후 반환 값 ret[5] 에 저장, 배열 반환 (setting.txt ; 2 3 / state.txt ; 0 0 / Reservation ; 2)
CMS_UTP_010_000	w_flag = 1	"Water Lack" Message 출력
CMS_UTP_010_001	w_flag = 2	"Bean Lack" Message 출력
CMS_UTP_010_002	w_flag = 3	"Coffee Powder Remained" Message 출력
CMS_UTP_010_003	w_flag = 4	"Syntax Error" Message 출력
CMS_UTP_011_000	Call Clean (2.1.4)	Ingredient.txt -> (water bean powder) wait : 10 sec & state.txt -> (4 2) ingredient.txt -> (w b-500 p) & state.txt -> (0 0)
CMS_UTP_012_000	Coffee(1)	Ingredient.txt -> (w b p) wait : 10 sec & state.txt -> (3 1) ingredient.txt -> (w b-300 p) & State.txt -> (0 1)
CMS_UTP_012_001	Coffee(2)	Ingredient.txt -> (w b p) wait : 10 sec & state.txt -> (3 1) ingredient.txt -> (w b-200 p) & State.txt -> (0 1)
CMS_UTP_012_002	Coffee(3)	Ingredient.txt -> (w b p) wait : 10 sec & state.txt -> (3 1) ingredient.txt -> (w b-100 p) & State.txt -> (0 1)
CMS_UTP_013_000	Call Grind	Ingredient.txt -> (w b p) wait : 5 sec & state.txt -> (1 1) ingredient.txt -> (w b 1) & state.txt -> (0 1)
CMS_UTP_014_000	Call Boil	wait : 10 sec & state.txt -> (2 1) state.txt -> (0 1)
CMS_UTP_015_000	normalDisplay(t, water,	Current State : Ready

	bean, powder, cf_rtime, cl_rtime, 1, 1, 0)	Selected Concentration : Light Selected Temperature : Cold
CMS_UTP_015_001	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 2, 2, 1)	Current State : Grinding Selected Concentration : Normal Selected Temperature : Hot
CMS_UTP_015_002	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 2)	Current State : Boiling Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_015_003	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 3)	Current State : Extracting Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_015_004	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 4)	Current State : Cleaning Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_015_005	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 5)	Current State : Setting Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_015_006	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 6)	Current State : Reservating Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_015_007	normalDisplay(t, water, bean, powder, cf_rtime, cl_rtime, 3, 2, 7)	Current State : Refilling Selected Concentration : Strong Selected Temperature : Cold
CMS_UTP_016_000	ingredient.txt -> 500 10 0 setting.txt-> 1 3 state.txt-> 3 1	warning_flag.txt->0
CMS_UTP_016_001	ingredient.txt -> 500 10 0 setting.txt-> 1 3 state.txt-> 3 1 warning_flag.txt->4	warning_flag.txt->4->(2seconds)->0
CMS_UTP_016_002	ingredient.txt -> 299 10 0 setting.txt-> 1 1 state.txt-> 3 1	warning_flag.txt->1
CMS_UTP_016_003	ingredient.txt	warning_flag.txt->1

	-> 199 10 0 setting.txt-> 1 2 state.txt-> 3 1	
CMS_UTP_016_004	ingredient.txt -> 99 10 0 setting.txt-> 1 3 state.txt-> 3 1	warning_flag.txt->1
CMS_UTP_016_005	ingredient.txt -> 500 5 0 setting.txt-> 1 3 state.txt-> 3 1	warning_flag.txt->2
CMS_UTP_016_006	ingredient.txt -> 500 5 1 setting.txt-> 1 3 state.txt-> 3 1	warning_flag.txt->0
CMS_UTP_016_007	ingredient.txt -> 500 50 2 setting.txt-> 1 3 state.txt-> 3 1	warning_flag.txt->3
CMS_UTP_016_008	ingredient.txt -> 499 50 2 setting.txt-> 1 3 state.txt-> 3 2	warning_flag.txt->1

2.2 Test items

2.1의 Test case specification identifier 참조

2.3 Input specifications

2.1의 Test case specification identifier 참조

2.4 Output specifications

2.1의 Test case specification identifier 참조

3 Environmental needs

CMS의 Unit Test를 위한 환경적 요구사항은 다음과 같다.

- (1) Hardware & Platform : GCC compiler/linker
- (2) CTIP Environment : Cygwin
- (3) Unit Test Tool : Simplectest

4 Unit test summary report

4.1 Test summary report identifier

Identifier	Result
CMS_UTP_000_000	Passed
CMS_UTP_000_001	Passed
CMS_UTP_001_000	Passed
CMS_UTP_001_001	Passed
CMS_UTP_001_002	Passed
CMS_UTP_001_003	Passed
CMS_UTP_001_004	Passed
CMS_UTP_001_005	Passed
CMS_UTP_001_006	Passed
CMS_UTP_001_007	Passed
CMS_UTP_002_000	Passed
CMS_UTP_002_001	Passed
CMS_UTP_003_000	Passed
CMS_UTP_004_000	Passed
CMS_UTP_004_001	Passed
CMS_UTP_004_002	Passed
CMS_UTP_004_003	Passed
CMS_UTP_004_004	Passed
CMS_UTP_004_005	Passed
CMS_UTP_004_006	Passed
CMS_UTP_004_007	Passed
CMS_UTP_004_008	Passed
CMS_UTP_004_009	Passed
CMS_UTP_004_010	Passed

CMS_UTP_004_011	Passed
CMS_UTP_004_012	Passed
CMS_UTP_004_013	Passed
CMS_UTP_004_014	Passed
CMS_UTP_005_000	Passed
CMS_UTP_005_001	Passed
CMS_UTP_005_002	Passed
CMS_UTP_005_003	Passed
CMS_UTP_006_000	Passed
CMS_UTP_007_000	Passed
CMS_UTP_007_001	Passed
CMS_UTP_007_002	Passed
CMS_UTP_007_003	Passed
CMS_UTP_007_004	Passed
CMS_UTP_008_000	Passed
CMS_UTP_008_001	Passed
CMS_UTP_008_002	Passed
CMS_UTP_009_000	Passed
CMS_UTP_009_001	Passed
CMS_UTP_009_002	Passed
CMS_UTP_009_003	Passed
CMS_UTP_010_000	Passed
CMS_UTP_010_001	Passed
CMS_UTP_010_002	Passed
CMS_UTP_010_003	Passed
CMS_UTP_011_000	Passed
CMS_UTP_012_000	Passed
CMS_UTP_012_001	Passed
CMS_UTP_012_002	Passed
CMS_UTP_013_000	Passed
CMS_UTP_014_000	Passed
CMS_UTP_015_000	Passed
CMS_UTP_015_001	Passed

CMS_UTP_015_002	Passed
CMS_UTP_015_003	Passed
CMS_UTP_015_004	Passed
CMS_UTP_015_005	Passed
CMS_UTP_015_006	Passed
CMS_UTP_015_007	Passed
CMS_UTP_016_000	Passed
CMS_UTP_016_001	Passed
CMS_UTP_016_002	Passed
CMS_UTP_016_003	Passed
CMS_UTP_016_004	Passed
CMS_UTP_016_005	Passed
CMS_UTP_016_006	Passed
CMS_UTP_016_007	Passed
CMS_UTP_016_008	Passed

4.2 Evaluation

Total test case : 72개

Passed : 72개

Failed : 0개