

# System Testing Plan for Public Transportation System

- Test Plan
- Test Design Specification
- Test Cases Specification

## Project Team

AT2 Team

Latest update on:

**2014-12-04**

---

## Team Information

201011337 변영석

201011349 이규진

201011365 정재혁

## 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	4
2.1	Software requirements specification	4
2.1.1	Product functions	4
2.1.2	Constraints	5
2.2	Software requirements analysis	6
2.2.1	System Context Diagram	6
2.2.1.1	Public Transportation System	6
2.2.1.2	Recharger System	6
2.2.1.3	Fee Calculation System	7
2.2.2	Event List	7
2.2.2.1	Public Transportaion System	7
2.2.2.2	Recharger System	8
2.2.2.3	Fee Calculation System	8
3	Features to be tested	8
3.1	Public Transportation System	8
3.2	Recharger System	9
3.3	Fee Calculation System	9

4	Features not to be tested	10
4.1	Public Transportation	10
4.2	Recharger System	11
4.3	Fee Calculation	11
5	Approach	11
6	Item pass/fail criteria	11
7	System test design specification	11
7.1	Test design specification identifier	12
7.2	Features to be tested	12
7.2.1	Bus System	12
7.2.2	Subway System	12
7.2.3	Adjust System	12
7.3	Approach refinements	12
7.4	Test identification	12
	<i>DWS.STC.Number.Number</i>	12
7.5	Feature pass/fail criteria	12
8	System test case specification	12
8.1	Test case specification identifier	12
8.2	Test items	13
8.3	Input specifications	13
8.4	Output specifications	13
9	Testing tasks	13
10	Environmental needs	13
11	System test deliverables	13
12	Schedules	13

## 1 Introduction

### 1.1 Objectives

Public Transportation System (이하 PTS)의 Unit test를 위해 필요한 활동과 기준에 대한 정의, 환경적인 요구사항, System Testing 도구들에 관해 세부적으로 명시한다.

### 1.2 Background

PTS는 대중 교통 시스템으로 사용자가 카드를 이용하여 센스에 태그 했을 때 입력 값에 따라서 알맞은 요금을 계산하여 그 결과를 다시 그 카드에 출력하는 시스템이다. 또한 그 요금을 매일 1회씩 정산하여 버스와 지하철 회사에 결과를 보내준다.

### 1.3 Scope

이 계획 문서는 PTS의 System Testing (이하 ST)을 수행하기 위한 모든 것을 포함한다.

### 1.4 Project plan

완성된 PTS의 Software (SW)를 ST하기 위한 본 계획서(STP)를 작성한다.

작성된 STP에 기반하여 ST를수행한다.

ST를 결과를 기반으로 프로그램을 수정, 보완하여 프로젝트를 완성한다.

### 1.5 Configuration management plan

### 1.6 References

[2014SE\_A][T2]SRA\_2\_1

[2014SE\_A][T2]SDS

DS-2014SE-PTS-SRS-1.0

## 2 Test items

### 2.1 Software requirements specification

#### 2.1.1 Product functions

버스용 단말기는 버스에 부착돼 탑승 태그와 하차 태그가 가능하다. 기본료를 지불하면 1 회 탑승이 가능하다.

지하철용 단말기는 역에 부착돼 탑승 태그와 하차 태그가 가능하다. 1 개역 이하를 이동하면 기본료가 부가되며 두 개역을 이동하면 추가 요금이 부가된다. 버스와 지하철 간에는 정해진 시간 내에 환승이 가능하다. 지하철에서 버스로 환승한 경우는 단위 시간 당 버스의 환승 요금이 추가로 부가된다. 하차 시 단말기에 태그를 하지 않으면 환승은 적용되지 않는다.

정산은 하루에 한 번 이뤄진다. 버스와 지하철의 기록을 분석하고 버스 지하철에 각각 수익을 배분한다.

### 2.1.2 Constraints

버스 승차는 지역에 상관 없이 할 수 있다.

지하철은 2 호선 역 중 5 개만 고려한다. : 건대 입구, 동대문역사문화공원, 합정, 신림, 강남.

하루는 3 분으로 가정한다.

다음 날 운행이 시작하기 전까지는 정산이 반드시 이루어 져야 한다.

정산 후 모든 프로그램의 정보는 초기화된다. (사용자 카드 정보 제외)

환승이 가능한 시간은 15 초 이내다.

버스 환승 시 추가요금의 기준인 단위 시간은 30 초이다.

버스와 지하철의 기본료는 1050 원이다.

교통카드는 저장된 텍스트 파일로 가정하고, 교통카드 태그 행위를 해당 파일을 입력하는 것으로 가정한다.

잔액이 모자를 경우 태우지 않는다.

버스 환승 최고 부과금액인 700 원이 남아 있지 않으면 버스로 환승시키지 않는다.

지하철 환승 최고 부과금액인 600 원이 남아 있지 않으면 지하철로 환승시키지 않는다.

하루(3 분) 종료 시 탑승되어 있는 승객은 미정산으로 처리한다.

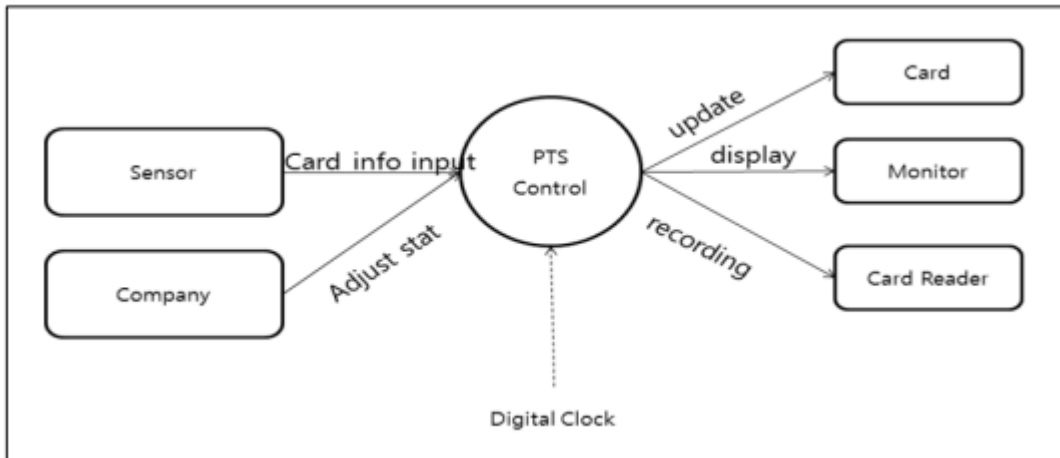
정산시 소수점 이하는 반올림한다.

승차 태그와 하차 태그의 구분은 사용자 입력 시 데이터로 넣어준다.

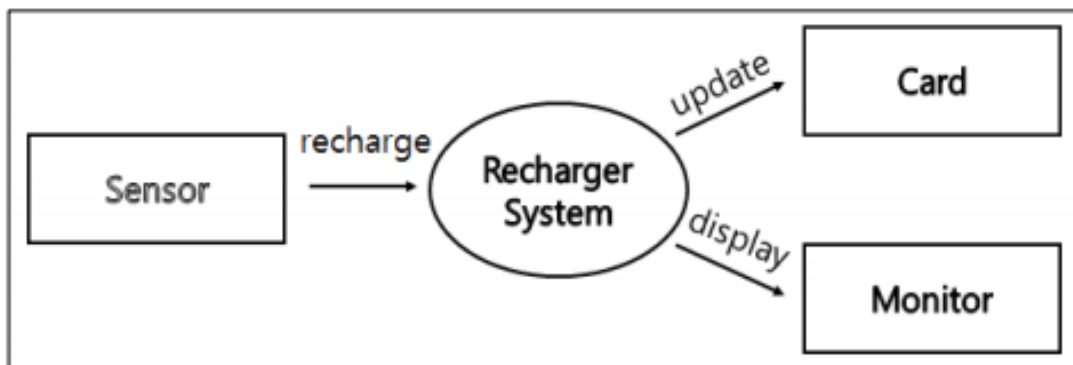
## 2.2 Software requirements analysis

### 2.2.1 System Context Diagram

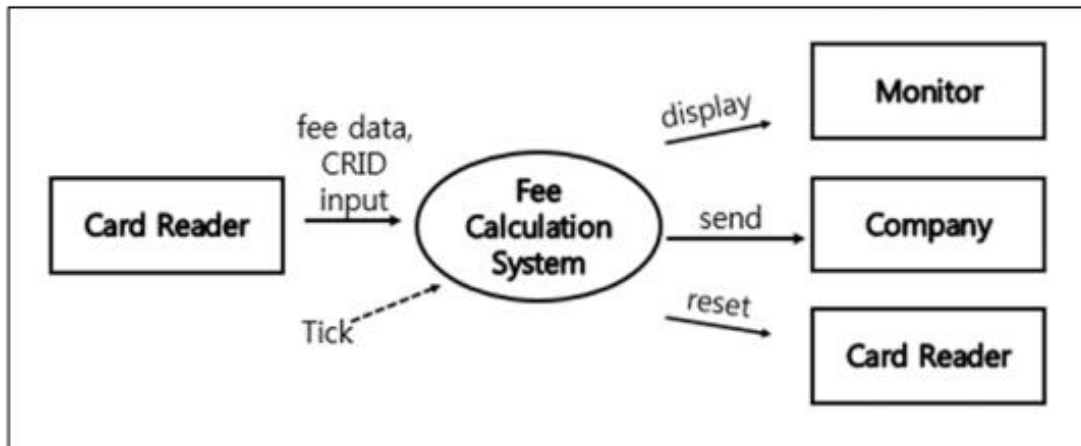
#### 2.2.1.1 Public Transportation System



#### 2.2.1.2 Recharger System



2.2.1.3 Fee Calculation System



2.2.2 Event List

2.2.2.1 Public Transportation System

Input / Output Event	Description
<b>Card info input</b>	Card의 information을 PTS 시스템으로 넘겨준다.
<b>adjust stat</b>	매 3분마다 정산이 진행되는데, Company로부터 정산이 완료되었는지에 대한 정보를 받아온다. 정산이 완료되지 않았을 경우 PTS 시스템은 동작되지 않는다.
<b>update</b>	PTS 시스템에서 처리된 정보를 카드에 update해준다.
<b>display</b>	PTS 시스템에서 처리된 정보를 Monitor에 Display해준다.
<b>recording</b>	각 역 단말기에 처리된 정보를 기록한다.

## 2.2.2.2 Recharger System

Input / Output Event	Description
<b>recharge</b>	충전 단말기 센서에 카드를 대서 감지된 후, 충전할 만큼의 현금을 투입할 때, 교통카드와 현금의 정보
<b>updated</b>	충전 시스템 안에서 계산된 결과를 가지고 갱신시킬 카드의 정보
<b>display</b>	단말기 시스템 안에서 계산된 결과를 가지고 Monitor화면에 출력할 정보

## 2.2.2.3 Fee Calculation System

Input / Output Event	Description
<b>fee data input</b>	카드 단말기로부터 저장된 요금 정보를 불러온다.
<b>display</b>	모니터에 정산된 요금을 출력한다.
<b>send</b>	회사에 정산된 요금을 보낸다.
<b>reset</b>	정산이 잘된 경우 카드 단말기의 요금 정보를 초기화 시킨다.

## 3 Features to be tested

- 1) Process in SRA : 각 프로세스가 가지고 있는 요구사항을 만족하는지를 Test한다.
- 2) Modules in SDS : 각 모듈이 가지고 있는 데이터 인터페이스를 Test 한다.

## 3.1 Public Transportation System

&lt;Table 1.1 테스트할 Process(DFD) 리스트&gt;

ID	Name	Description
<b>1.2</b>	Car Info load	CID를 indexing하여 해당 Card의 잔액, 탑승 단말기, 승/하차, 태그 시간, 환승상태 정보를 불러온다.



2.1.1.1	Catch Error Controller	정산 여부에 관한 정보를 받아온 뒤, 정산이 이루어지지 않았을 경우 경고메시지를 출력하도록 한다. 정산이 이루어졌을 경우 Card Info를 받아와서 정상적인 카드 Tag가 이루어졌는지 판별하고, 아닐 경우 경고메시지를 출력하도록 한다. 정상적일 경우 가격을 측정하도록 한다.
2.1.1.2	Error	정상적이지 않은 Tag나 정산이 이루어지지 않았을 경우 경고 메시지를 보낸다.
2.1.1.3	Fix price	Card Info를 받아온 뒤, 해당 조건에 맞는 가격을 측정한다.
2.1.2.1	Money Check Controller	측정된 가격을 받아온 뒤, Card Info의 잔액과 비교한 뒤 부족하면 Short, 충분하면 Calculation을 실행한다.
2.1.2.2	Short	가격이 부족할 경우 경고메시지를 출력한다.
2.1.2.3	Calculation	잔액이 충분할 경우 잔액-결제금액을 한 뒤, 승/하차 상태 및 환승 상태, 역 단말기 정보, 잔액을 갱신한 뒤 Updated data로 보내준다.
2.1.3	Payment Controller	Card 결제가 이루어진 후, 처리된 정보와 현재 시각을 받아와서 결과값을 출력하고, Card에 갱신, 역 단말기에 기록한다.
2.1.4	Result	Card 결제가 이루어지고 난 뒤의 결제 금액, 잔액, 현재 시각 (Display data)을 보내준다.
2.1.5	Card update	Card 결제가 이루어지고 난 뒤 바뀐 Card Info를 갱신한다.
2.1.6	Card Reader Record	Card 결제가 이루어지고 난 뒤, 결제 금액을 Card Reader에 기록한다.

### 3.2 Recharger System

<Table 2.1 테스트할 Process(DFD) 리스트>

ID	Name	Description
1.2	Card Info Loader	CID를 사용해서 Card Info를 불러와 Recharger Control 에 전달한다.
2.1.1	Recharger Controller	입력받은 Card Info, money Data를 종합하여 충전 계산을 한 뒤, 적절한 update와 display를 실행 해준다.
2.1.2	Update	충전된 정보로 교통카드를 갱신한다.
2.1.3	Display	교통카드에 충전된 정보를 Monitor에 보여준다.

### 3.3 Fee Calculation System

ID	Name	Description
1	Card Reader Info Loader	CRID(단말기 고유 ID)를 받아서 해당하는 단말기파일을 열어서 데이터들을 불러와서 데이터 형식에 맞게 각각 저장한다.

2.1	Fee Calculation Controller	CRID를 통해서 얻은 데이터와 Tick을 받아서 정산을 한 후 상태를 통해서 적절한 프로세스에 명령을 전달한다.
2.2	Display	Fee Calculation Controller가 정산한 결과를 출력할 때 호출하는 프로세스. 명령을 받으면 adjust_well==0인지 비교(정산이 잘 되었는지)를 하여 정산이 잘 되었으면 adjusted data (fee_bus, fee_metro, time_now)를 출력한다.
2.3	Send	Fee Calculation Controller가 정산한 결과를 지하철회사와 버스 회사에 보낼 때 호출하는 프로세스. Trigger 명령을 받으면 adjust_well==0인지 비교를 하여 정산이 잘 되었으면 버스 회사와 지하철 회사에 정산이 잘되었다는 값(c_well)과 정산 결과값 (fee_bus or fee_metro)을 보낸다.
2.4	Reset	Fee Calculation Controller가 정산한 결과를 초기화 시킬 때 호출하는 프로세스. Trigger 명령을 받으면 adjust_well==0인지 비교를 하여 정산이 잘 되었으면 모든 단말기 파일을 초기화 시킨다.

#### 4 Features not to be tested

Process in SRA : 외부 장치 드라이버, 단순 데이터 전달 프로세스 등은 Test에서 제외한다.

##### 4.1 Public Transportation

<Table 1.2 테스트하지 않을 Process(DFD) 리스트>

ID	Name	Description
1.1	Card Reading Interface	카드를 Tag했을 때 전달되는 신호를, 컴퓨터가 해석할 수 있는 값을 바꾸어 보내준다.
2.2	Display interface	결정된 금액 혹은 경고 메시지, 현재 시각에 관한 정보를 Display data를 통해 받아오고, 정리된 정보를 보내준다.
2.3	Card interface	결제 후, 갱신해야 할 Card data를 받아온 뒤, Card에 넘겨준다.
2.4	Card Reader Recording interface	결제 후, 결제 금액을 받아온 뒤, 각 Card Reader에 기록하기 위한 정보를 보내준다

## 4.2 Recharger System

&lt;Table 2.2 테스트하지 않을 Process(DFD) 리스트&gt;

ID	Name	Description
1.1	Card ID Receive Interface	Recharger Sensor로부터 받은 아날로그 신호를 디지털 신호로 변환한다
2.2	Update Interface	update card data를 받아서 Card 정보를 Update 시키는 정보를 보내준다
2.3	Display interface	display data를 받아서 Monitor에 출력할 display정보를 보내준다.
2.4	Money Sensor Interface	Money Sensor로부터 받은 아날로그 신호를 디지털 신호로 변환한다.

## 4.3 Fee Calculation

&lt;Table 3.2 테스트하지 않을 Process(DFD) 리스트&gt;

ID	Name	Description
3	Display Interface	adjusted data(fee_bus, fee_metro, time_now)를 받아서 정산 결과를 출력해주는 Display 장치에 보내준다.
4	Send Interface	adjusted data(fee_bus, fee_metro, c_well)를 받아서 정산 결과를 버스 회사와 지하철 회사에 보내주는 Send 장치에 보내준다.

## 5 Approach

Public Transportation System의 program source code 및 unit test를 위한 test code는 CTIP(Continuous Testing & Integration Platform) 환경에서 이루어지며, program source code/test code 의 변경 및 수정 사항은 지속적으로 통합되고 test 된다.

## 6 Item pass/fail criteria

Functional test pass/fail criteria : 각 모듈은 요구사항을 모두 만족하여야 한다

## 7 System test design specification

7.1 Test design specification identifier

*PTS.STC.Number*

7.2 Features to be tested

7.2.1 Bus System

승차, 하차 시 태그 입력에 따른 올바른 요금을 모니터 출력한다.

올바른 요금에 따른 승차/하차 정보를 카드에 입력한다.

매 3분마다 정산 과정과 초기화 과정을 진행한다.

7.2.2 Subway System

승차, 하차 시 태그 입력에 따른 올바른 요금을 출력한다.

올바른 요금에 따른 승차/하차 정보를 카드에 입력한다.

매 3분마다 정산 과정과 초기화 과정을 진행한다.

7.2.3 Adjust System

매 3분마다 카드 단말기로부터 저장된 요금정보를 불러와 모니터에 정산된 요금을 출력한다.

회사에 정산된 요금을 보낸다.

정산이 잘 된 경우 요금 내역을 초기화 시킨다.

7.3 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.

7.4 Test identification

*DWS.STC.Number.Number*

7.5 Feature pass/fail criteria

8 System test case specification

8.1 Test case specification identifier

<Table 8 Test Case Identification 참조>

## 8.2 Test items

<Table 9 Test Case Identification 참조>

## 8.3 Input specifications

<Table 8 Test Case Identification 참조>

## 8.4 Output specifications

<Table 8 Test Case Identification 참조>

## 9 Testing tasks

**Table 7 Testing Tasks & Schedule**

Task	Predecessor tasks	Special skills	Effort	Finish date
(1) STP 작성	SRS, SRA, UTP, UTR, PTS 구현 완료		2	
(2) Test case specification 작성	Task 1	PTS에 대한 지식	3	
(3) ST를 위한 인력 배치	Task 2		6	
(4) ST를 위한 환경 구축	Task 3		5	
(5) ST 수행	Task 4		5	
(6) ST report 작성	Task 5		2	

## 10 Environmental needs

PTS를 실행하기 위한 PC – Cigwin 환경 필요

## 11 System test deliverables

## 12 Schedules

<Table 7 Testing Tasks & Schedule> 참조

Table 8 Test Case Identification

Test Case	Input Specification	Output Specification
PTS.STC.000.000	프로그램 실행	-
PTS.STC.001.000	Bus : 기본요금 탑승	교통수단 : BUS / 승차 / 결제금액 : 1050
PTS.STC.001.001	Bus : 환승요금 탑승	교통수단 : BUS / 승차 / 결제금액 : 0
PTS.STC.001.002	Bus : 미정산요금 추가탑승 (지하철->버스미정산)	교통수단 : BUS / 승차 / 결제금액 : 700
PTS.STC.001.003	Bus : 미정산요금 추가 탑승 (지하철 미정산)	교통수단 : BUS / 승차 / 결제금액 : 200
PTS.STC.001.004	Bus : 미정산요금 추가탑승 (버스->지하철미정산)	교통수단 : BUS / 승차 / 결제금액 : 600
PTS.STC.001.005	Bus : 기본요금 하차	교통수단 : BUS / 하차 / 결제금액 : 0
PTS.STC.001.006	Bus : 시간비례요금 하차 (30초 경과)	교통수단 : BUS / 하차 / 결제금액 : 100
PTS.STC.001.007	Bus : 시간비례요금 하차 (1분 경과)	교통수단 : BUS / 하차 / 결제금액 : 200
PTS.STC.001.008	Bus : 시간비례요금 하차 (4분 경과)	교통수단 : BUS / 하차 / 결제금액 : 700
PTS.STC.002.000	Subway : 기본요금 탑승	교통수단 : SUBWAY / 승차 / 결제금액 : 1050
PTS.STC.002.001	Subway : 환승요금 탑승	교통수단 : SUBWAY / 승차 / 결제금액 : 0
PTS.STC.002.002	Subway : 미정산요금 추가탑승 (지하철->버스미정산)	교통수단 : SUBWAY / 승차 / 결제금액 : 700
PTS.STC.002.003	Subway : 미정산요금 추가 탑승 (지하철 미정산)	교통수단 : SUBWAY / 승차 / 결제금액 : 200
PTS.STC.002.004	Subway : 미정산요금 추가탑승 (버스->지하철미정산)	교통수단 : SUBWAY / 승차 / 결제금액 : 600
PTS.STC.002.005	Subway : 기본요금 하차	교통수단 : SUBWAY / 하차 / 결제금액 : 0
PTS.STC.002.006	Subway : 기본요금 하차 (2정거장 이동)	교통수단 : SUBWAY / 하차 / 결제금액 : 200
PTS.STC.002.007	Subway : 거리비례요금 하차 (1정거장 이동)	교통수단 : SUBWAY / 하차 / 결제금액 : 300
PTS.STC.002.008	Subway : 시간비례요금 하차 (2정거장 이동)	교통수단 : SUBWAY / 하차 / 결제금액 : 600
PTS.STC.003.000	기본정산 0 (Table 1.1 참조)	(정산) 버스 : 1050, 지하철 : 1050
PTS.STC.003.001	기본정산 1 (Table 1.1 참조)	(정산) 버스 : 1210, 지하철 : 440
PTS.STC.003.002	기본정산 2 (Table 1.1 참조)	(정산) 버스 : 92, 지하철 : 1058
PTS.STC.003.003	기본정산 3 (Table 1.1 참조)	(정산) 버스 : 1204, 지하철 : 344
PTS.STC.003.004	기본정산 4 (Table 1.1 참조)	(정산) 버스 : 1447, 지하철 : 901

Table 8.1 정산 Case

기본정산 0	버스 기본요금 승차 지하철 기본요금 승차
기본정산 1	버스 기본요금 승차 → 지하철 환승 후 2정거장 이동
기본정산 2	지하철 기본요금 승차 → 버스 환승 후 30초 경과
기본정산 3	버스 기본요금 승차 → 지하철 환승 후 1정거장 이동 → 버스 환승 후 1분 경과
기본정산 4	버스 기본요금 승차 → 지하철 환승 후 2정거장 이동 → 버스 환승 후 30초 경과 → 지하철 환승 후 2정거장 이동

Table 9 Test Design Identification

Identifier	Feature	Valid / Invalid value
PTS.STC.000	프로그램 실행	-
PTS.STC.001	Bus System 확인	버스 승차, 하차 command 입력 시 교통카드를 대체하는 txt 파일과 단말기 txt 파일에 제대로 저장되는지 확인
PTS.STC.002	Subway System 확인	지하철 승차, 하차 command 입력 시 교통카드를 대체하는 txt 파일과 단말기 txt 파일에 제대로 저장되는지 확인
PTS.STC.003	정산 System 확인	정산 시에 요금이 제대로 정산되는지 확인