

## 프로그램 매뉴얼

### 1.PTS

-간단한 실행법.

(1)cygwin 프롬프트 6개를 연 후 PTSTerm.exe를 모두 실행한다.

(2)6개에 1~6까지 순서대로 입력한다.(지하철역과 버스역 입력)

(3)탑승하고 싶은 터미널에서 1번을 입력한다.

(4)하차하고 싶은 터미널에서 2번을 입력한다.

-상세 설명

cygwin 프롬프트 창 6개를 켜 후 각 창마다 PTSTerm을 실행시켜준다. 6개를 실행 시킨 후, 각 프롬프트별로 지정된 지하철, 버스 역에 따라 1~6까지의 숫자를 입력한다. (이때, 숫자 입력시 그 텀을 최소화하여야 한다. 이유인 즉슨, 사실 각 단말기마다 독립적으로 동작하는 프로그램을 한번에 6개 띄우고 하는 것이기 때문에 시간적인 부분에서 오차가 생기는데, 그 부분을 최소화하기 위해서이다)

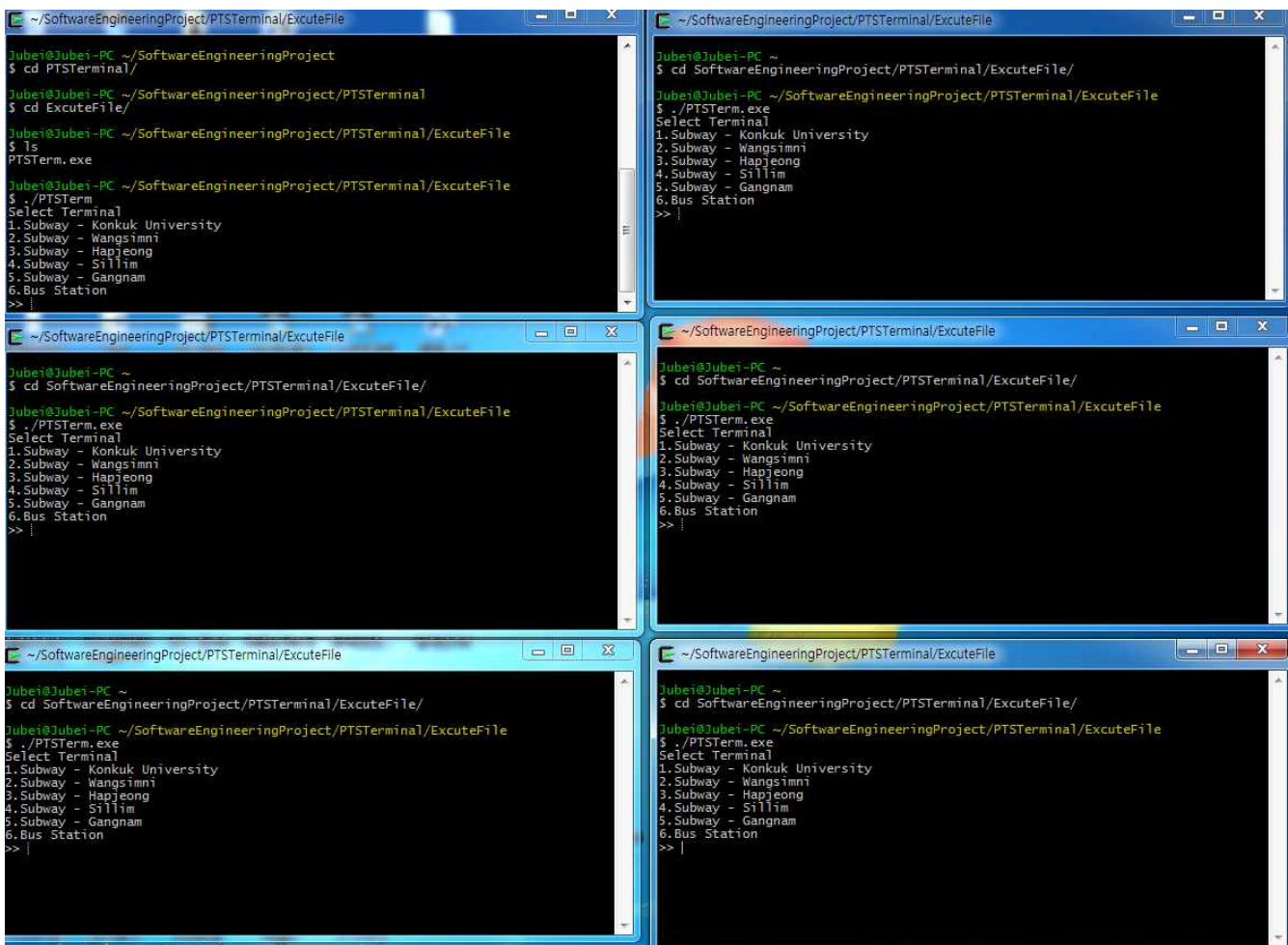
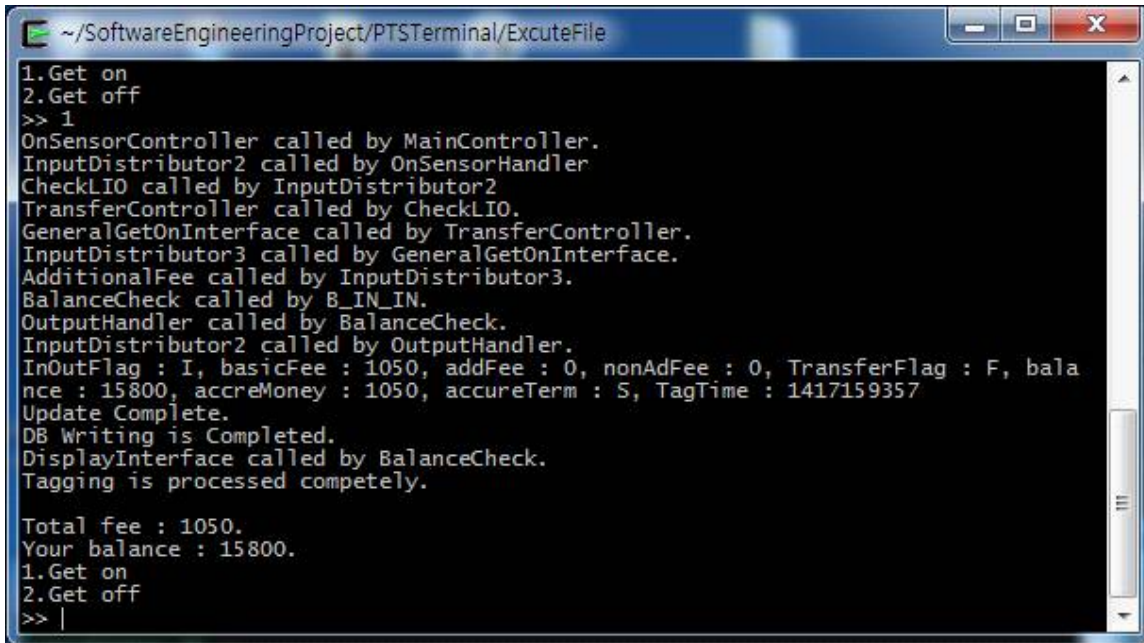


그림1) 6개의 cygwin 창을 키고 PTS를 띄운 모습

각 터미널이 해당 역 정보를 입력받고 초기화되면 1111.txt부터 6666.txt까지 각 역에 해당하는 DB파일이 초기화되면서 3분 카운팅이 시작된다. 이후 승차하고자 하는 역에서 1.Get on 메뉴를, 하차하고자 하는 역에서 2.Get off 메뉴를 이용하여 하차하면 된다.

예) 지하철 건대입구역(1번역) 승차 → 강남역(5번역) 하차 → 버스(6번역) 환승 승차 → 버스 하차

1) 건대입구로 설정한 1번 터미널 에서 1)GetOn을 선택하여 탑승한다. 탑승 프로세스가 처리되면 사용자 교통카드인 usrcard.txt 파일에 탑승 과정에 따른 정보가 업데이트가 되고, 건대입구역 데이터베이스 1111.txt 파일에 탑승정보가 기록된다.

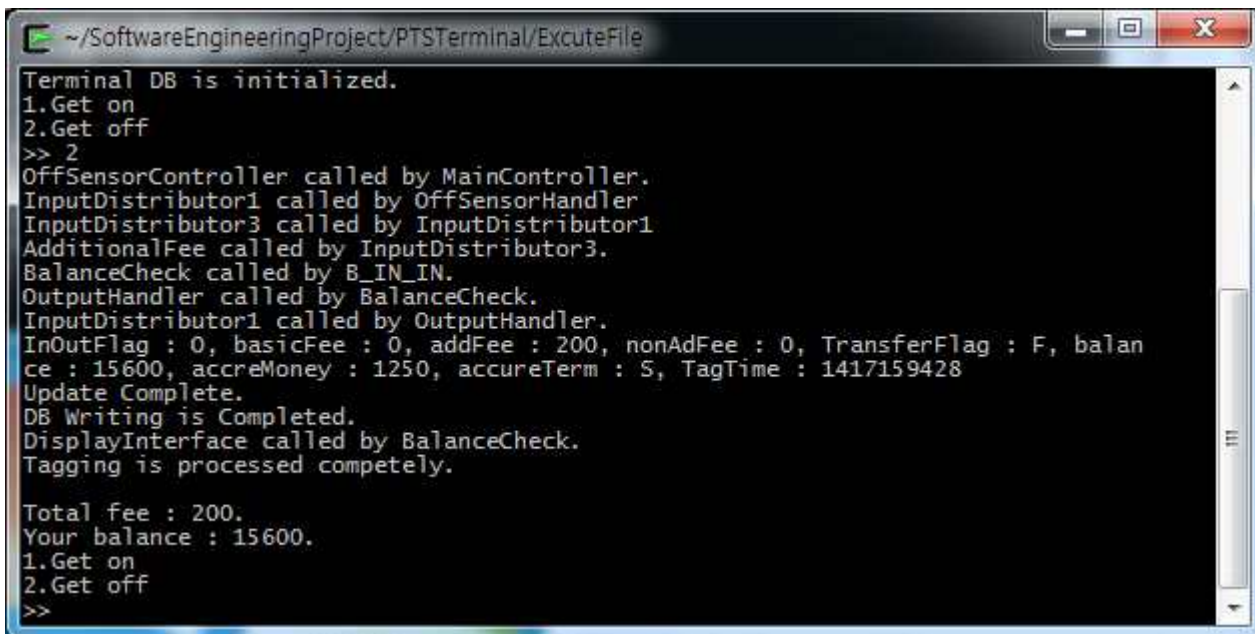


```
~/SoftwareEngineeringProject/PTSTerminal/ExcuteFile
1. Get on
2. Get off
>> 1
OnSensorController called by MainController.
InputDistributor2 called by OnSensorHandler
CheckLIO called by InputDistributor2
TransferController called by CheckLIO.
GeneralGetOnInterface called by TransferController.
InputDistributor3 called by GeneralGetOnInterface.
AdditionalFee called by InputDistributor3.
BalanceCheck called by B_IN_IN.
OutputHandler called by BalanceCheck.
InputDistributor2 called by OutputHandler.
InOutFlag : I, basicFee : 1050, addFee : 0, nonAdFee : 0, TransferFlag : F, balance : 15800, accreMoney : 1050, accureTerm : S, TagTime : 1417159357
Update Complete.
DB Writing is Completed.
DisplayInterface called by BalanceCheck.
Tagging is processed competely.

Total fee : 1050.
Your balance : 15800.
1. Get on
2. Get off
>> |
```

그림2) 건대입구역에서 승차한 화면. 함수 호출과정을 살피기 용이하도록 printf로 메시지들을 삽입하였음.(최종 데모버전이 나올 때에는 메시지 삭제 예정)

2) 5번역인 강남역 터미널에서 2)GetOff 메뉴를 선택하여 하차하게 되면, 터미널 창에는 해당 하차에 따른 추가요금과 잔액 등의 정보가 출력되고, 사용자 교통카드인 usrcard.txt파일도 갱신된다.



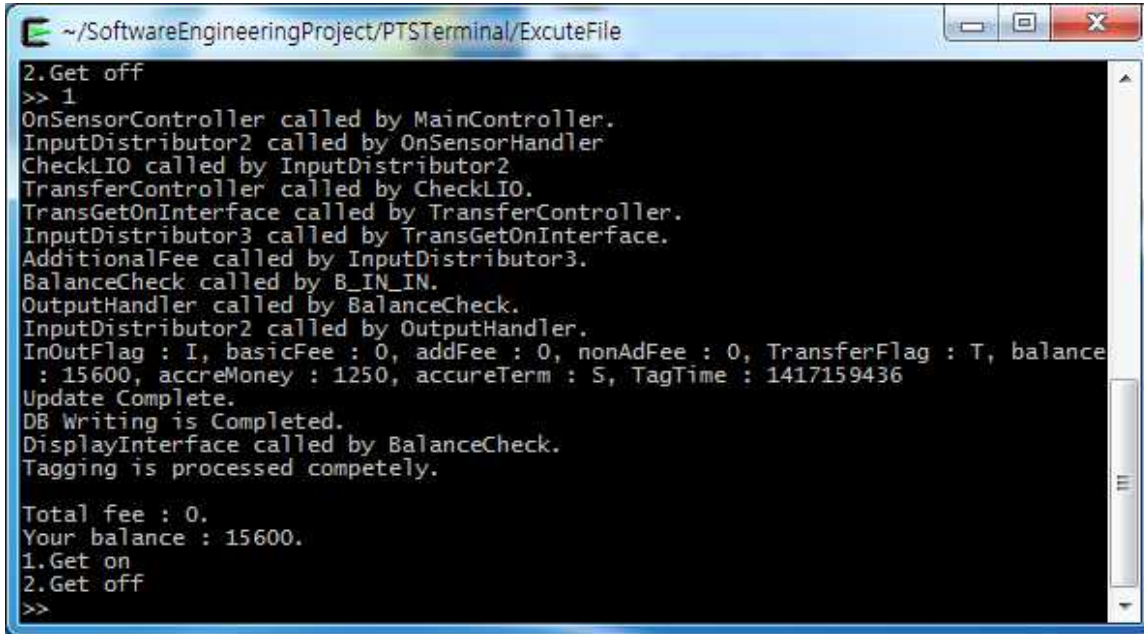
```
~/SoftwareEngineeringProject/PTSTerminal/ExcuteFile
Terminal DB is initialized.
1. Get on
2. Get off
>> 2
OffSensorController called by MainController.
InputDistributor1 called by OffSensorHandler
InputDistributor3 called by InputDistributor1
AdditionalFee called by InputDistributor3.
BalanceCheck called by B_IN_IN.
OutputHandler called by BalanceCheck.
InputDistributor1 called by OutputHandler.
InOutFlag : 0, basicFee : 0, addFee : 200, nonAdFee : 0, TransferFlag : F, balance : 15600, accreMoney : 1250, accureTerm : S, TagTime : 1417159428
Update Complete.
DB Writing is Completed.
DisplayInterface called by BalanceCheck.
Tagging is processed competely.

Total fee : 200.
Your balance : 15600.
1. Get on
2. Get off
>>
```

그림3) 강남역에서 하차한 화면. 추가요금이 부과되고 남은 잔액이 표시된다.

3) 6번역인 버스에 환승 승차하기 위해서는 5번 강남역에서 하차한 후 15초 안에 버스 터미널에서 승차하여야 한다. 만약 15초가 지날 경우 환승처리 되지 않고 일반승차 처리 되어 기본료를 다시 지불하게 된다.

이 과정을 거치면서 교통수단을 이용하다 3분이 지나게되면 터미널은 종료된다.



```
~/SoftwareEngineeringProject/PTSTerminal/ExcuteFile
2. Get off
>> 1
OnSensorController called by MainController.
InputDistributor2 called by OnSensorHandler
CheckLIO called by InputDistributor2
TransferController called by CheckLIO.
TransGetOnInterface called by TransferController.
InputDistributor3 called by TransGetOnInterface.
AdditionalFee called by InputDistributor3.
BalanceCheck called by B_IN_IN.
OutputHandler called by BalanceCheck.
InputDistributor2 called by OutputHandler.
InOutFlag : I, basicFee : 0, addFee : 0, nonAdFee : 0, TransferFlag : T, balance
: 15600, accreMoney : 1250, accureTerm : S, TagTime : 1417159436
Update Complete.
DB Writing is Completed.
DisplayInterface called by BalanceCheck.
Tagging is processed competely.

Total fee : 0.
Your balance : 15600.
1. Get on
2. Get off
>>
```

그림4) 버스 환승 승차한 화면. 환승이므로 기본료가 0원임을 확인할 수 있다.

## 2. 정산시스템

-간단한실행법

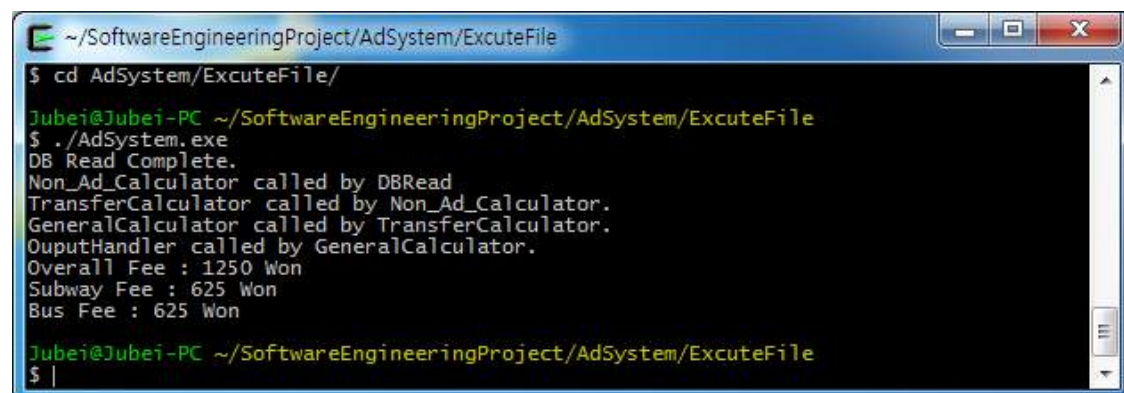
(1)PTS가 모두 종료된 후 정산시스템 AdSystem.exe 를 실행한다.

-상세설명

터미널들이 종료하게 되면 하루가 종료된 상황이기 때문에 정산시스템을 동작 시켜준다.

정산시스템은 각 단말기의 DB에 해당하는 1111~6666.txt 6개의 파일들을 읽어와서 정산을 시작한다. 정산이 종료된 후 터미널에 정산완료 정보가 Display되고, BusCompany.txt와 SubwayCompany.txt(버스회사, 지하철회사)에 해당하는 DB에 정산된 정보를 기록해준다.

아래 정산시스템 동작 예시 화면은 위에서 테스트하였던 건대입구->강남->버스환승->하차 의 과정에 따른 요금정산을 나타낸다. 기본료 1050원 + 지하철 추가이용료 200원 + 환승이용료 0원 = 1250원에 환승 규칙에 따라 지하철/버스 회사가 분배한 금액이다.



```
~/SoftwareEngineeringProject/AdSystem/ExcuteFile
$ cd AdSystem/ExcuteFile/
Jubei@Jubei-PC ~/SoftwareEngineeringProject/AdSystem/ExcuteFile
$ ./AdSystem.exe
DB Read Complete.
Non_Ad_Calculator called by DBRead
TransferCalculator called by Non_Ad_Calculator.
GeneralCalculator called by TransferCalculator.
OuputHandler called by GeneralCalculator.
Overall Fee : 1250 Won
Subway Fee : 625 Won
Bus Fee : 625 Won
Jubei@Jubei-PC ~/SoftwareEngineeringProject/AdSystem/ExcuteFile
$ |
```

그림5) 정산시스템 동작 화면

```
~/SoftwareEngineeringProject
Jubei@Jubei-PC ~/SoftwareEngineeringProject
$ cat BusCompany.txt
Date : Fri Nov 28 07:31:59 2014
Bus Income : 625 Won

Jubei@Jubei-PC ~/SoftwareEngineeringProject
$ cat SubwayCompany.txt
Date : Fri Nov 28 07:31:59 2014
Subway Income : 625 Won

Jubei@Jubei-PC ~/SoftwareEngineeringProject
$
```

그림6) 정산시스템이 정산을 마친 후 버스/지하철 회사로 전달되는 정산 데이터베이스