

System Testing Plan for Public Transportation System

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

Project Team

Team 3

Latest update on:

2014-12-03

Team Information

201311269 김제현

201311275 박상희

201311276 박형민

201311287 엄현식

Table of Contents

| | | |
|-----------|---|----------|
| 1 | Introduction | 3 |
| 1.1 | Objectives | 3 |
| 1.2 | Background | 3 |
| 1.3 | Scope | 3 |
| 1.4 | Project plan | 3 |
| 1.5 | Configuration management plan | 3 |
| 1.6 | References | 4 |
| 2 | Test items | 4 |
| 2.1 | Software requirements specification | 4 |
| 3 | Approach | 4 |
| 4 | Item pass/fail criteria | 4 |
| 5 | System test design specification | 4 |
| 5.1 | Test design specification identifier | 4 |
| 5.2 | Features to be tested | 4 |
| 5.3 | Approach refinements | 6 |
| 6 | System test case specification | 6 |
| 6.1 | Test case specification identifier | 6 |
| 6.2 | Test items | 6 |
| 6.3 | Input specifications | 6 |
| 6.4 | Output specifications | 6 |
| 7 | Testing tasks | 6 |
| 8 | Environmental needs | 7 |
| 9 | System test deliverables | 7 |
| 10 | Schedules | 7 |

1 Introduction

1.1 Objectives

본 문서는 2014년 2학기 소프트웨어 공학 개론 수업에서 T3이 개발한 PTS(Public Transportation System), RS(Recharger System), FCS(Fee Calculation System)의 System Testing Plan (STP)을 위한 문서이다.

1.2 Background

Public Transportation System(이하 PTS)은 대중교통 카드 시스템으로, 사용자가 카드 태그를 통해 요금 결제, 환승 및 카드 정보를 출력하는 시스템이다.

Recharger System(이하 GS)은 충전 시스템으로, 사용자가 카드를 태그하고 돈을 넣어 카드를 충전 하는 시스템이다.

Fee Calculation System(이하 FCS)은 정산 시스템으로, 3분마다 수익을 계산하고 수익을 출력하고 전송하는 시스템이다.

PTS와 RS, FCS의 System Testing (ST)을 위하여 PTS의 Software Requirement Specification (SRS)에 대한 이해가 필요하다.

1.3 Scope

이 계획 문서는 PTS, GS, FCS의 System Test를 수행하기 위한 모든 것을 포함한다. PTS, GS, FCS의 System Test를 수행하기 위한 자원과 절차, test approach와 technique과 필요로 하는 환경 및 도구 등을 정의한다. 구현된 System이 요구사항을 만족하는 지 test 한다.

1.4 Project plan

완성된 PTS, RS, FCS의 Software (SW) 를 ST하기위한 본 계획서를 작성한다.

본 계획서에 작성된 내용에 기반하여 ST를 수행한다.

ST에 대한 결과를 개발팀에 통보한다.

1.5 Configuration management plan

SASD를 이용하여 SW를 구현한 후, System Testing (ST)/Unit Testing (UT)를 수행한다. ST/UT의 결과를 대상 SW에 반영한다. SRS 수정 후 반영된 SW를 대상으로 2차ST/UT

를 수행한다.

1.6 References

DS-2014SE-PTS-SRS-1.0

T3-2014-PTS-SRA-1.3

2 Test items

2.1 Software requirements specification

<Table 1 테스트할 목록들> 참조

3 Approach

요구사항에 정의되어 있는 기능들을 확인하기 위한 시나리오를 작성한다. 작성한 시나리오에 기반하여 Testing design 및 Test cases를 작성하여 Testing을 수행한다.

4 Item pass/fail criteria

5 System test design specification

5.1 Test design specification identifier

PTS.STC.Number

5.2 Features to be tested

<Table 1 테스트할 목록들>

| Test ID | Description |
|---------|---|
| 1. | 버스 1대와 지하철 5개역 (건대입구, 왕십리(동대문 역사 문화공원), 합정, 신림, 강남)이 있는지 확인 한다. |
| 2. | PTS가 총 3가지(버스용 단말기, 지하철용 단말기, 정산 시스템)로 구성 되었는지 여부 |
| 3. | 탑승 태그와 하차 태그가 가능한지의 여부 |
| 4. | 1개역 이하를 이동하면 기본료(1050원)가 부과되며 두 개역을 이동하면 추가 요금(200원)이 부과되는지 여부 |
| 5. | 하차 태그 했을 경우에만 bus와 지하철 간의 정해진 시간(15초)내에 환승이 가능한지 여부. |
| 6. | 지하철->버스 단위 시간 당 버스의 환승 요금이 추가로 부과(30초당 100원/최대 |

| | |
|-----|---|
| | 700원) 되는지에 대한 여부 |
| 7. | 버스->지하철 역 당 지하철의 환승 요금(역당 300원/최대 600원)이 추가로 부과되는지에 대한 여부 |
| 8. | 정산은 하루(3분)에 한번만 이뤄지는 지에 대한 여부. |
| 9. | 버스와 지하철에 각각 수익을 배분하는지에 대한 여부 |
| 10. | 승차 시 카드를 태그하며 하차 시 카드를 태그 할 수도 있고 안 할 수도 있는지에 대한 여부 |
| 11. | 운행이 시작하기 전까지 정산이 반드시 이루어 지는지에 대한 여부 |
| 12. | 정산 후 사용자 카드 정보를 제외한 모든 정보가 초기화 되는지에 대한 여부 |
| 13. | 교통카드 태그 행위 시 해당 파일에 제대로 입력이 되었는지에 대한 여부 |
| 14. | 잔액이 모자를 경우 태우지 않는지에 대한 여부(초기 탑승 시 : 기본요금인 1050원 / 환승 시 : [버스: 환승 최고 부과금액인 700원/지하철 환승 최고 부과금액인 600원] / *미정산 요금이 있을 시 : 기본요금(1050원)+ 미 정산 요금(200원or600원or700원)) |
| 15. | 하루(3분) 종료 시 탑승되어 있는 승객은 미정산으로 처리되는지에 대한 여부 |
| 16. | 정산 시 소수점 이하는 반올림하는지에 대한 여부 |
| 17. | PTS의 사용자 인터페이스에서 교통카드 정보를 입력 받고 사용금액, 잔액/시간, 일별 요금 목록, 거래 후 교통 카드 정보를 출력하는지에 대한 여부 |
| 18. | 하드웨어 인터페이스는 고려하지 않는다. |
| 19. | 카드 인식은 태그(카드ID를 입력하는 행위)를 통해 수행되는지에 대한 여부 |
| 20. | 지하철에서 버스 환승 후 하차 시 단말기를 태그 하지 않았을 경우 환승 최고 요금인 700원, 지하철 하차 후 미정산 금액은 200원, 버스에서 지하철 환승 후 미정산 금액은 지하철 환승 최대 요금인 600원을 부과하는지에 대한 여부 |
| 21. | 계산된 부과 요금과 기타 정보를 사용자 카드와 단말기 파일에 기록하는지에 대한 여부 |
| 22. | 각 역마다 하나의 지하철 단말기가 독립적으로 작동하는지에 대한 여부 |
| 23. | Data 구분은 쉼표(.)를 사용하는지에 대한 여부 |
| 24. | 각 단말기는 탑승 단말기 정보 (단말기 고유 ID + 하루 동안 태그 된 Count)를 가지고 있는지에 대한 여부 |
| 25. | 사용자가 단말기에 카드를 태그 할 경우 단말기는 환승 여부를 확인 후 환승이 아닐 경우 해당 탑승 단말기 정보를 교통카드와 단말기 누적 결제 기록에 기록되는지에 대한 여부 |
| 26. | 사용자가 단말기에 카드를 태그 할 경우 단말기는 환승 여부를 확인 후 환승일 경우 카드에 저장되어 있는 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록하는지에 대한 여부 |
| 27. | 정산 시스템의 사용자 인터페이스에서 버스과 지하철 단말기의 일별 요금 목록의 입력을 받고 정산 결과(데이터)를 출력 하는지에 대한 여부 |
| 28. | 각각의 일별 요금 목록을 버스과 지하철 단말기로부터 입력 받는지에 대한 여부 |

| | |
|-----|-------------------------------------|
| 29. | 정해진 정산 방법을 따르는지에 대한 여부 |
| 30. | 정산한 금액을 모니터에 출력하는 지에 대한 여부 |
| 31. | 정산한 금액을 각 회사(버스, 지하철)에 전송하는지에 대한 여부 |
| 32. | 정산 완료 신호를 전송하는지에 대한 여부 |

5.3 Approach refinements

PTS는 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

6 System test case specification

6.1 Test case specification identifier

PTS.STC.Number.Number

6.2 Test items

6.3 Input specifications

<Table 4 Test Cases Identification> 참조

6.4 Output specifications

<Table 4 Test Cases Identification> 참조

7 Testing tasks

<Table 2 Testing Tasks & Schedule>

| Task | Predecessor tasks | Special skill | Effort |
|-------------------------------|--------------------------------------|---------------|--------|
| 1. STP 작성 | DS-2014SE-PTS-SRS-1.0 , PTS 구현 완료 | | 3 |
| 2. Test Case Specification 작성 | Task1 | PTS에 대한 지식 | 4 |
| 3. ST를 위한 인력 배치 | | | 2 |
| 4. ST를 위한 환경 구축 | Task 3 | | 3 |
| 5. ST 수행 | Task 4 | | 5 |
| 6. ST Report 작성 | Task 5 | | 3 |

8 Environmental needs

PTS를 실행하기 위한 PC – Cygwin & GCC Compiler 필요

9 System test deliverables

10 Schedules

<Table 2 Testing Tasks & Schedule> 참조

<Table 3 System Test Design Identification>

| Identifier | Feature | Vaild/Invaild |
|-------------|---------------|---|
| PTS.STC.000 | 기본 구성 | 버스 1대와 지하철 5개역 (건대입구, 왕십리(동대문 역사 문화공원), 합정, 신림, 강남)이 있는지 확인 한다. |
| PTS.STC.001 | 기본 구성 | PTS가 총 3가지(버스용 단말기, 지하철용 단말기, 정산 시스템)로 구성 되었는지 확인 |
| PTS.STC.002 | 기본 구성 | 하드웨어 인터페이스는 고려하지 않는다. |
| PTS.STC.003 | 기본 구성 | 카드 인식은 태그(카드ID를 입력하는 행위)를 통해 수행되는지에 대한 확인 |
| PTS.STC.004 | 기본 구성 | 각 역마다 하나의 지하철 단말기가 독립적으로 작동하는지에 대한 확인 |
| PTS.STC.100 | 태그 | 탑승 태그와 하차 태그가 가능한지의 확인 |
| PTS.STC.200 | 기본요금 | 1개역 이하를 이동하면 기본료(1050원)가 부과되며 두 개역을 이동하면 추가 요금(200원)이 부과되는지 확인 |
| PTS.STC.201 | 환승 조건 | 하차 태그 했을 경우에만 버스와 지하철 간의 정해진 시간(15초)내에 환승이 가능한지 확인. |
| PTS.STC.202 | 지하철->버스 환승 요금 | 지하철->버스 단위 시간 당 버스의 환승 요금이 추가로 부과(30초당 100원/최대 700원) 되는지에 대한 확인 |
| PTS.STC.203 | 버스->지하철 환승 요금 | 버스->지하철 역 당 지하철의 환승 요금(역당 300원/최대 600원)이 추가로 부과되는지에 대한 확인 |
| PTS.STC.204 | 잔액 부족 | 잔액이 모자를 경우 태우지 않는지에 대한 확인 |

| | | |
|-------------|--------------|--|
| | | (초기 탑승 시 : 기본요금인 1050원 / 환승 시 : [버스: 환승 최고 부과금액인 700원/지하철 환승 최고 부과금액인 600원] / *미정산 요금이 있을 시 : 기본요금(1050원)+ 미 정산 요금(200원 or 600원 or 700원)) |
| PTS.STC.205 | 환승 요금 결정 조건 | 지하철에서 버스 환승 후 하차 시 단말기를 태그하지 않았을 경우 환승 최고 요금인 700원, 지하철 하차 후 미정산 금액은 200원, 버스에서 지하철 환승 후 미정산 금액은 지하철 환승 최대 요금인 600원을 부과하는지에 대한 확인 |
| PTS.STC.300 | 정산 주기 | 정산은 하루(3분)에 한번만 이뤄지는 지에 대한 확인. |
| PTS.STC.301 | 수익 배분 | 버스와 지하철에 각각 수익을 배분하는지에 대한 확인 |
| PTS.STC.302 | 정산 여부 | 하루(3분) 종료 시 탑승되어 있는 승객은 미정산으로 처리되는지에 대한 확인 |
| PTS.STC.303 | 정산 제약 조건 | 정산 시 소수점 이하는 반올림하는지에 대한 확인 |
| PTS.STC.304 | 정산 시스템 인터페이스 | 정산 시스템의 사용자 인터페이스에서 버스과 지하철 단말기의 일별 요금 목록의 입력을 받고 정산 결과(데이터)를 출력 하는지에 대한 확인 |
| PTS.STC.305 | 정산 조건 | 정해진 정산 방법을 따르는지에 대한 확인 |
| PTS.STC.306 | 정산 결과 전송 | 정산한 금액을 각 회사(버스, 지하철)에 전송하는지에 대한 확인 |
| PTS.STC.307 | 정산 완료 신호 전송 | 정산 완료 신호를 전송하는지에 대한 확인 |
| PTS.STC.400 | 사용자의 행동 | 승차 시 카드를 태그하며 하차 시 카드를 태그할 수도 있고 안 할 수도 있는지에 대한 확인 |
| PTS.STC.500 | 운행 시작 조건 | 운행이 시작하기 전까지 정산이 반드시 이루어지는지에 대한 확인 |
| PTS.STC.600 | 단말기 정보 초기화 | 정산 후 사용자 카드 정보를 제외한 모든 정보가 초기화 되는지에 대한 확인 |
| PTS.STC.700 | 파일 저장 | 교통카드 태그 행위 시 해당 파일에 제대로 입력이 되었는지에 대한 확인 |
| PTS.STC.701 | 파일 저장 | 계산된 부과 요금과 기타 정보를 사용자 카드와 단말기 파일에 기록하는지에 대한 확인 |
| PTS.STC.702 | 파일 저장 | Data 구분은 쉼표(,)를 사용하는지에 대한 확인 |
| PTS.STC.703 | 파일 저장 | 각 단말기는 탑승 단말기 정보 (단말기 고유 ID + 하루 동안 태그 된 Count)를 가지고 있는지에 대한 확인 |

| | | |
|-------------|-----------|---|
| | | 대한 확인 |
| PTS.STC.704 | 파일 저장 | 사용자가 단말기에 카드를 태그 할 경우 단말기는 환승 확인을 확인 후 환승이 아닐 경우 해당 탑승 단말기 정보를 교통카드와 단말기 누적 결제 기록에 기록되는지에 대한 확인 |
| PTS.STC.705 | 파일 저장 | 사용자가 단말기에 카드를 태그 할 경우 단말기는 환승 확인을 확인 후 환승일 경우 카드에 저장되어 있는 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록하는지에 대한 확인 |
| PTS.STC.706 | 파일 입력 | 각각의 일별 요금 목록을 버스와 지하철 단말기로부터 입력 받는지에 대한 확인 |
| PTS.STC.800 | PTS 인터페이스 | PTS의 사용자 인터페이스에서 교통카드 정보를 입력 받고 사용금액, 잔액/시간, 일별 요금 목록, 거래 후 교통 카드 정보를 출력하는지에 대한 확인 |

<Table 4 Test Cases Identification>

| Test Case Identifier | Input Specification | Output Specification |
|----------------------|--|-----------------------|
| PTS.STC.000.000 | | |
| PTS.STC.001.000 | | |
| PTS.STC.100.000 | | |
| PTS.STC.200.000 | 지하철 일반 승/하차 시 지하철 역 간격이 1정거장 이내 | 0원 부과 |
| PTS.STC.200.001 | 지하철 일반 승/하차 시 지하철 역 간격이 2정거장 일 때 | 200원부과 |
| PTS.STC.201.000 | 하차 태그 했을 때 다음 승차 태그 시간과 15초이내고, 교통수단이 달라졌을 때 | 환승 상태 |
| PTS.STC.202.000 | 버스를 환승 요금으로 탔을 때 | 30초마다 하차요금이 100원씩 증가 |
| PTS.STC.203.000 | 지하철을 환승 요금으로 탔을 때 | 1정거장마다 하차요금이 300원씩 증가 |

| | | |
|-----------------|--|---|
| | 로 탔을 때 | |
| PTS.STC.204.000 | 전에 미정산 요금부과 없이 정상 승차 처리 시, 기존 카드에 1050 원 보다 적은 금액이 들어있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.001 | 전에 지하철 미정산 요금부과상태에서 정상 승차 처리시, 기존 카드에 1250원 보다 적은 금액이 들어있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.002 | 지하철에서 버스로 환승 할 때,기존 카드에 700원 보다 적은 금액이 들어 있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.003 | 버스에서 지하철로 환승 할 때 기존 카드에 600원 보다 적은 금액이 들어 있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.004 | 전에 지하철 환승 미정산 요금부과 상태에서 정상 승차처리시, 기존 카드에 1650원 보다 적은 금액이 들어있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.005 | 전에 버스 환승 미정산 요금부과 상태에서 정상 승차처리시, 기존 카드에 1750원 보다 적은 금액이 들어있다. | "잔액 부족"이란 문구를 출력하고 승차되지 않은 채 초기화면으로 돌아간다. |
| PTS.STC.204.006 | 카드잔액이 부과금액보다 많다. | 카드의 잔액에서 부과금액을 빼고 카드의 상태를 승차 상태로 바꿔준다. 현재 타고 있는 교통의 정보를 입력시켜준다. |
| PTS.STC.205.000 | 지하철승차 -> 지하철 하차 -> 버스승차 -> 승차f | 1750원 |
| PTS.STC.205.001 | 지하철승차 -> 승차 | 1250원 |
| PTS.STC.205.002 | 버스승차 -> 버스하차 | 1650원 |

| | | |
|-----------------|--|--|
| | -> 지하철승차 -> 승차 | |
| PTS.STC.300.000 | 실행한지 3분마다 | 정산 실행 |
| PTS.STC.300.001 | 실행한지 3분마다가 아니면 | 정산을 실행하지 않는다. |
| PTS.STC.301.000 | 하루(3분)종료 시, 버스와 지하철의 일별 요금 목록 | 버스와 지하철에 각각 수익을 배분하는지에 대한 확인 |
| PTS.STC.302.000 | 정산 완료 신호 | 탑승되어 있는 승객은 미 정산으로 처리 |
| PTS.STC.303.000 | 정산 완료 신호 | 정산 시 소수점 이하는 반올림한다. |
| PTS.STC.304.000 | 하루(3분)이 종료될 시, 버스와 지하철 단말기의 각 단말기 파일로부터 하루치 결제 정보 (일별 요금 목록)를 전달 받는다 | 정산 결과(데이터)를 출력 버스 : □□원 지하철 : ○○원 |
| PTS.STC.305.000 | 환승, 미정산과 관계 없이 그냥 버스만 탑승한 상황 | 정산 금액 버스 += 버스 기본 요금(1050원) |
| PTS.STC.305.001 | 환승, 미정산과 관계 없이 그냥 버스만 탑승한 상황으로 0~1개역을 이동한 상황 | 정산 금액 지하철 += 지하철 기본 요금(1050원) |
| PTS.STC.305.002 | 환승, 미정산과 관계 없이 그냥 버스만 탑승한 상황으로 2개역을 이동한 상황 | 정산 금액 지하철 += 지하철 기본 요금(1250원) |
| PTS.STC.305.003 | 미정산과 관계 없이 버스→지하철 환승한 상황으로 n개역을 이동한 상황 | 정산 금액 버스 += $(1050+300*n)*(1050+300*n)/((1050+300*n)+300*n)$ 원 정산 금액 지하철 += $300*n*(1050+300*n)/((1050+300*n)+300*n)$ 원 |
| PTS.STC.305.004 | 미정산과 관계 없이 지하철→버스 환승한 상황으로 탑승후 0~n 초일 때 하차태그한 상 | 정산 금액 버스 += $(1050+n/30*100)*$ $(n/30*100)/((1050+n/30*100)+n/30*100)$ 원 정산 금액 지하철 += |

| | | |
|------------------|--|---|
| | 항 | $(1050+n/30*100)* (1050+n/30*100)/$ $((1050+n/30*100)+n/30*100)$ 원 |
| PTS.STC.305.005 | 전에 정상처리로 지하철을 탄 상황에서 하차 태그를 찍지 않고 내려 미정산 요금이 있을 때, 버스를 탄 상황 | 정산 금액 버스 += 버스 기본 요금(1050원) 정산 금액 지하철 += 지하철 미정산 요금(200원) |
| PTS.STC.305.006 | 전에 정상처리로 지하철을 탄 상황에서 하차 태그를 찍지 않고 내려 미정산 요금이 있을 때, 지하철을 탄 상황 | 정산 금액 버스 += 지하철 기본 요금(1050원) 정산 금액 지하철 += 지하철 미정산 요금(200원) |
| PTS.STC.305.007 | 전에 환승처리로 지하철을 탄 상황에서 하차 태그를 찍지 않고 내려 미정산 요금이 있을 때, 버스를 탄 상황 | 정산 금액 버스 += 버스 기본 요금(1050원) 정산 금액 지하철 += 지하철 최대 미정산 요금 (600원) |
| PTS.STC.305.008 | 전에 환승처리로 버스를 탄 상황에서 하차태그를 찍지 않고 내려 미정산 요금이 있을 때, 버스를 탄 상황 | 정산 금액 버스 += 지하철 기본 요금(1050원) 정산 금액 지하철 += 버스 최대 미정산 요금(700원) |
| PTS.STC.305.009 | 미정산과 관계 없이 버스→지하철→버스 환승한 상황 | $**(총요금 = (버스1 + 지하철 + 버스2) + (지하철 + 버스2) + (버스2))$ $**(부과요금 = 버스1 + 지하철 + 버스2)$ 정산 금액 버스 += $(버스1 + 지하철 + 버스2) / (총요금) * (부과요금)$ 정산 금액 지하철 += $(지하철 + 버스2) / (총요금) * (부과요금)$ 정산 금액 버스 += $(버스2) / (총요금) * (부과요금)$ |
| PTS.STC.305.0010 | 미정산과 관계 없이 지하철→버스→지하철 환승한 상황 | $**(총요금 = (지하철1 + 버스 + 지하철2) + (버스 + 지하철2) + (지하철2))$ $**(부과요금 = 지하철1 + 버스 + 지하철2)$ 정산 금액 버스 += $(지하철1 + 버스 + 지하철2) / (총요금) * (부과요금)$ 정산 금액 지하철 += $(버스 + 지하철2) / (총요금) * (부과요금)$ 정산 금액 지하철 += $(지하철2) / (총요금) * (부과$ |

| | | |
|------------------|--|--|
| | | 요금) |
| PTS.STC.305.0011 | 미정산과 관계 없이 버스→지하철→버스→ 지하철 환승한 상황 | <p>**$(\text{총요금} = (\text{버스1} + \text{지하철1} + \text{버스2} + \text{지하철2}) + (\text{지하철1} + \text{버스2} + \text{지하철2}) + (\text{버스2} + \text{지하철2}) + (\text{지하철2}))$</p> <p>**$(\text{부과요금} = \text{버스1} + \text{지하철1} + \text{버스2} + \text{지하철2})$</p> <p>정산 금액 버스 += $(\text{버스1} + \text{지하철1} + \text{버스2} + \text{지하철2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 지하철 += $(\text{지하철1} + \text{버스2} + \text{지하철2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 버스 += $(\text{버스2} + \text{지하철2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 지하철 += $(\text{지하철2}) / (\text{총요금}) * (\text{부과요금})$</p> |
| PTS.STC.305.012 | 미정산과 관계 없이 지하철→버스→지하철 →버스 환승한 상황 | <p>**$(\text{총요금} = (\text{지하철1} + \text{버스1} + \text{지하철2} + \text{버스2}) + (\text{버스1} + \text{지하철2} + \text{버스2}) + (\text{지하철2} + \text{버스2}) + (\text{버스2}))$</p> <p>**$(\text{부과요금} = \text{지하철1} + \text{버스1} + \text{지하철2} + \text{버스2})$</p> <p>정산 금액 지하철 += $(\text{지하철1} + \text{버스1} + \text{지하철2} + \text{버스2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 버스 += $(\text{버스1} + \text{지하철2} + \text{버스2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 지하철 += $(\text{지하철2} + \text{버스2}) / (\text{총요금}) * (\text{부과요금})$</p> <p>정산 금액 버스 += $(\text{버스2}) / (\text{총요금}) * (\text{부과요금})$</p> |
| PTS.STC.306.000 | 정산 완료 시 | 버스회사 및 지하철 회사로 정산한 금액을 각각 전송 |
| PTS.STC.307_000 | 정산 완료 시 | 버스회사 및 지하철 회사로 정산 완료 신호를 각각 전송 |
| PTS.STC.400.000 | CardID | 승차 시 항상 카드를 태그하며 하차 시 카드를 태그 할 수도 있고 안 할 수도 있다. |
| PTS.STC.500.000 | 정산 완료 신호 | PTS 작동 |
| PTS.STC.500.001 | 정산 완료 신호가 없다. | PTS 작동되지 않는다. |
| PTS.STC.600.000 | 정산 완료 신호 | 사용자 카드 정보를 제외한 모든 정보가 초기화 |
| PTS.STC.700.000 | 파일 저장 | 교통카드 태그 행위 시 해당 파일에 제대로 입력이 되었는지에 대한 확인 |

| | | |
|-----------------|----------------------------------|--|
| PTS.STC.701.000 | Tag | Updated 사용자 카드 |
| PTS.STC.701.001 | Tag | Updated 단말기 파일 |
| PTS.STC.704.000 | 버스에서 정상 탑승 태그시 환승이 아닐경우, | 버스에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.704.001 | 건대입구역에서 정상 탑승 태그시 환승이 아닐경우, | 건대입구역에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.704.002 | 강남역에서 정상 탑승 태그시 환승이 아닐경우, | 강남역에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.704.003 | 신림역에서 정상 탑승 태그시 환승이 아닐경우, | 신림역에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.704.004 | 합정역에서 정상 탑승 태그시 환승이 아닐경우, | 합정역에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.704.005 | 동대문역사문화공원역에서 정상 탑승 태그시 환승이 아닐경우, | 동대문역사문화공원역에서 탑승했다는 정보를 교통카드와 단말기 누적 결재 기록에 기록 |
| PTS.STC.705.000 | 버스에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.705.001 | 건대입구역에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.705.002 | 강남역에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.705.003 | 신림역에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.705.004 | 합정역에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.705.005 | 동대문역사문화공원역에서 정상 탑승 태그시 환승인 경우, | 카드에 저장되어 있는 이전 탑승 단말기 정보를 읽어와 교통카드와 단말기 누적 결제 기록에 기록 |
| PTS.STC.706.000 | 각 파일로부터 입력을 받는다. | 만약 매번 각 파일로부터 입력받은 값으로부터 올바르게 정산이 되었다면, 그것을 통해, 각각의 일별 요금 목록을 버스와 지하철 단말기로부터 입력 받는지를 증명할 수 있다. |
| PTS.STC.800.000 | Card Information | price, cash, Tag time, Daily Charge List, Card |

| | | |
|--|--|--------------------------|
| | | information after charge |
|--|--|--------------------------|