

RM+ requirements management

정세진
김재엽

1. User requirements
2. Functional requirements
3. Design Architecture

INDEX

Introduce one

USER REQUIREMENTS

User Requirements

- 대중교통시스템 (PTS)의 지하철 및 버스 정산 시스템
- 구현할 시스템의 목적에 맞는 User Requirements를 생성
- 요금 부과와 정산, 요금 부과를 위한 카드 인식 및 환승, 추가 요금 계산이 필요

User Requirements

User Requirement				
SR	하루 요금 정산	40	하루 단위로 버스/지하철 탑승/환승 요금을 정산할 수 있다	Test
SR	버스 요금 부과	43	승차시 요금을 부과 할 수 있어야 한다 / 필요한 경우 미정산 요금도 부과 할 수 있다	Test
SR	단말 정보 표현	44	버스/지하철의 카드 단말기에 정보를 표시 할 수 있어야 한다 사용금액, 잔액/시간, digital clock, 교통카드 기록 장치	Test
SR	버스 추가 요금 부과	46	환승 후 하차시 추가 요금을 부과 할 수 있어야 한다	Test
SR	사용자 카드 인식	47	단말을 통해 사용자의 교통 카드를 인식 할 수 있어야 한다	Test
SR	환승 요금	55	환승의 경우 승차시에는 추가 요금 없이 환승이 가능 할 수 있어야 한다	Test
SR	지하철 요금 부과	56	지하철 승차시 요금을 부과 할 수 있어야 한다 / 필요한 경우 미정산 요금도 부과 할 수 있다	Test
SR	지하철 추가 요금 부과	57	환승 후 지하철 하차시 추가 요금을 부과 할 수 있어야 한다	Test
SR	탑승 여부 결정	62	요금 부과 결과에 따라 탑승 여부를 결정 할 수 있다	Test
SR	카드 정보 변경	69	탑승 여부가 결정 되면 사용자 카드에 정보 갱신이 이루어 져야 한다	Test
SR	단말 누적 정보 갱신	71	단말기의 누적 결제 정보가 승/하차시 추가 되어야 한다	Test
SR	정산 요금 분배	72	정산된 요금을 버스/지하철에 분배 할 수 있다	Test
SR	환승 여부 결정	73	환승 시간에 따라 환승 여부를 결정 할 수 있다	Test

Introduce one

FUNCTIONAL REQUIREMENTS

Functional Requirements

- User Requirements를 만족시키기 위해 필요한 시스템의 기능 요구 사항
- User Requirements를 기반으로 작성 됨
- 탑승 시 카드 인식, 요금 부과, 요금 정산 으로 구성
- User Requirements와 하나 혹은 여러 개로 매칭시켜 모든 User Requirements를 반영함

Functional Requirements

Project - Default Edit

- System Architecture tmony
 - User Requirement
 - 하루 요금 계산
 - 버스 요금 부과
 - 단말 정보 표현
 - 버스 추가 요금 부과
 - 사용자 카드 인식
 - 환승 요금
 - 지하철 요금 부과
 - 지하철 추가 요금 부과
 - 합승 여부 결정
 - 카드 정보 변경
 - 단말 누락 정보 경신
 - 경신 요금 분배
 - 환승 여부 결정
 - Functional Requirements
 - 요금 계산
 - 단말 요금 계산
 - 경신 요금 분배
 - 카드 정보 인식
 - 카드에 저장된 정보 인식
 - 카드 인식시 단말에 정...
 - 요금 부과
 - 요금 계산
 - 요금 부과
 - 합승 여부 알림
 - 잔액 확인
 - 카드 정보 확인
 - 누락 결제 정보
 - 카드 정보 변경
 - 환승 여부 결정
 - Design Architecture

Functional Requirements 요금 부과

Requirement name	Id	Description	Verification Met...	Priority
요금 부과				
요금 계산	59	카드 정보를 이용해 부과할 요금을 계산한다. 표 1 버스 요금 규정 기본 요금 1050원	Test	
요금 부과	60	계산한 요금을 부과한다	Test	
합승 여부 알림	61	요금 정보와 잔액 정보를 이용하여 결정된 합승 여부를 알린다	Test	
잔액 확인	65	인식한 카드 정보의 잔액을 확인한다	Test	
카드 정보 확인	66	환승 여부, 승/하차 여부, 지하철/버스 여부, 미경산 여부, 추가요금 여부를 확인한다	Test	
누락 결제 정보	68	단말기에 결제 기록을 저장한다	Test	
카드 정보 변경	70	합승 여부가 결정에 따라서 카드의 잔액 및 합승 정보가 경신 된다	Test	
환승 여부 결정	75	카드 정보를 읽어 환승 여부를 결정한다	Test	

Baseline System Architecture tmony

- Realized
- (T) User Requirement

Requirements

Requirement name
(T) 하루 요금 계산
(T) 버스 요금 부과
(T) 단말 정보 표현
(T) 버스 추가 요금 부과
(T) 사용자 카드 인식
(T) 환승 요금
(T) 지하철 요금 부과
(T) 지하철 추가 요금 부과
(T) 합승 여부 결정
(T) 카드 정보 변경
(T) 단말 누락 정보 경신
(T) 경신 요금 분배
(T) 환승 여부 결정

Functional Requirements

카드 정보 인식					
§	카드에 저장된 정보 인식	54	태그시 교통카드에 저장된 정보를 인식한다 	Test	
§	카드 인식시 단말에 정보 표현	64	카드 인식시 단말에 필요한 정보를 표현한다	Test	
요금 부과					
§	요금 계산	59	카드 정보를 이용해 부과할 요금을 계산한다 표 1 버스 요금 규정 기본 요금 1050원	Test	
§	요금 부과	60	계산한 요금을 부과한다	Test	
§	환승 여부 알림	61	요금 정보와 잔액 정보를 이용하여 결정된 환승 여부를 알린다	Test	
§	잔액 확인	65	인식한 카드 정보의 잔액을 확인한다	Test	
§	카드 정보 확인	66	환승 여부, 승/하차 여부, 지하철/버스 여부, 미정산 여부, 추가요금 여부를 확인한다	Test	
§	누적 결제 정보	68	단말기에 결제 기록을 저장한다 	Test	
§	카드 정보 변경	70	환승 여부가 결정에 따라서 카드의 잔액 및 환승 정보가 갱신 된다	Test	
§	환승 여부 결정	75	카드 정보를 읽어 환승 여부를 결정한다	Test	

Functional Requirements

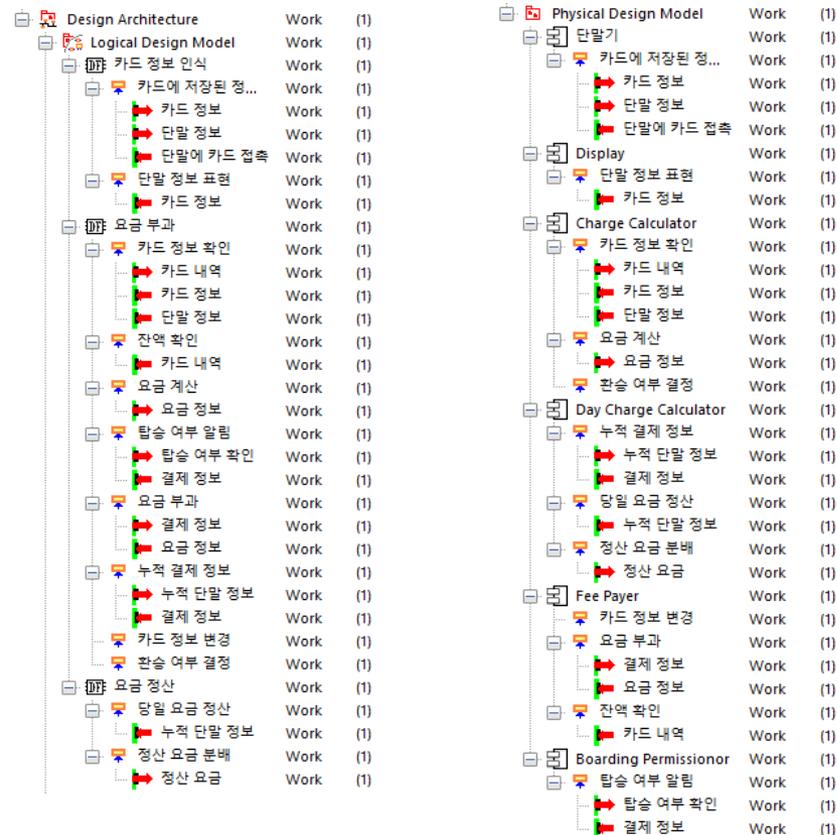
 요구 정산				
  중립 요구 정산	53	매 자정이 되면 각 단말의 누적 정보를 받아서 하루 동안 사용 요금의 정산과 리베이가 되어야 한다	Test	
 정산 요구 목록	74	정산된 요금을 버스/지하철로 환산한다	Test	

Introduce one

DESIGN ARCHITECTURE

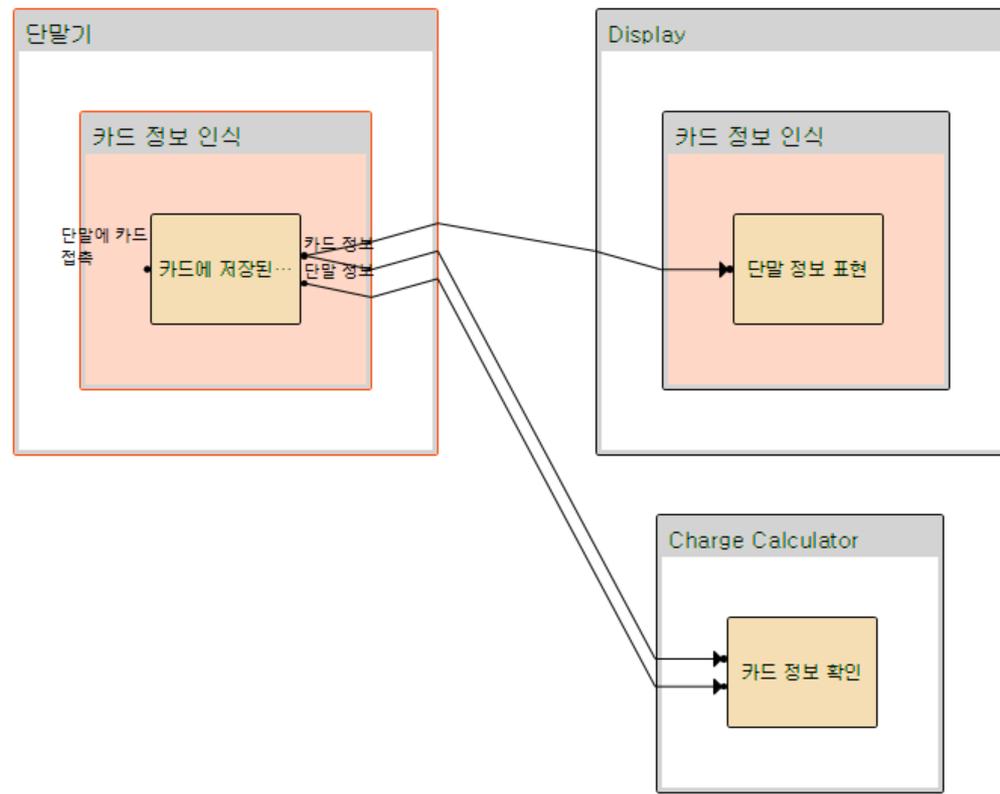
Design Architecture

- 도구에서 지원하는 방식에 따라 시스템을 디자인
- Logical design model을 작성 후 physical design model과 연결



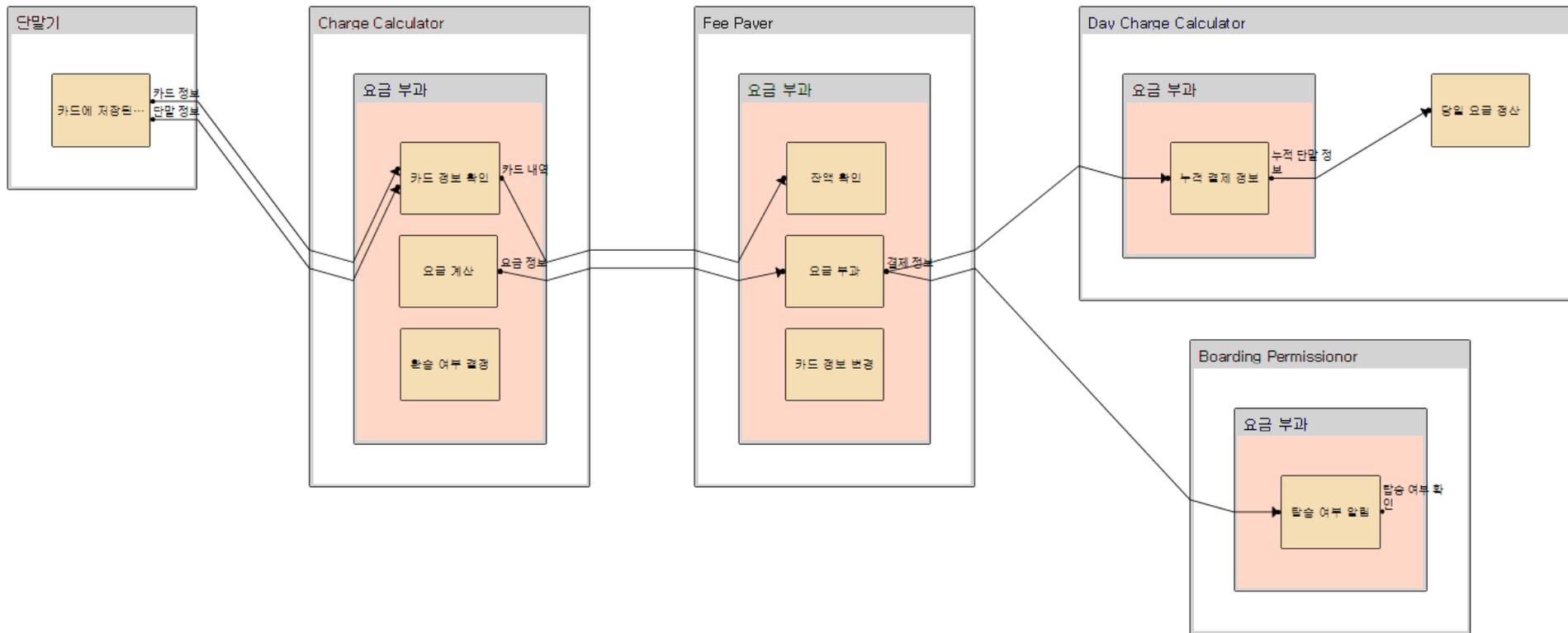
Design Architecture - logical

- 카드 정보 인식



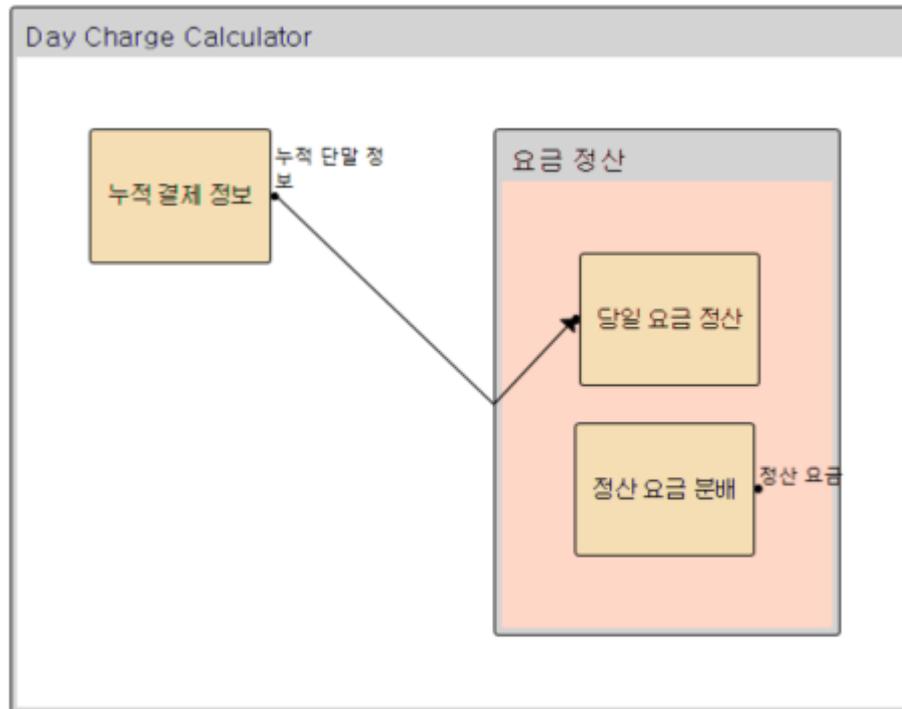
Design Architecture - logical

- 카드 정보 인식



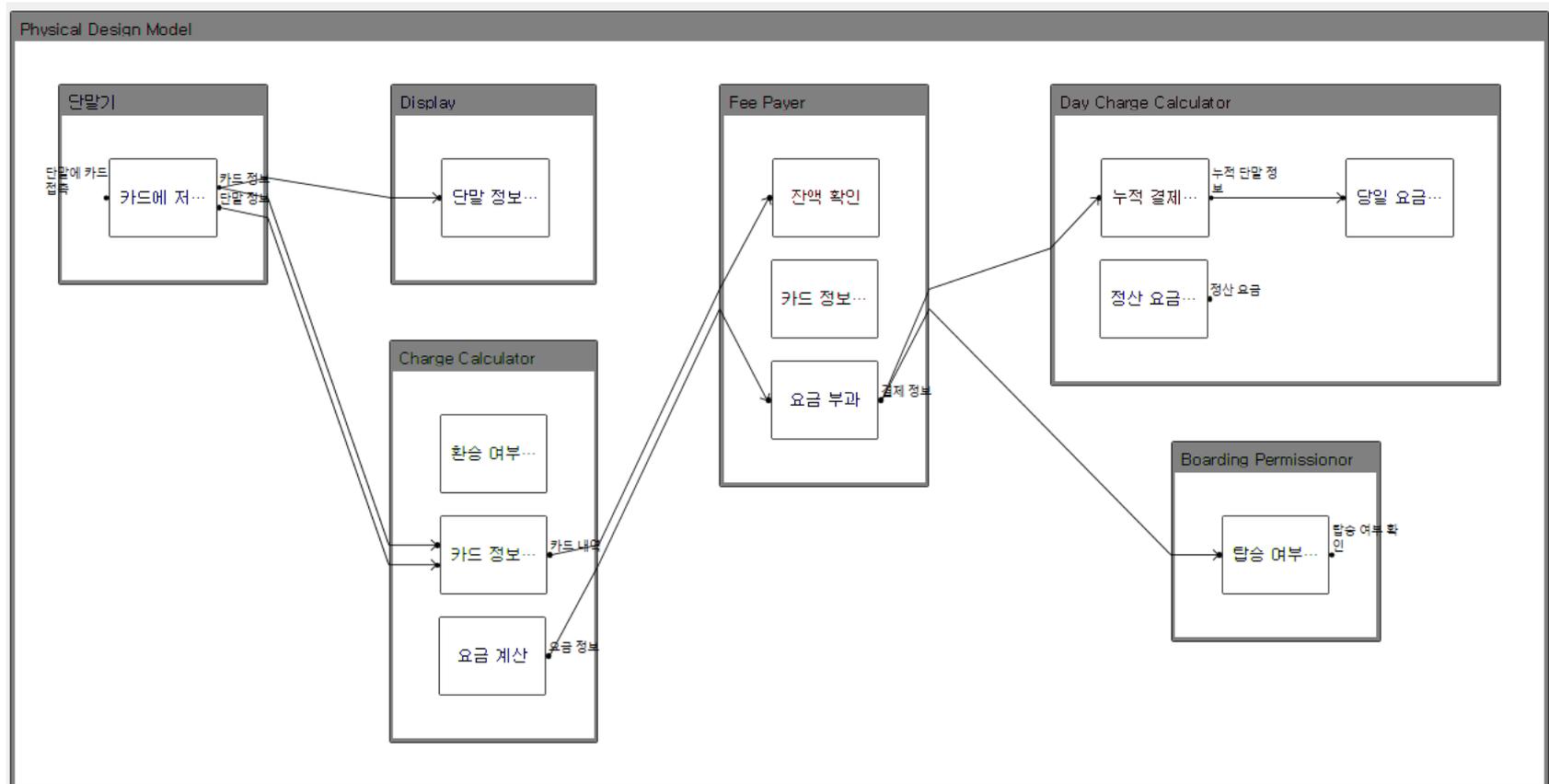
Design Architecture - logical

- 요금 정산



Design Architecture - physical

- Physical design model



End

- 단계별로 requirements를 작성하고 user/function requirements의 연결을 쉽게 표시함으로써 User requirements반영이 용이함
- Send/receive(input/output)만 잘 지정하면 디자인 모델을 자동으로 그려주는 것이 인상적
- 하지만, 도구 사용에 불편한 점이 존재

Q & A

END