Ver. **1.0.0**

# System Testing Plan

# for POS System

- **Test Plan**
- **Test Design Specification**
- **Test Cases Specification**

**Project Team**
**Team 5**

Latest update on:
**2017-11-20**

---

**201211355 손지웅**
**201611303 조정익**
**201610401 손하영**

# 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 5

3 Features to be tested 5

4 Features not to be tested 5

5 Approach 6

6 Item pass/fail criteria 6

7 System test design specification 6

7.1 Test design specification identifier 6

7.2 Features to be tested 6

7.3 Approach refinements 9

7.4 Test identification 9

7.5 Feature pass/fail criteria 14

8 System test case specification 14

8.1 Test case specification identifier 14

8.2 Test items 14

8.3 Input specifications 19

8.4 Output specifications _____ 19

9 Testing tasks _____ 19

10 Environmental needs _____ 19

11 System test deliverables _____ 19

12 Schedules _____ 19

# 1 Introduction

## 1.1 Objectives

본 문서는 2017년 건국대학교의 소프트웨어 공학 개론 강의의 실습과제를 설명한다. 실습 과제는 Point Of Sale (POS) System을 소프트웨어만을 이용한 가상의 시스템으로 구현하는 것이다.

## 1.2 Background

POS System이란 판매와 관련한 데이터를 일괄적으로 관리하고, 고객정보를 수집하여 부가 가치를 향상시키는 시스템이다.

System Testing은 통합된 각 모듈들이 원래 계획했던 대로 작동하는지, 시스템의 실제 동작과 원래 의도했던 요구사항과는 차이가 없는지 등을 판단하게 된다. 수행 시간, 파일 저장 및 처리 능력, 최대 부하, 복구 및 재시동 능력, 수작업 절차 등을 점검한다.

## 1.3 Scope

POS System 이란 판매와 관련한 데이터를 일괄적으로 관리하고, 고객정보를 수집하여 부가 가치를 향상시키는 시스템이다. 본 프로젝트는 전체 POS System 중 POS 단말기만을 대상으로 구현하는 것으로 규모를 제한한다. 모든 시스템은 SW 만으로 구현하고, HW가 필요한 부분은 SW 모듈을 만들어 가상의 HW를 구현한다.

## 1.4 Project plan

완성된 POS System의 Software(SW)를 System Testing(ST)하기 위한 본 계획서를 작성한다.

본 계획서에 작성된 내용에 기반하여 ST를 수행한다.

ST에 대한 결과를 System Testing Report(STR)를 작성한다.

## 1.5 Configuration management plan

SASD를 이용하여 SW를 구현한 후, System Testing(ST)를 수행한다.

## 1.6 References

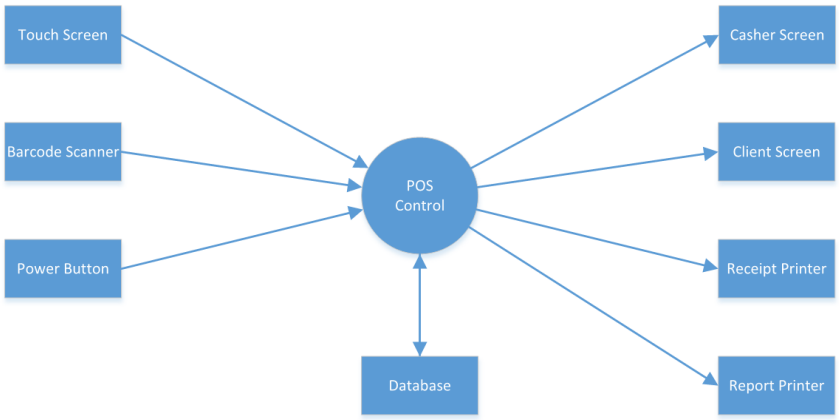
T5_SRA_rev3.0

T5_SDS_rev2.0

## 2 Test items

### 2.1 Software requirements specification

### 2.2 Software requirements analysis




| 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 |
| Casher Screen Display | 데이터베이스 업데이트 |
| Client Screen Display | 캐셔 화면 출력 |
| Receipt Print | 고객 화면 출력 |
| Report Print | 영수증 출력 |
| Stock Update | 정산 보고서 출력 |

## 3 Features to be tested

<Table 1 Testing List> 참고.

## 4 Features not to be tested

SRS에서 요구한 사항 이외의 추가된 사항에 대해서는 Test하지 않는다.

| Category | description |
|---|---|
| 출력 | POST는 캐셔 화면을 이용해 판매, 환불, 재고 확인을 할 수 있다.<br>해당 정보는 캐셔 화면과 고객 화면에 각각 업데이트된다.<br>POST의 모든 모니터의 출력은 Cygwin Console로 대체한다. |

| | |
|---|---|
| | POST의 프린터를 사용한 모든 출력은 텍스트 파일 생성으로 대체한다. |
| 성능 | 버튼에 대한 반응 속도는 1ms 이하이다. |
| 모드 | POST는 기본화면에서 판매를 진행할 수 있다.<br>POST는 사용자가 캐셔 화면의 기본 화면에서 환불 버튼을 누를 시 환불을 진행할 수 있다.<br>정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다.<br>POST는 기본화면에서 재고 확인 버튼 입력 시 재고 확인을 할 수 있다.<br>POST는 판매 중이거나 환불 중에는 재고 확인을 할 수 없다. |
| 상품입력 | POST는 캐셔 화면(터치 스크린)을 이용해 판매, 환불, 재고 확인을 할 수 있다. |
| 환불 모드 | 고객이 환불 요청 시, 영수증 내역의 전체 상품에 대해 환불을 수행한다.<br>POST는 사용자가 캐셔화면의 기본 화면에서 환불 버튼을 누를 시 환불을 진행할 수 있다.<br>POST는 바코드 센서에서 영수증 바코드를 읽었을 때 바코드로 읽은 영수증이 이전에 발급했던 영수증인지 재고 서버에 확인을 요청한다. |

## 5 Approach

CMS는 SRS의 Specification Requirements에 나온 모든 동작을 명세된 내용과 동일하게 수행해야 한다. 요구사항에 정의되어 있는 기능들을 확인하기 위한 시나리오를 작성 후 작성한 시나리오에 기반하여 Testing design및 Test cases를 작성하여 Testing을 수행한다.

## 6 Item pass/fail criteria

pass/fail criteria : 각 시스템의 Feature를 확인하기 위한 시나리오의 Input에 따라 올바른 Output을 도출한다.

## 7 System test design specification

### 7.1 Test design specification identifier

Team5_N_nnn_mmm

### 7.2 Features to be tested

<Table 1 Testing List>

| ID | Description |
|---|---|

<table>
<tr><td>0_000</td><td>판매 모드, 환불 모드, 재고 확인 모드 중 마음대로 전원을 종료할 수 있어야 한다.</td></tr>
<tr><td>1_000</td><td>바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다.</td></tr>
<tr><td>1_001</td><td>캐셔 화면을 통해 판매 목록에 있는 상품의 개수를 추가/감소시키거나 상품을 목록에서 제거할 수 있다.</td></tr>
<tr><td>1_002</td><td>판매하려는 상품의 개수가 상품의 재고 개수보다 부족할 경우 캐셔 화면에 "XX개수가 부족합니다"라는 메시지를 띄운다.</td></tr>
<tr><td>1_003</td><td>캐셔 화면을 통해 고객에게 받은 금액을 입력할 수 있고, 결제를 진행할 수 있다.</td></tr>
<tr><td>1_004</td><td>결제가 완료되면 캐셔화면과 고객화면에 각각 판매 정보를 띄운다.</td></tr>
<tr><td>1_005</td><td>판매가 완료되면 판매 영수증을 출력한다.<br>(sale_YYYYMDDhhmm.txt파일을 생성)</td></tr>
<tr><td>1_006</td><td>결제가 완료되면 재고 서버의 상품의 개수를 업데이트한다.</td></tr>
<tr><td>1_007</td><td>결제가 완료되면 재고 서버의 판매 정보를 추가한다.</td></tr>
<tr><td>1_008</td><td>
<table>
<tr><th colspan="2">캐셔 화면(터치 스크린)</th></tr>
<tr><td colspan="2">Display example</td></tr>
<tr><td>총 판매액</td><td>20,000</td></tr>
<tr><td>현금</td><td>30,000</td></tr>
<tr><td>거스름돈</td><td>10,000</td></tr>
<tr><td>상품</td><td>수량 : 5 / 단가 : 500 / 판매 금액 : 2,500</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>상품</td><td>수량 : 4 / 단가 : 1,000 / 판매 금액 : 4,000</td></tr>
</table>
</td></tr>
<tr><td>1_009</td><td>
<table>
<tr><th colspan="2">고객 화면</th></tr>
<tr><td colspan="2">Display example</td></tr>
<tr><td>총 판매액</td><td>20,000</td></tr>
<tr><td>현금</td><td>30,000</td></tr>
<tr><td>거스름돈</td><td>10,000</td></tr>
</table>
</td></tr>
<tr><td>1_010</td><td>
영수증 양식<br>
영수증 번호 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>
날짜 : YYYY.MM.DD (년.월.일)<br>
판매 상품<br>
상품, 단가, 수량, 판매 금액<br>
...<br>
상품, 단가, 수량, 판매 금액<br>
총 판매액 : x
</td></tr>
<tr><td>2_000</td><td>이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다.</td></tr>
<tr><td>2_001</td><td>이전에 발급하지 않았던 영수증 바코드를 읽었을 때 환불 불가능하다는 응답이 오고, 캐셔 화면과 고객 화면에 각각 "해당 영수증은 환불이 불가능합니다"라는 메시지를 띄운다.</td></tr>
<tr><td>2_002</td><td>환불이 완료되면 환불 영수증을 출력한다.</td></tr>
<tr><td>2_003</td><td>환불이 완료되면 재고 서버의 상품의 개수를 업데이트한다.</td></tr>
</table>

<table>
<tr><td>2_004</td><td>
<table>
<tr><th colspan="2">캐셔 화면(터치 스크린)</th></tr>
<tr><td colspan="2">Display example</td></tr>
<tr><td>판매 날짜</td><td>2017년 09월 07일 05시 20분</td></tr>
<tr><td>환불 금액</td><td>20,000</td></tr>
<tr><td>상품</td><td>수량 : 5 / 단가 : 500 / 판매 금액 : 2,500</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>상품</td><td>수량 : 4 / 단가 : 1,000 / 판매 금액 : 4,000</td></tr>
</table>
</td></tr>
<tr><td>2_005</td><td>
<table>
<tr><th colspan="2">고객 화면</th></tr>
<tr><td colspan="2">Display example</td></tr>
<tr><td>판매 날짜</td><td>2017년 09월 07일 05시 20분</td></tr>
<tr><td>환불 금액</td><td>20,000</td></tr>
</table>
</td></tr>
<tr><td>2_006</td><td>
영수증 양식<br>
영수증 번호 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>
날짜 : YYYY.MM.DD (년.월.일)<br>
환불 상품<br>
상품, 단가, 수량, 판매 금액<br>
...<br>
상품, 단가, 수량, 판매 금액<br>
환불 금액 : x
</td></tr>
<tr><td>3_000</td><td>상품 파일, 판매 목록 파일의 2개의 재고서버가 하루(3분)가 지나면 상품 파일, 판매 관리 파일은 다음 날짜로 새로 생성된다.</td></tr>
<tr><td>3_001</td><td>하루(3분)가 지나면 상품 파일의 모든 상품은 100개로 초기화된다.</td></tr>
<tr><td>3_002</td><td>재고서버에서 당일 재고 정보를 받아와 동기화를 하고 캐셔 화면에 재고 정보를 출력하고 영수증 프린터기를 이용해 재고 정보를 출력한다.</td></tr>
<tr><td>3_003</td><td>판매 중이거나 환불 중에는 재고 확인을 할 수 없다.</td></tr>
<tr><td>3_004</td><td>
재고 확인 양식<br>
날짜 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>
상품, 단가, 재고<br>
...<br>
상품, 단가, 재고
</td></tr>
<tr><td>4_000</td><td>프로그램 종료 후 재실행 시 기존에 진행하던 시각이 이어지도록 한다.</td></tr>
<tr><td>4_001</td><td>현실 시간 3분이 지나면 하루가 넘어가며 (settle_YYYYMMDD.txt)파일을 생성한다.</td></tr>
<tr><td>4_002</td><td>하루 동안 판매했던 내역을 재고 서버에서 받아와 영수증 프린터기를 이용해 출력한다.</td></tr>
<tr><td>4_003</td><td>정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다.</td></tr>
</table>

| | |
|---|---|
| 4_004 | 정산 영수증 양식<br>날짜 : YYYY.MM.DD (년.월.일)<br>판매 상품<br>상품, 단가, 수량, 금액<br>…<br>상품, 단가, 수량, 금액<br>판매 금액 :<br>환불 상품<br>상품, 단가, 수량, 금액<br>…<br>상품, 단가, 수량, 금액<br>환불 금액 : |

## 7.3 Approach refinements

CMS의 각 시스템이 요구사항을 만족하는지 확인하기 위해서 SRS에 정의된 내용에 기반하여 test case를 작성한다.

## 7.4 Test identification

<Table 2 Test Design Identification>

| Identifier | Feature | Valid/Invalid |
|---|---|---|
| 0_000_000 | 판매 모드 중 마음대로 전원을 종료할 수 있어야 한다. | 판매 모드 진행 중 전원을 종료한다. |
| 0_000_001 | 환불 모드 중 마음대로 전원을 종료할 수 있어야 한다. | 환불 모드 진행 중 전원을 종료한다. |
| 0_000_002 | 재고 확인 모드 중 마음대로 전원을 종료할 수 있어야 한다. | 재고 확인 모드 진행 중 전원을 종료한다. |
| 1_000_000 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 001 입력 |
| 1_000_001 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 010 입력 |
| 1_000_002 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 011 입력 |
| 1_000_003 | 바코드로 읽은 상품이 판매 목록 | 바코드 100 입력 |

| | | |
|---|---|---|
| | 에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | |
| 1_000_004 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 101 입력 |
| 1_000_005 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 110 입력 |
| 1_000_006 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 111 입력 |
| 1_000_007 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 000 입력 |
| 1_000_008 | 바코드로 읽은 상품이 판매 목록에 없는 경우 판매 목록에 추가하고, 판매 목록에 있는 경우 해당 상품의 개수를 늘린다. | 바코드 001 입력 후 001 입력 |
| 1_001_000 | 캐셔 화면을 통해 판매 목록에 있는 상품의 개수를 추가/감소시키거나 상품을 목록에서 제거할 수 있다. | 1_000_000 상태에서 과자를 3 개 추가 |
| 1_001_001 | 캐셔 화면을 통해 판매 목록에 있는 상품의 개수를 추가/감소시키거나 상품을 목록에서 제거할 수 있다. | 1_001_000 상태에서 과자를 4 개 감소 |
| 1_001_002 | 캐셔 화면을 통해 판매 목록에 있는 상품의 개수를 추가/감소시키거나 상품을 목록에서 제거할 수 있다. | 1_001_001 상태에서 과자를 1 개 감소시켜 제거 |
| 1_001_003 | 캐셔 화면을 통해 판매 목록에 있는 상품의 개수를 추가/감소시키거나 상품을 목록에서 제거할 수 | 1_001_002 상태에서 과자를 1 개 감소 |

| | | |
|---|---|---|
| | 있다. | |
| 1_002_000 | 판매하려는 상품의 개수가 상품의 재고 개수보다 부족할 경우 캐셔 화면에 "XX개수가 부족합니다"라는 메시지를 띄운다. | 하루가 지나기 전에 001 상품을 102 개 판매하려고 한다. |
| 1_003_000 | 캐셔 화면을 통해 고객에게 받은 금액을 입력할 수 있고, 결제를 진행할 수 있다. | 1_000_000 상태에서 판매 목록의 합계 < 고객에게 받은 돈 |
| 1_003_001 | 캐셔 화면을 통해 고객에게 받은 금액을 입력할 수 있고, 결제를 진행할 수 있다. | 1_000_000 상태에서 판매 목록의 합계 > 고객에게 받은 돈 |
| 1_004_000 | 결제가 완료되면 캐셔화면과 고객 화면에 각각 판매 정보를 띄운다. | 1_003_000 |
| 1_005_000 | 판매가 완료되면 판매 영수증을 출력한다.<br>(sale_YYYYMDDhhmm.txt파일을 생성) | 1_003_000 |
| 1_006_000 | 결제가 완료되면 재고 서버의 상품의 개수를 업데이트한다 | 1_003_000 |
| 1_007_000 | 결제가 완료되면 재고 서버의 판매 정보를 추가한다. | 1_003_000 |
| 1_008_000 | 화면 출력 | 화면 출력 |
| 1_009_000 | 화면 출력 | 화면 출력 |
| 1_010_000 | 화면 출력 | 화면 출력 |
| 2_000_000 | 이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다. | 1_003_000 에서 발급했던 영수증 바코드를 입력하고 환불 취소를 입력한다. |
| 2_000_001 | 이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다. | 2_000_000 에 이어<br>1_003_000 에서 발급했던 영수증 바코드를 입력하고 환불을 완료한다. |
| 2_000_002 | 이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 | 2_000_001 에 이어<br>1_003_000 에서 발급했던 영수증 바코드를 입력한다. |

| | | |
|---|---|---|
| | 수도 있다. | |
| 2_000_003 | 이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다. | 1_000_006 에 이어 판매를 종료한 후 발급했던 영수증 바코드를 입력하고 환불을 완료한다. |
| 2_000_004 | 이전에 발급했던 영수증 바코드를 읽었을 때 환불 가능하다는 응답이 오고, 캐셔 화면을 통해 환불을 완료할 수도 있고 환불을 취소할 수도 있다. | 2_000_003에 이어 1_000_006에서 발급했던 영수증 바코드를 입력한다 |
| 2_001_000 | 이전에 발급하지 않았던 영수증 바코드를 읽었을 때 환불 불가능하다는 응답이 오고, 캐셔 화면과 고객 화면에 각각 "해당 영수증은 환불이 불가능합니다"라는 메시지를 띄운다. | 발급하지 않았던 임의의 영수증 바코드를 입력한다. |
| 2_002_000 | 환불이 완료되면 환불 영수증을 출력한다. | 2_000_001 |
| 2_002_001 | 환불이 완료되면 환불 영수증을 출력한다. | 2_000_003 |
| 2_003_000 | 환불이 완료되면 재고 서버의 상품의 개수를 업데이트한다. | 2_000_001 |
| 2_003_001 | 환불이 완료되면 재고 서버의 상품의 개수를 업데이트한다. | 2_000_003 |
| 2_004_000 | 화면 출력 | 화면 출력 |
| 2_005_000 | 화면 출력 | 화면 출력 |
| 2_006_000 | 화면 출력 | 화면 출력 |
| 3_000_000 | 상품 파일, 판매 목록 파일의 2개의 재고서버가 하루(3분)가 지나면 상품 파일, 판매 관리 파일은 다음 날짜로 새로 생성된다. | 과자(001)의 재고를 98 개로 맞춰놓고 하루(3 분)이 넘어갈 때를 기다린다. |
| 3_000_001 | 상품 파일, 판매 목록 파일의 2개의 재고서버가 하루(3분)가 지나면 상품 파일, 판매 관리 파일은 다음 날짜로 새로 생성된다. | 1_000_006 의 상태에서 판매를 하고 하루(3 분)이 넘어갈 때를 기다린다. |
| 3_001_000 | 하루(3분)가 지나면 상품 파일의 모든 상품은 100개로 초기화된다. | 과자(001)의 재고를 98 개로 맞춰놓고 하루(3 분)이 넘어 |

| | | |
|---|---|---|
| | | 갈 때를 기다린다. |
| 3_001_001 | 하루(3분)가 지나면 상품 파일의 모든 상품은 100개로 초기화된다. | 1_000_006 의 상태에서 판매를 하고 하루(3 분)이 넘어갈 때를 기다린다. |
| 3_002_000 | 재고서버에서 당일 재고 정보를 받아와 동기화를 하고 캐셔 화면에 재고 정보를 출력하고 영수증 프린터기를 이용해 재고 정보를 출력한다. | 하루(3 분)이 지나기 전에 과자(001)의 재고를 98 개로 맞춰놓고 전원을 껐다 키고 재고 확인을 한다. |
| 3_002_001 | 재고서버에서 당일 재고 정보를 받아와 동기화를 하고 캐셔 화면에 재고 정보를 출력하고 영수증 프린터기를 이용해 재고 정보를 출력한다. | 하루(3 분)이 지나기 전에 1_000_006 의 상태에서 판매를 하고 전원을 껐다 키고 재고 확인을 한다. |
| 3_003_000 | 판매 중이거나 환불 중에는 재고 확인을 할 수 없다. | 판매 중일 때 재고 확인이 되는지 확인한다. |
| 3_003_001 | 판매 중이거나 환불 중에는 재고 확인을 할 수 없다. | 환불 중일 때 재고 확인이 되는지 확인한다. |
| 3_004_000 | 화면 출력 | 화면 출력 |
| 4_000_000 | 프로그램 종료 후 재실행 시 기존에 진행하던 시각이 이어지도록 한다. | 프로그램 종료하기 전에 시간을 체크하고 재실행 시 기존 시각이 이어지는지 체크한다. |
| 4_001_000 | 현실 시간 3분이 지나면 하루가 넘어가며 (settle_YYYYMMDD.txt) 파일을 생성한다. | 현실 시간 3 분이 지나갔을 때 하루가 넘어가며 settle_YYYYMMDD.txt 가 생성되는지 확인한다. (기본화면) |
| 4_001_001 | 현실 시간 3분이 지나면 하루가 넘어가며 (settle_YYYYMMDD.txt) 파일을 생성한다. | 현실 시간 6 분이 지나가고 settle_YYYYMMDD.txt 가 생성되는지 확인한다. (기본화면) |
| 4_001_002 | 현실 시간 3분이 지나면 하루가 넘어가며 (settle_YYYYMMDD.txt) 파일을 생성한다. | 현실 시간 9 분이 지나가고 settle_YYYYMMDD.txt 가 생성되는지 확인한다. (기본화면) |
| 4_002_000 | 하루 동안 판매했던 내역을 재고서버에서 받아와 영수증 프린터기 | 하루가 지나가기 전에 1_003_000을 진행하고 |

| | | |
|---|---|---|
| | 를 이용해 출력한다. | 4_001_000을 진행한다 |
| 4_003_000 | 정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다. | 4_003_000 의 상태에서 판매 모드를 완료한다 |
| 4_003_001 | 정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다. | 4_003_000 의 상태에서 판매 모드를 완료한다 |
| 4_003_002 | 정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다. | 프로그램을 환불 화면으로 두고 3분을 기다린다. |
| 4_003_003 | 정산은 POST가 기본 화면인 경우에서만 진행이 되며, 판매 또는 환불 중에는 완료가 된 후에 정산이 진행된다. | 4_003_001의 상태에서 환불 모드를 완료한다 |
| 4_004_000 | 화면 출력 | 화면 출력 |
| 4_004_001 | 화면 출력 | 화면 출력 |

### 7.5 Feature pass/fail criteria

<Table 1 System Test Design Identification> 참조

위의 7.4의 Valid value 경우를 도출해 냈을 때, Pass이며 이외의 다른 값이 나오는 경우 Fail로 판단한다.

## 8 System test case specification

### 8.1 Test case specification identifier

Team5_N_nnn_mmm

### 8.2 Test items

<Table 3 Test Case Identification>

| Test Case Identification | Input Specification | Output Specification |
|---|---|---|
| Team5_0_000_000 | 판매 모드 진행 중 전원을 | 전원이 종료된다 |

| | | |
|---|---|---|
| | 종료한다 | |
| Team5_0_000_001 | 환불 모드 진행 중 전원을 종료한다 | 전원이 종료된다 |
| Team5_0_000_002 | 재고 확인 모드 진행 중 전원을 종료한다 | 전원이 종료된다 |
| Team5_1_000_000 | 바코드 001 입력 | 과자(001)가 추가 |
| Team5_1_000_001 | 바코드 010 입력 | 아이스크림(010)가 추가 |
| Team5_1_000_002 | 바코드 011 입력 | 과일(110)이 추가 |
| Team5_1_000_003 | 바코드 100 입력 | 물(100)이 추가 |
| Team5_1_000_004 | 바코드 101 입력 | 라면(101)이 추가 |
| Team5_1_000_005 | 바코드 110 입력 | 음료수(110)이 추가 |
| Team5_1_000_006 | 바코드 111 입력 | 커피(111)가 추가 |
| Team5_1_000_007 | 바코드 000 입력 | 예외 처리 |
| Team5_1_000_008 | 바코드 001 입력 후 001 입력 | 과자(001)가 2개 추가된다. |
| Team5_1_001_000 | 1_000_000상태에서 과자를 3개 추가 | 과자(001)가 5개가 된다. |
| Team5_1_001_001 | 1_001_000상태에서 과자를 4개 감소 | 과자(001)가 1개가 된다. |
| Team5_1_001_002 | 1_001_001상태에서 과자를 1개 감소시켜 제거 | 과자(001)이 목록에서 제거된다 |
| Team5_1_001_003 | 1_001_002 상태에서 과자를 1개 감소 | 예외 처리 |
| Team5_1_002_000 | 하루가 지나기 전에 001 상품을 102개 판매하려고 한다. | "개수가 부족합니다" |
| Team5_1_003_000 | 1_000_000상태에서 판매 목록의 합계 < 고객에게 받은 돈 | 결제가 진행된다. |
| Team5_1_003_001 | 1_000_000상태에서 판매 목록의 합계 > 고객에게 받은 돈 | 결제가 진행되지 않는다. |
| Team5_1_004_000 | 1_003_000 | 캐셔화면과 고객화면에 판매 정보가 업데이트된다. |
| Team5_1_005_000 | 1_003_000 | sale_YYYYMDDhhmm.txt파일이 생성된다. |
| Team5_1_006_000 | 1_003_000 | YYYYMMDD_product.txt의 상품의 개수가 업데이트된다. |

| | | |
|---|---|---|
| Team5_1_007_000 | 1_003_000 | YYYYMMDD_sale_management.txt 에 현재 판매 정보가 추가된다. |
| Team5_1_008_000 | 1_003_000 | 캐셔 화면(터치 스크린)<br>Display example<br>총 판매액 : 20,000<br>현금 : 30,000<br>거스름돈 : 10,000<br>상품 : 수량 : 5 / 단가 : 500 / 판매 금액 : 2,500<br>...<br>상품 : 수량 : 4 / 단가 : 1,000 / 판매 금액 : 4,000 |
| Team5_1_009_000 | 1_003_000 | 고객 화면<br>Display example<br>총 판매액 : 20,000<br>현금 : 30,000<br>거스름돈 : 10,000 |
| Team5_1_010_000 | 1_003_000 | 영수증 양식<br>영수증 번호 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>날짜 : YYYY.MM.DD (년.월.일)<br>판매 상품<br>상품, 단가, 수량, 판매 금액<br>...<br>상품, 단가, 수량, 판매 금액<br>총 판매액 : x |
| Team5_2_000_000 | 1_003_000에서 발급했던 영수증 바코드를 입력하고 환불 취소를 입력한다. | 환불이 되지 않는다 |
| Team5_2_000_001 | 2_000_000에 이어 1_003_000에서 발급했던 영수증 바코드를 입력하고 환불을 완료한다. | 환불이 된다 |
| Team5_2_000_002 | 2_000_001에 이어 1_003_000에서 발급했던 영수증 바코드를 입력한다. | "환불 불가능" |
| Team5_2_000_003 | 1_000_006에 이어 판매를 종료한 후 발급했던 영수증 바코드를 입력하고 환불을 완료한다. | 환불이 된다 |
| Team5_2_000_004 | 2_000_003에 이어 1_000_006에서 발급했던 영수증 바코드를 입력한다. | "환불 불가능" |
| Team5_2_001_000 | 발급하지 않았던 임의의 영수증 바코드를 입력한다 | "환불 불가능" |
| Team5_2_002_000 | 2_000_001 | refund_YYYYMMDDhhmm.txt를 생성 |
| Team5_2_002_001 | 2_000_003 | refund_YYYYMMDDhhmm.txt를 생성 |
| Team5_2_003_000 | 2_000_001 | YYYYMMDD_product.txt의 상품의 개수를 업데이트한다. |
| Team5_2_003_001 | 2_000_003 | YYYYMMDD_product.txt의 상품의 개수를 업데이트한다. |

| | | |
|---|---|---|
| Team5_2_004_000 | 2_000_001 | 캐셔 화면(터치 스크린)<br>Display example<br>판매 날짜 2017년 09월 07일 05시 20분<br>환불 금액 20,000<br>상품 수량 : 5 / 단가 : 500 / 판매 금액 : 2,500<br>...<br>상품 수량 : 4 / 단가 : 1,000 / 판매 금액 : 4,000 |
| Team5_2_005_000 | 2_000_001 | 고객 화면<br>Display example<br>판매 날짜 2017년 09월 07일 05시 20분<br>환불 금액 20,000 |
| Team5_2_006_000 | 2_000_001 | 영수증 양식<br>영수증 번호 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>날짜 : YYYY.MM.DD (년.월.일)<br>환불 상품<br>상품, 단가, 수량, 판매 금액<br>...<br>상품, 단가, 수량, 판매 금액<br>환불 금액 : x |
| Team5_3_000_000 | 과자(001)의 재고를 98개로 맞춰놓고 하루(3분)이 넘어갈 때를 기다린다. | 상품 파일, 판매 과리 파일이 다음 날짜로 생성된다. |
| Team5_3_000_001 | 1_000_006의 상태에서 판매를 하고 하루(3분)이 넘어갈 때를 기다린다. | 상품 파일, 판매 관리 파일이 다음 날짜로 생성된다. |
| Team5_3_001_000 | 과자(001)의 재고를 98개로 맞춰놓고 하루(3분)이 넘어갈 때를 기다린다. | 상품파일의 모든 상품은 100개로 초기화한다. |
| Team5_3_001_001 | 1_000_006의 상태에서 판매를 하고 하루(3분)이 넘어갈 때를 기다린다. | 상품파일의 모든 상품은 100개로 초기화한다. |
| Team5_3_002_000 | 하루(3분)이 지나기 전에 과자(001)의 재고를 98개로 맞춰놓고 전원을 껐다 키고 재고 확인을 한다. | 캐셔 화면에 재고 정보가 출력되고 Stock_YYYYMMDD.txt파일을 이용해 재고 정보를 출력한다. 당일 재고 정보이므로 과자의 재고 98개여야 한다. |
| Team5_3_002_001 | 하루(3분)이 지나기 전에 1_000_006의 상태에서 판매를 하고 전원을 껐다 키고 재고 확인을 한다. | 캐셔 화면에 재고 정보가 출력되고 Stock_YYYYMMDD.txt파일을 이용해 재고 정보를 출력한다. 당일 재고 정보이므로 상품의 재고는 99개여야 한다. |
| Team5_3_003_000 | 판매 중일 때 재고 확인이 되는지 확인한다. | 불가능해야 한다. |
| Team5_3_003_001 | 환불 중일 때 재고 확인이 되는지 확인한다 | 불가능해야 한다 |

| | | |
|---|---|---|
| Team5_3_004_000 | 3_001_000 | 재고 확인 양식<br>날짜 : YYYY.MM.DD.hh.mm (년.월.일.시.분)<br>상품, 단가, 재고<br>…<br>상품, 단가, 재고 |
| Team5_4_000_000 | 프로그램 종료하기 전에 시간을 체크하고 재실행 시 기존 시각이 이어지는지 체크한다. | 기존 시각이 이어져야 한다. |
| Team5_4_001_000 | 현실 시간 3분이 지나갔을 때 하루가 넘어가며 settle_YYYYMMDD.txt가 생성되는지 확인한다.(기본화면) | 하루가 넘어가야 하며 settle_YYYYMMDD.txt가 생성되는지 확인한다. |
| Team5_4_001_001 | 현실 시간 6분이 지나갔을 때 하루가 넘어가며 settle_YYYYMMDD.txt가 생성되는지 확인한다.(기본화면) | 하루가 넘어가야 하며 settle_YYYYMMDD.txt가 생성되는지 확인한다. |
| Team5_4_001_002 | 현실 시간 9분이 지나갔을 때 하루가 넘어가며 settle_YYYYMMDD.txt가 생성되는지 확인한다.(기본화면) | 하루가 넘어가야 하며 settle_YYYYMMDD.txt가 생성되는지 확인한다. |
| Team5_4_002_000 | 하루가 지나가기 전에 1_003_000을 진행하고 4_001_000을 진행한다. | settle_YYYYMMDD.txt가 생성되며 1_003_000의 정보가 있어야 한다. |
| Team5_4_003_000 | 프로그램을 판매 화면으로 두고 3분을 기다린다. | 정산이 진행되지 않는다. |
| Team5_4_003_001 | 4_003_000의 상태에서 판매 모드를 완료한다 | 정산이 진행된다. |
| Team5_4_003_002 | 프로그램을 환불 화면으로 두고 3분을 기다린다. | 정산이 진행되지 않는다 |
| Team5_4_003_003 | 4_003_001의 상태에서 환불 모드를 완료한다 | 정산이 진행된다 |
| Team5_4_004_000 | 4_002_000 | 정산 영수증 양식<br>날짜 : YYYY.MM.DD (년.월.일)<br>판매 상품<br>상품, 단가, 수량, 금액<br>…<br>상품, 단가, 수량, 금액<br>판매 금액 :<br>환불 상품<br>상품, 단가, 수량, 금액<br>…<br>상품, 단가, 수량, 금액<br>환불 금액 : |
| Team5_4_004_001 | 4_003_003 | "" |

8.3 Input specifications

<Table 3 Test Case Identification> 참조

8.4 Output specifications

<Table 3 Test Case Identification> 참조

9 Testing tasks

<Table4 Testing Tasks & Schedule>

| Task | Predecessor task | Special skills | Effort | Finish date |
|---|---|---|---|---|
| STP 작성 | PTS 구현 완료 | | | |
| Test Case Specification | Task1 | PTS에 대한 이해 | | |
| ST를 위한 인력 배치 | Task2 | PTS에 대한 지식 | | |
| ST를 위한 환경 구축 | Task3 | | | |
| ST 수행 | Task4 | | | |

10 Environmental needs

POS System의 System Test를 위한 환경적 요구사항은 다음과 같다.

(1) Hardware & Platform

GCC compiler / linker

(2) CTIP(Continuous Testing & Integrated Platform) Environment

Cygwin

11 System test deliverables

12 Schedules

<Table4 Testing Tasks & Schedule> 참조