

System Testing Plan for Point of Sales System

- Test Plan
- Test Design Specification
- Test Cases Specification

Project Team

Team 4

Latest update on:

2017-11-20

Team Information

산업공학과 201211178 민경훈
산업공학과 201211187 배승현
컴퓨터공학과 201311283 송형선
컴퓨터공학과 201611299 정희승

Table of Contents

1	Introduction	4
1.1	Objectives	4
1.2	Background	4
1.3	Scope	4
1.4	Project plan	4
1.5	Configuration management plan	4
1.6	References	4
2	Test items	5
2.1	Software requirements specification	5
2.2	Software requirements analysis	7
3	Features to be tested	8
4	Features not to be tested	11
5	Approach	11
6	Item pass/fail criteria	12
7	System test design specification	12
7.1	Test design specification identifier	12
7.2	Features to be tested	13
7.3	Approach refinements	13
7.4	Test identification	13
7.5	Feature pass/fail criteria	14
8	System test case specification	14
8.1	Test case specification identifier	14
8.2	Test items	17
8.3	Input specifications	17
	[텍스트 입력]	
	Team 4	2

8.4	Output specifications _____	17
9	Testing tasks _____	17
10	Environmental needs _____	18
11	System test deliverables _____	18
12	Schedules _____	18

1 Introduction

1.1 Objectives

본 문서는 Team 5가 개발한 POS System의 System testing plan(STP)를 위한 문서이다. System Test(ST)를 위해 필요한 활동과 기준에 대한 정의, 최종적인 환경적인 요구사항들에 관하여 세부적으로 명시한다.

1.2 Background

2017년 2학기 소프트웨어 공학 개론 수업에서 개발하는 SASD기법을 이용한다. System requirement specification(SRS)에 기반하여 각각의 requirement를 커버하는 Scenario를 작성하고, Scenario에 기반하여 Team5가 개발한 POS System의 System Testing을 진행한다.

1.3 Scope

이 문서는 TEAM 5의 POS System의 System Test를 수행하기 위한 모든 것을 포함한다. POS System의 System Test를 수행하기 위한 자원과 절차, Test Approach와 Technique, 필요로 하는 환경 및 도구 등을 정의한다. POS는 Unit Test를 성공적으로 거친 각 모듈이 통합된 전체 시스템을 대상으로 하는 Test이며, 요구사항 명세서에 나타난 기능들이 제대로 작동하는 지 Test 한다.

1.4 Project plan

완성된 POS의 SW를 System Testing하기 위해 본 계획서를 작성한다. 본 계획서에 작성된 내용에 기반하여 System Testing을 수행한다. System Testing에 대한 결과를 2017년 11월 22일 소프트웨어 공학개론 실습 수업 때 발표한다.

1.5 Configuration management plan

POS의 실행 및 System Test를 위한 작업을 Cygwin에서 이루어지며, Test code의 변경사항 및 수정 사항은 지속적으로 통합되고 Test 된다.

1.6 References

Team 5 SRA, Team 5 SDS, Team 5 UTP, Team5 UTR, Team 5 readme

2 Test items

2.1 Software requirements specification

<Product Function>

Product functions POST는 캐시 화면(터치 스크린)을 이용해 판매, 환불, 재고 확인을 할 수 있다.

POST는 바코드 스캐너를 통해 상품 또는 영수증의 바코드를 스캔하여 정보를 입력 받는다.

POST는 판매 상황에서 상품의 바코드를 바코드 스캐너로 스캔하여 판매 목록에 상품을 추가하고 상품의 수량을 증가시킬 수 있다.

POST는 판매 상황에서 캐시 화면(터치 스크린)의 입력을 통해 판매할 상품의 수량을 증가, 감소시키거나 상품을 목록에서 제거할 수 있다.

POST는 판매 상황에서 고객 화면을 통해 판매할 상품의 총액을 보여 준다.

POST는 판매를 완료한 후 고객 화면과 캐시 화면에 각각 판매 정보를 출력한다.

POST는 판매를 완료한 후 프린터기를 이용해 판매 영수증을 출력한다.

POST는 판매를 완료한 후 재고 서버의 상품 수량 정보를 업데이트 한다.

POST는 환불 상황에서 영수증의 바코드를 바코드 스캐너로 스캔하여 환불 가능한 영수증인지 판단 할 수 있다.

POST는 환불을 완료한 후 고객 화면과 캐시 화면에 각각 환불 정보를 출력한다.

POST 는 환불을 완료한 후 프린터기를 이용해 환불 영수증을 출력한다.

POST는 환불을 완료한 후 재고 서버의 상품 수량 정보를 업데이트 한다.

POST는 재고 확인 요청이 들어오면 현재 매장의 재고 정보를 화면에 출력한다.

POST는 매일 24시(자정)에 정산이 이루어지며 재고 서버에 있는 하루 동안 있었던 판매, 환불 정보를 프린터기를 이용해 정산 보고서를 출력한다

<User Characteristics>

POST 사용자는 캐셔이다.

사용자는 언제든지 전원을 끌 수 있다.

사용자는 바코드 스캐너를 이용해 상품, 영수증을 스캔할 수 있다.

<Constraints & Assumptions>

POST의 캐셔 화면(터치 스크린) 입력 및 스캐너 스캔 행위는 키보드 입력으로 대체한다.

POST의 모든 모니터의 출력은 Cygwin Console(Cygwin Bash Shell)로 대체한다.

POST의 프린터를 사용한 모든 출력은 텍스트 파일 생성으로 대체한다.

POST 프로그램의 하루는 현실시간 3분으로 가정한다.(현실시간 1초 -> 가상시간 8분)

POST 프로그램에서 시각 표현이 필요한 부분은 "YYYYMMDDhhmm"으로 "연월일시분"으로 표현한다.

- 201709111843

POST 프로그램 종료 후 재실행 시 기존에 진행하던 시각이 이어지도록 한다.

POST에서 다루는 상품은 총 7종류(과자, 아이스크림, 과일, 물, 라면, 음료수, 커피)로 한 정한다.

- 상품의 정보는 하위 표 참고

상품	단가	바코드
과자	1,000	001
아이스크림	1,500	010
과일	3,000	011
물	500	100
라면	800	101
음료수	1,200	110
커피	2,000	111

고객이 환불 요청 시, 영수증 내역의 전체 상품에 대해 환불을 수행한다.

재고 서버는 2개의 텍스트 파일로 대체한다.

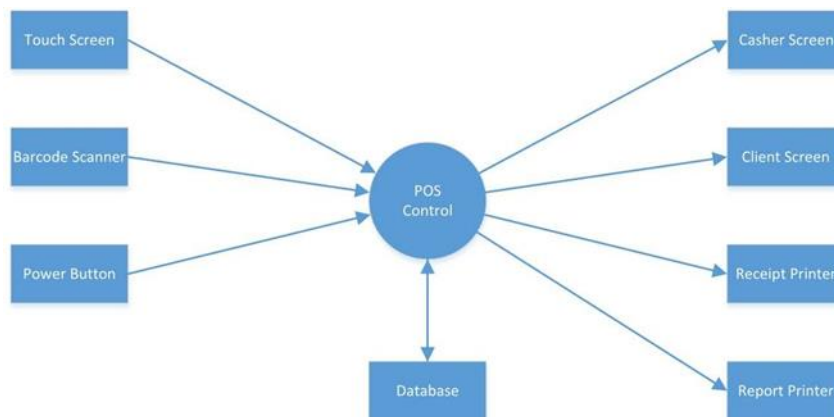
(상품 파일: YYYYMMDD_product.txt, 판매 목록 파일: YYYYMMDD_sale_management.txt)

하루가 지나면 재고 서버의 상품 파일, 판매 관리 파일은 다음 날짜로 새로 생성이 되며 상품 파일의 모든 상품은 100개로 초기화 한다.

POST의 전원이 켜지면 자동으로 재고 서버와 상품을 동기화한다

2.2 Software requirements analysis

<System context diagram>



<Event List>

Input/ Output Event	Description
Touch Sensor Input	터치 센서 입력
Barcode Scan	바코드 스캔 정보
Power Button Input	파워 버튼 입력
Read	데이터 베이스 읽기
Update	데이터베이스 업데이트
Casher Screen Display	캐셔 화면 출력
Client Screen Display	고객 화면 출력
Receipt Print	영수증 출력
Report Print	영수증 출력

3 Features to be tested

3.1 기능 요구사항

SpecID	Description
SpecID_000_001	POST는 캐셔 화면(터치 스크린)을 이용해 판매, 환불, 재고 확인을 할 수 있다.
SpecID_000_002	POST는 바코드 스캐너를 통해 상품 또는 영수증의 바코드를 스캔하여 정보를 입력 받는다.
SpecID_000_003	POST는 판매 상황에서 상품의 바코드를 바코드 스캐너로 스캔하여 판매 목록에 상품을 추가하고 상품의 수량을 증가시킬 수 있다.
SpecID_000_004	POST는 판매 상황에서 캐셔 화면(터치 스크린)의 입력을 통해 판매할 상품의 수량을 증가, 감소시키거나 상품을 목록에서 제거할 수 있다.
SpecID_000_005	POST는 판매 상황에서 고객 화면을 통해 판매할 상품의 총액을 보여 준다.
SpecID_000_006	판매를 완료한 후 고객 화면과 캐셔 화면에 각각 판매 정보를 출력한다.
SpecID_000_007	POST는 판매를 완료한 후 프린터를 이용해 판매 영수증을 출력한다.
SpecID_000_008	POST는 판매를 완료한 후 재고 서버의 상품 수량 정보를 업데이트 한다.
SpecID_000_009	POST는 환불 상황에서 영수증의 바코드를 바코드 스캐너로 스캔하여 환불 가능한 영수증인지 판단 할 수 있다.
SpecID_000_010	POST는 환불을 완료한 후 고객 화면과 캐셔 화면에 각각 환불 정보를 출력한다.
SpecID_000_011	POST는 환불을 완료한 후 프린터를 이용해 환불 영수증을 출력한다.
SpecID_000_012	POST는 환불을 완료한 후 재고 서버의 상품 수량 정보를 업데이트 한다.
SpecID_000_013	POST는 재고 확인 요청이 들어오면 현재 매장의 재고 정보를 화면에 출력한다.

SpecID_000_014	POST는 매일 24시(자정)에 정산이 이루어지며 재고 서버에 있는 하루 동안 있었던 판매, 환불 정보를 프린터기를 이용해 정산 보고서를 출력한다
----------------	---

3.2 판매

SpecID	Description
SpecID_001_001	POST는 기본화면에서 판매를 진행할 수 있다.
SpecID_001_002	POST는 바코드 센서에서 상품의 바코드를 읽을 수 있다.
SpecID_001_003	POST는 바코드로 읽은 상품이 판매 목록에 없는 경우 바코드로 읽은 상품을 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다
SpecID_001_004	해당 정보는 캐시 화면(터치 스크린)과 고객 화면에 각각 업데이트 된다.
SpecID_001_005	POST는 캐시 화면(터치 스크린)을 통해 판매 목록에 있는 상품의 개수를 추가, 감소시키거나 상품을 목록에서 제거 할 수 있다.
SpecID_001_006	해당 정보는 캐시 화면(터치 스크린)과 고객 화면에 각각 업데이트 된다.
SpecID_001_007	POST는 판매하려는 상품의 개수가 상품의 재고 개수보다 부족할 경우 캐시 화면(터치 스크린)에 "XX 개수가 부족합니다" 라는 메시지를 띄운다.
SpecID_001_008	POST는 캐시 화면(터치 스크린)을 통해 고객에게 받은 금액을 입력할 수 있고, 결제를 진행 할 수 있다.
SpecID_001_009	POST는 결제가 완료되면 캐시 화면(터치 스크린)과 고객 화면에 각각 판매 정보를 띄운다.
SpecID_001_010	POST는 판매가 완료되면 영수증 프린터기를 이용해 판매 영수증을 출력한다.
SpecID_001_011	POST는 결제가 완료되면 재고 서버의 상품의 개수를 업데이트 한다. (재고 서버의 당일 상품 파일(YYYYMMDD_product.txt)의 상품의 개수를 업데이트 하는 것으로 대체한다.)
SpecID_001_012	POST는 결제가 완료되면 재고 서버의 판매 정보를 추가한다.
SpecID_001_013	판매 영수증 파일은 sale_YYYYMMDDhhmm.txt형식이고, 양식은 SRS를 따른다.

3.3 환불

SpecID	Description
SpecID_002_001	POST는 사용자가 캐시 화면(터치 스크린)의 기본 화면에서 환불 버튼을 누를 시 환불을 진행할 수 있다.
SpecID_002_003	재고 서버에서 환불이 가능하다는 응답이 올 경우, 캐시 화면(터치 스크

	린)을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다.
SpecID_002_004	재고 서버에서 환불이 불가능하다는 응답이 올 경우, 캐시 화면(터치 스크린) 와 고객 화면에 각각 "해당 영수증은 환불이 불가능 합니다." 라는 메시지를 띄운다.
SpecID_002_005	POST는 환불이 완료되면 캐시 화면(터치 스크린)와 고객 화면에 각각 환불 정보를 띄운다.
SpecID_002_006	POST는 환불이 완료되면 영수증 프린터기를 이용해 환불 영수증을 출력한다.
SpecID_002_007	POST는 환불이 완료되면 재고 서버의 상품의 개수를 업데이트 한다
SpecID_002_008	환불 영수증 파일은 refund_YYYYMMDDhhmm.txt형식이고, 양식은 SRS를 따른다.

3.4 정산

SpecID	Description
SpecID_003_001	POST는 매일 12시가 되면 하루 동안 판매 했던 내역을 재고 서버에서 받아 와 영수증 프린터기를 이용해 출력한다
SpecID_003_002	정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다.
SpecID_003_003	정산 영수증 파일은 settle_YYYYMMDD.txt 형식이고, 양식은 SRS를 따른다.

3.5 재고 확인

SpecID	Description
SpecID_004_001	POST는 기본화면에서 재고 확인 버튼 입력 시 재고 확인을 할 수 있다.
SpecID_004_002	POST는 재고 서버에서 당일 재고 정보를 받아와 동기화를 하고 캐시 화면(터치 스크린)에 재고 정보를 출력하고 영수증 프린터기를 이용해 재고 정보를 출력한다.
SpecID_004_003	POST는 판매 중이거나 환불 중에는 재고 확인을 할 수 없다.

3.6 제약 및 가정 요구사항

SpecID	Description
SpecID_007_003	POST의 프린터를 사용한 모든 출력은 텍스트 파일 생성으로 대체한다.
SpecID_007_004	POST 프로그램의 하루는 현실시간 3분으로 가정한다.(현실시간 1초 가상

	시간 8분)
SpecID_007_005	POST 프로그램에서 시각 표현이 필요한 부분은 “YYYYMMDDhhmm”으로 “연월일시분” 으로 표현한다. - 201709111843
SpecID_007_006	POST 프로그램 종료 후 재실행 시 기존에 진행하던 시각이 이어지도록 한다.
SpecID_007_007	POST에서 다루는 상품은 총 7종류(과자, 아이스크림, 과일, 물, 라면, 음료수, 커피)로 한정한다.
SpecID_007_008	고객이 환불 요청 시, 영수증 내역의 전체 상품에 대해 환불을 수행한다.
SpecID_007_009	재고 서버는 2개의 텍스트 파일로 대체한다. (상품 파일: YYYYMMDD_product.txt, 판매 목록 파일: YYYYMMDD_sale_management.txt)
SpecID_007_010	하루가 지나면 재고 서버의 상품 파일, 판매 관리 파일은 다음 날짜로 새로 생성이 되며 상품 파일의 모든 상품은 100개로 초기화 한다.
SpecID_007_011	POST의 전원이 켜지면 자동으로 재고 서버와 상품을 동기화한다.

4 Features not to be tested

외부 센서 및 인터페이스 부분은 Test하지 않으며, 하위 표에 표기된 부분은 Test하지 않는다.

SpecID	Description
SpecID_002_002	POST는 바코드 센서에서 영수증 바코드를 읽었을 때 바코드로 읽은 영수증 이전에 발급 했던 영수증인지 재고 서버에 확인을 요청한다.
SpecID_005_001	버튼에 대한 반응 속도는 1ms 이하이다.
SpecID_006_001	SASD 개발 방법론을 이용하여 분석 및 설계한다
SpecID_007_001	POST의 캐시 화면(터치 스크린) 입력 및 스캐너 스캔 행위는 키보드 입력으로 대체한다.
SpecID_007_002	POST의 모든 모니터의 출력은 Cygwin Console(Cygwin Bash Shell)로 대체한다.

5 Approach

요구사항에 정의되어 있는 기능들을 확인하기 위하여 다음과 같이 9가지 시나리오를 작성한다.

5.1 Scenario

1) 세일모드 들어가서 과자를 선택하고 수량을 조절(수량 1->0)하고 품목을 물로 변경하고

(수량 0->1) 라면 품목을 추가하여 수량을 조절(수량 0->1)하고 돈을 지불하고 잔돈을 받은 뒤 해당 영수증 번호를 입력하여 환불 받는다. 그리고 나서 판매서버에 과자가 없고, 물이 1개, 라면 1개 있는지 확인한다. 또, 환불 전에 영수증에 과자가 안찍히고, 물이 1개, 라면 1개 찍혀 있는지 확인하고 환불 후 환불 영수증의 모든 상품이 환불되었는지 수량 정보를 확인한다. 마지막으로, 재고 서버에서 과자 수량이 100, 물 수량이 100, 라면 수량이 100인지 확인한다.

2) 거래 도중에 하루(3 분)가 끝나면 정산 영수증이 출력되지 않아야 하고, 거래가 끝나야 정산 영수증이 출력되어야 한다. 정산이 끝난 후에는 재고서버에 제품 수량이 모두 100 으로 리셋 되었는지 확인한다.

3) 아이스크림의 수량을 1 개 판매 한 뒤 재고 확인을 한다. 이 때 영수증과 화면을 출력한다.

4) 메인 화면에서 판매, 환불, 재고 확인을 할 수 있다.

5) 과자의 수량을 101 개로 조정하여 판매를 진행한다.

6) 임의의 영수증 번호(11111111)로 환불을 진행한다.

7) 상품의 정보, 서버, 영수증의 형식이 txt 파일이고, 요구사항과 맞는지 확인한다.

8) 프로그램 내에서 시각 표현이 필요한 모든 부분은 YYYYMMDDhhmm 형식인지 확인한다.

9) 종료 후에 POS SYSTEM 의 시간이 계속 이어지고, 재고 서버와 상품이 동기화 되는지 확인한다.

6 Item pass/fail criteria

Functional Test Pass/Fail Criteria : 각각의 입력에 대해 요구사항 혹은 예상결과와 일치 여부

Non-Functional Test Pass/Fail Criteria : Non-Functional Requirement 중 Test한 항목에 대하여 SRA와 일치 여부

7 System test design specification

7.1 Test design specification identifier

T4_POS.STP.Number

7.2 Features to be tested

본 문서의 3. Features to be tested 참조

7.3 Approach refinements

POS는 SRS의 Specific Requirements에 나온 모든 동작을 명세 된 내용과 동일하게 수행해야 한다. 각 시나리오들이 정해진 순서의 입력을 받았을 때, SRS에 명세 된 동작을 수행해야 한다.

7.4 Test identification

Identifier	Feature	Description
T4_POS.STP.000	SpecID_000_002	Scenario1과 관련된 Test
	SpecID_000_003	
	SpecID_000_004	
	SpecID_000_005	
	SpecID_000_006	
	SpecID_000_007	
	SpecID_000_008	
	SpecID_000_009	
	SpecID_000_010	
	SpecID_000_011	
	SpecID_000_012	
	SpecID_001_002	
	SpecID_001_003	
	SpecID_001_004	
	SpecID_001_005	
	SpecID_001_006	
	SpecID_001_008	
	SpecID_001_009	
	SpecID_001_010	
	SpecID_001_011	
	SpecID_002_003	
	SpecID_002_005	
	SpecID_002_006	
	SpecID_002_007	
	SpecID_004_003	

	SpecID_007_008	
T4_POS.STP.001	SpecID_000_014 SpecID_003_001 SpecID_003_002 SpecID_007_004 SpecID_007_010	Scenario2와 관련된 Test
T4_POS.STP.002	SpecID_000_013 SpecID_001_013 SpecID_004_002	Scenario3과 관련된 Test
T4_POS.STP.003	SpecID_000_001 SpecID_001_001 SpecID_002_001 SpecID_004_001	Scenario4과 관련된 Test
T4_POS.STP.004	SpecID_001_007	Scenario5과 관련된 Test
T4_POS.STP.005	SpecID_002_004	Scenario6과 관련된 Test
T4_POS.STP.006	SpecID_001_013 SpecID_003_003 SpecID_003_003 SpecID_007_007 SpecID_007_009	Scenario7과 관련된 Test
T4_POS.STP.007	SpecID_007_005	Scenario8과 관련된 Test
T4_POS.STP.008	SpecID_007_011	Scenario9과 관련된 Test

7.5 Feature pass/fail criteria

각각의 입력에 대해 요구사항에서 명시한 결과 혹은 예상한 결과값과 일치 여부

8 System test case specification

8.1 Test case specification identifier

Identifier	Input specification	output specification
T4_POS.STP.000.000	바코드로 과자(001) 추가한다.	과자가 판매목록에 상품이 추가되었는지 확인한다.
T4_POS.STP.000.001	판매 수량이 1인 과자(001)에 바코드로 과자(001)을 추가적으로 입력받는다.	과자가 판매목록에서 수량이 2개인지 확인한다.
T4_POS.STP.000.002	바코드로 물을 추가로 입	과자 2개, 물1개가 캐셔화

	력받는다.	면과 고객화면에 업데이트 되어있는지 확인한다.
T4_POS.STP.000.003	수량증가를 통해 라면을 1 개증가시킨다.	과자 2개, 물1개, 라면1개 가 캐쉬화면과 고객화면에 업데이트 되어있는지 확인 한다.
T4_POS.STP.000.004	목록제거를 통해 과자를 모두 제거한다.	물1개, 라면1개로 캐쉬화면 과 고객화면에 업데이트 되어있는지 확인한다.
T4_POS.STP.000.005	결제를 완료하기전에 재고 확인기능을 실시한다.	판매중에는 재고확인을 할 수없다는 메시지를 출력한다.
T4_POS.STP.000.006	고객에게 1300원을 입력 받는다.	판매금액 1300원, 받은금액 1300원, 거스름돈 0을 화면에 출력하고 판매영수 증을 출력한다.
T4_POS.STP.000.007	결제 완료시킨다.	재고 서버의 상품의 개수를 업데이트한다.
T4_POS.STP.000.008	결제 완료시킨다.	재고 서버의 판매정보를 추가한다.
T4_POS.STP.000.009	바코드로 환불처리가 아직 되지 않은 환불영수증을 입력받는다.	환불을 할것인지 안할것인지에대한 메시지를 출력한다.
T4_POS.STP.000.010	환불을 완료시킨다.	캐쉬화면과 고객화면에 환불 정보를 띄우고 환불 영수증을 출력한다.
T4_POS.STP.000.011	환불을 완료시킨다.	재고 서버의 상품의 개수를 업데이트 한다.
T4_POS.STP.001.000	새로운 3분이 넘어가기 전에 환불 기능을 실시하고 3분이 넘어 갔을때 환불을 완료한다.	환불기능을 수행하는 중에는 정산 기능을 수행하지 않고 환불이 완료되는 시점에 정산 영수증을 출력하고 재고서버에 수량을 모두 100으로 리셋한다.
T4_POS.STP.001.001	새로운 3분이 넘어가기 전에 판매 기능을 실시하고 3분이 넘어 갔을때 환불을 완료한다.	판매기능을 수행하는 중에는 정산 기능을 수행하지 않고 환불이 완료되는 시점에 정산 영수증을 출력

		하고 재고서버에 수량을 모두 100으로 리셋한다.
T4_POS.STP.002.000	과자 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 과자의 재고를 99개로 출력한다.
T4_POS.STP.002.001	아이스크림 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 아이스크림의 재고를 99개로 출력한다.
T4_POS.STP.002.002	과일 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 과일의 재고를 99개로 출력한다.
T4_POS.STP.002.003	물 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 물의 재고를 99개로 출력한다.
T4_POS.STP.002.004	라면 1개를 판매한 후 재고확인을 한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 라면의 재고를 99개로 출력한다.
T4_POS.STP.002.005	음료수 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 음료수의 재고를 99개로 출력한다.
T4_POS.STP.002.006	커피 1개를 판매한 후 재고확인을 한다.	판매시 판매영수증을 출력하고 재고확인시 커피의 재고를 99개로 출력한다.
T4_POS.STP.003.000	메인화면에서 판매기능을 입력한다.	판매기능과 관련된 정보를 화면에 출력한다.
T4_POS.STP.003.001	메인화면에서 환불기능을 입력한다.	환불기능과 관련된 정보를 화면에 출력한다.
T4_POS.STP.003.002	메인화면에서 재고확인기능을 입력한다.	재고확인기능과 관련된 정보를 화면에 출력한다.
T4_POS.STP.004.000	과자의 수량을 103개로 조정한다.	100개를 초과할때마다 재고가 부족하다는 메시지를 출력한다.
T4_POS.STP.004.001	물의 수량을 40개로 조정 한후 과자의 수량을 103개로 조정한다.	100개를 초과할때마다 재고가 부족하다는 메시지를 출력한다.
T4_POS.STP.005.000	환불화면에서 임의의 영수증번호 111111을 입력한	환불이 불가능한 영수증이라는 메시지를 출력한다.

	다.	
T4_POS.STP.005.001	환불화면에서 이미 환불이 완료된 환불영수증 번호를 입력한다.	환불이 불가능한 영수증이라는 메시지를 출력한다.
T4_POS.STP.006.000	판매기능을 실시한다.	판매에 관한 영수증과 상품의정보와 서버의 형식이 txt파일이고 요구사항과 맞는지 확인한다.
T4_POS.STP.006.001	환불기능을 실시한다.	환불에 관한 영수증과 상품의정보와 서버의 형식이 txt파일이고 요구사항과 맞는지 확인한다.
T4_POS.STP.007.000	특별한 인풋이 존재하지 않는다.	프로그램 내 시간표현이 필요한 모든 부분이 YYYYMMDDhhmm 형식인지 확인한다.
T4_POS.STP.008.000	POS 시스템의 종료 입력 후 재실행	종료시간과 같은 시간으로 이어지고 재고서버와 상품이 동기화 되는지 확인

8.2 Test items

8.1table 확인

8.3 Input specifications

8.1table 확인

8.4 Output specifications

8.1table 확인

9 Testing tasks

Task	Predecessor	Special Skills	Effort	Finish Date
STP 작성	PTS 구현 완료			2017-11-15
Test Case specification 작성	Task1	PTS에 대한 이해		2017-11-15
ST를 위한 인력 배치	Task 2	PTS에대한 지식		2017-11-17
ST를 위한 환경 구축	Task 3			2017-11-17
ST 수행	Task 4			2017-11-18
ST report 수행	Task 5			2017-11-19

10 Environmental needs

Team5의 POST_Main.exe를 실행시키기 위한 PC, Cygwin 필요

11 System test deliverables

12 Schedules

-Requirement Specification을 보며 필요한 spec에 따른 SpecID부여

-SpecID를 토대로 test design 및 Test Case 작성

-각각의 test case의 input과 그에 따른 예상 output 작성

-Testing 결과와 예상 Output이 일치하는 지를 비교 판단.