

# Open Your Closet

Team DAUM

201810924 김지훈

201810986 임재훈

201811194 이진욱

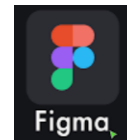
202010902 이한솔

## 목차

1. Open Your Closet 소개
2. 요구사항 분석
3. 기술 시연
4. 클래스 & 시퀀스 다이어그램
5. 한계점, 배운 점
6. 역할분담

# 소개

프로젝트명 : Open Your Closet



# 1. 요구사항 분석

## 요구사항 - 개요 및 고객 기능 요구사항

### 시스템 개요

자신의 의류를 사진을 포함하여 다양한 카테고리에 따라 분류, 입을 옷을 추천 받을 수 있는 통합 의류 관리 프로그램 개발

### 고객 기능 요구사항

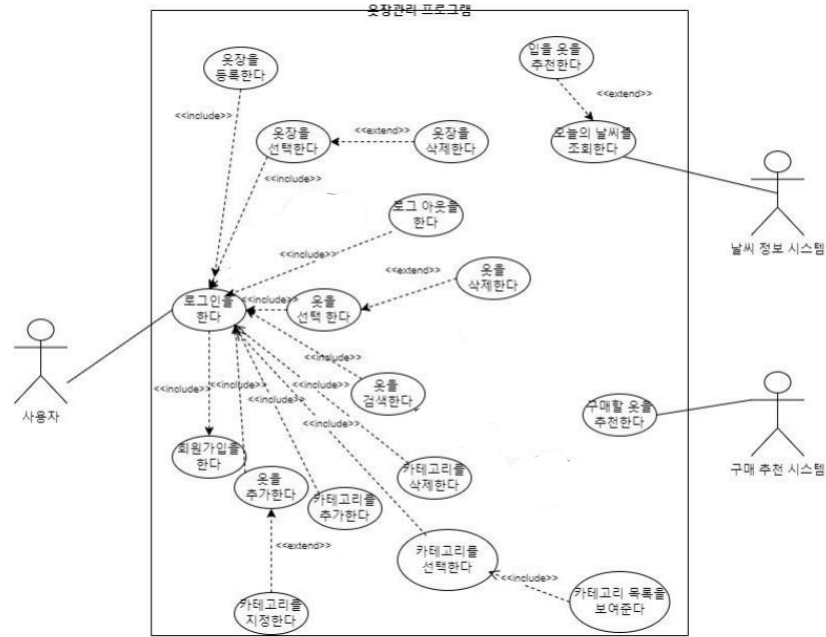
구분	상세	비고
통합 로그인	구글, 네이버, 카카오등이나 사이트 자체의 회원가입을 통해 로그인을 할 수 있게 한다. 로그인 실패 시 다시 로그인을 진행할 수 있도록 한다.	
옷 관리	가지고 있는 옷들을 사진과 함께 등록해 확인 할 수 있게 하고 수정, 삭제도 가능 하게 한다.	
카테고리 별 분류	등록한 옷들을 옷의 카테고리에 맞춰서 분류 할 수 있게 하고 검색해서 찾을 수 있게 한다.	
입을 옷 추천	하루의 날씨에 따라서 등록된 옷들의 정보를 토대로 오늘 입을 옷을 추천해 준다.	

## 유스케이스 목록

ID	유스케이스 명	설명	우선 순위
UC001	회원가입을 한다	페이스북, 구글을 사용한 로그인과 자체 사이트 로그인을 지원한다.	1
UC002	로그인을 한다	회원가입한 계정을 토대로 로그인을 한다.	1
UC005	옷을 추가한다	사용자는 옷을 추가할 때 이름, 옷장, 카테고리, 구입 날짜, 옷 이미지를 입력할 수 있다. 옷이 들어있는 옷장을 선택할 수 있다.	1
UC006	옷을 삭제한다	사용자는 제품 손상, 분실 등으로 옷장에 더 이상 존재하지 않는 옷을 삭제할 수 있다.	1
UC008	입을 옷을 추천한다	사용자가 선택한 옷장들에 존재하는 옷들을 기반으로 입을 옷을 코디해준다	4
UC009	옷장을 등록한다	옷장이 여러개일 수도 있기 때문에 회원가입 후 옷장을 반드시 한개 이상 등록하게 한다. 옷장에 별명을 등록할 수 있다.ex)외투 옷장에 옷칸이 여러개일 수 있기 때문에 옷장 등록시 반드시 옷칸을 한개 이상 등록하게 한다. 사용자는 옷장을 등록할때 자신의 옷장 구조와 동일한 레이아웃을 적용시킬 수 있다	3

UC010	옷장을 삭제한다	사용하지 않는 옷장을 삭제하고, 남아있는 옷이 있다면 사용자가 선택한 옷장으로 이동시킨다	3
UC011	옷장을 선택한다	옷장을 조회하기 위해 옷장의 리스트에서 원하는 옷장을 선택한다	3
UC012	옷을 검색한다	사용자는 등록된 옷을 검색할 수 있다.	1
UC013	오늘의 날씨를 조회한다	날씨 정보 중 옷과 관련된 정보를(온,습도, 우천 여부, 풍속, 체감 온도 등)표시하고 날씨에 맞는 옷을 추천한다	2
UC014	카테고리를 지정한다	옷의 종류와 특성별로 목록을 나열한다(위의 정보에 기반)	2
UC016	옷을 선택한다	등록된 옷을 선택해서 정보를 볼 수 있다	
UC017	로그아웃을 한다	로그아웃을 할 수 있게 한다	
UC020	카테고리를 선택한다	내가 보고싶은 카테고리 목록을 선택한다	
UC021	카테고리 목록을 보여준다	선택한 카테고리 안에있는 옷들의 목록을 보여준다	

# 유스케이스 다이어그램



## 2. 기술 시연 & 유스케이스 기술 소개



# 유스케이스 기술 – 회원가입, 로그인 관련

UC001: 회원가입을 한다

설명	페이스북, 구글을 사용한 로그인과 자체 사이트 로그인을 지원한다.	
관련 액터	사용자	
사전 조건	-	
사후 조건	로그인 화면으로 돌아간다.	
기본 흐름	B01	사용자는 회원가입 버튼을 클릭한다.
	B02	시스템은 회원가입 창을 띄워준다.
	B03	사용자는 창의 양식을 작성해 가입완료 버튼을 누르거나 외부 시스템과 연동 버튼을 클릭해 해당 연동을 완료한다
	B04	시스템은 회원가입 완료 문구를 출력해주고, 데이터를 저장한다.
대안 흐름	A01	B02이후 취소 버튼으로 이전 화면으로 돌아갈 수 있다.
예외 흐름	E01	B03에서 양식이 부족하거나 연동 실패 시 경고창을 띄워준다.
시나리오	SN001	B01-B02-B03-B04
	SN002	B01-B02-*** -A01
	SN003	B01-B02-B03-E01-B03-B04

UC002: 로그인을 한다

설명	회원이 가입한 계정을 토대로 로그인을 한다.	
관련 액터	사용자	
사전 조건	로그인이 필요한 기능 접근 시	
사후 조건	로그인 전 접근 시도한 기능으로 돌아간다.	
기본 흐름	B01	사용자는 로그인 버튼을 클릭하거나 로그인이 필요한 기능에 접근한다.
	B02	시스템은 로그인 창을 띄워준다.
	B03	사용자는 ID/PW를 작성하거나 연동 로그인을 이용해 로그인 버튼을 클릭한다.
	B04	시스템은 회원정보와 일치하는 경우 로그인을 승인한다.
대안 흐름	A01	B02이후 취소 버튼으로 이전 화면으로 돌아갈 수 있다.
	A02	B02에서 회원 가입 버튼 클릭 시 회원 가입으로 이동한다.
예외 흐름	E01	B03에서 ID/PW가 불일치하거나 연동 실패 시 경고창을 띄워준다.
	E02	B03에서 ID/PW 미입력시 경고창을 띄운다.
시나리오	SN001	B01-B02-B03-B04
	SN002	B01-B02-*** -A01
	SN003	B01-B02-B03-E01-B03-B04

UC017: 로그아웃을 한다

설명	로그아웃을 할 수 있게 한다.	
관련 액터	사용자	
사전 조건	로그인을 한다	
사후 조건	초기 화면으로 돌아간다.	
기본 흐름	B01	사용자는 로그아웃 버튼을 누른다.
	B02	시스템은 사용자를 로그아웃 시킨다.
대안 흐름	-	-
예외 흐름	-	-
시나리오	SN001	B01-B02

## 유스케이스 기술 – 옷 관련

UC005: 옷을 추가한다

<b>설명</b>	사용자는 옷을 추가할 때 이름, 옷장, 카테고리, 구입 날짜, 옷 이미지를 입력할 수 있다. 옷이 들어있는 옷장을 선택할 수 있다.
<b>관련 액터</b>	사용자
<b>사전 조건</b>	로그인한다.
<b>사후 조건</b>	이전 화면으로 돌아간다.
<b>기본 흐름</b>	B01 사용자는 옷 추가 기능에 접근한다.
	B02 시스템은 옷 추가 화면을 표시한다.
	B03 사용자는 옷을 추가할때 이름, 옷장, 카테고리, 구입 날짜, 옷 이미지를 입력한다.
	B04 시스템은 해당 옷 정보를 데이터에 저장한다.
<b>대안 흐름</b>	A01 B02이후 취소 버튼으로 이전 화면으로 돌아갈 수 있다.
	A02 B03에서 사진이나 구매일자 등의 특수한 내용 삽입 시 필요한 기능을 띄워준다.
<b>예외 흐름</b>	E01 B03에서 필수 내용이 기재되어 있지 않은 경우 경고창을 띄운다.
	E02 B03에서 동일한 이름의 옷이 있다면 경고창을 띄운다.
<b>시나리오</b>	SN001 B01-B02-B03-B04
	SN002 B01-B02-*** -A01
	SN003 B01-B02-B03-A02-B03-E01-B03-B04

UC016: 옷을 선택한다

<b>설명</b>	등록된 옷을 선택해서 정보를 볼 수 있다
<b>관련 액터</b>	사용자
<b>사전 조건</b>	로그인을 한다
<b>사후 조건</b>	이전 화면으로 돌아간다.
<b>기본 흐름</b>	B01 사용자는 옷 선택 기능에 접근한다.
	B02 시스템은 사용자에게 옷에 대한 정보를 표시해준다
<b>대안 흐름</b>	A01 B02에서 취소 버튼을 눌러 이전 화면으로 돌아간다.
<b>예외 흐름</b>	- -
<b>시나리오</b>	SN001 B01-B02
	SN002 B01-B02-A01

UC006 : 옷을 삭제한다

<b>설명</b>	사용자는 제품 손상, 분실 등으로 옷장에 더이상 존재하지 않는 옷을 삭제할 수 있다.
<b>관련 액터</b>	사용자
<b>사전 조건</b>	옷을 선택한다
<b>사후 조건</b>	이전 화면으로 돌아간다.
<b>기본 흐름</b>	B01 사용자는 옷 삭제 기능에 접근한다.
	B02 시스템은 사용자에게 옷 삭제 여부를 묻는다.
	B03 사용자는 확인 버튼을 누른다.
	B04 시스템은 해당 옷 정보를 데이터에서 삭제한다.
<b>대안 흐름</b>	A01 B02에서 취소 버튼을 눌러 이전 화면으로 돌아간다.
<b>예외 흐름</b>	- -
<b>시나리오</b>	SN001 B01-B02-B03-B04

UC012: 옷을 검색한다

<b>설명</b>	사용자는 등록한 옷을 검색할 수 있다.
<b>관련 액터</b>	사용자
<b>사전 조건</b>	로그인한다.
<b>사후 조건</b>	이전 화면으로 돌아간다.
<b>기본 흐름</b>	B01 사용자는 옷 검색 기능에 접근한다.
	B02 시스템은 옷 검색 화면을 표시한다.
	B03 사용자는 해당 화면에서 검색할 옷을 입력 후 검색 버튼을 클릭한다.
	B04 시스템은 해당 내용에 해당하는 옷을 출력해준다.
<b>대안 흐름</b>	A01 B02이후 취소 버튼으로 이전 화면으로 돌아갈 수 있다.
<b>예외 흐름</b>	E01 B03에서 검색된 옷이 없을 시 없다고 출력해준다.
<b>시나리오</b>	SN001 B01-B02-B03-B04
	SN002 B01-B02-*** -A01
	SN003 B01-B02-B03-E01-B03-B04

# 유스케이스 기술 - 옷장 관련

UC009 : 옷장을 등록한다

설명	옷장이 여러개일 수도 있기 때문에 회원가입 후 옷장을 반드시 한개 이상 등록하게 한다. 옷장에 별명을 등록할수 있다.ex)외투	
관련 액터	사용자	
사전 조건	UC002	
사후 조건		
기본 흐름	B01	옷장 등록 메뉴를 선택한다.
	B02	옷장에 별명을 설정한다.
	B03	기본 옷장으로 설정할지 여부를 입력받는다.
	B04	옷장을 옷장 리스트에 추가한다.
	B05	프로세스를 종료한다.
대안 흐름	A01	별명이 설정되지 않은경우 default값인 옷장n(n=1,2,3,...)을 사용한다.
예외 흐름	E01	입력받은 별명이 이미 존재하는 별명이라면 에러 메시지를 화면에 출력하고 입력을 다시받는다.
시나리오	SN001	B01-B02-B03-B04-B05
	SN002	B01-A01-B03-B04-B05
	SN003	B01-B02-E01-B04-B05

UC010: 옷장을 삭제한다

설명	사용하지 않는 옷장을 삭제하고 남아있는 옷이 있다면 사용자가 선택한 옷장으로 이동시킨다.	
관련 액터	사용자	
사전 조건	UC011	
사후 조건		
기본 흐름	B01	삭
	B02	남아 있는 옷들을 선택한 옷장으로 이동시킨다.
	B03	프로세스를 종료한다.
대안 흐름	A01	남아 있는 옷들을 이동시킬 옷장이 설정되지 않은 경우 기본 옷장으로 이동시킨다.
	A02	선택한 옷장이 기본 옷장인경우 기본옷장은 삭제할 수 없다는 메시지를 화면에 출력한뒤B03을 실행한다.
예외 흐름	E01	선택한 옷장이 유일한 옷장인 경우 에러메시지를 화면에 출력하고 UC011로 돌아간다.
시나리오	SN001	B01-B02-B03
	SN002	B01-A01-B03
	SN003	B01-A02-B03
	SN004	B01-E01-B03

UC011: 옷장을 선택한다

설명	사용하지 않는 옷장을 삭제하고 남아있는 옷이 있다면 사용자가 선택한 옷장으로 이동시킨다.	
관련 액터	사용자	
사전 조건	UC011	
사후 조건		
기본 흐름	B01	옷장 삭제 메뉴를 선택한다.
	B02	남아 있는 옷들을 선택한 옷장으로 이동시킨다.
	B03	프로세스를 종료한다.
대안 흐름	A01	남아 있는 옷들을 이동시킬 옷장이 설정되지 않은 경우 기본 옷장으로 이동시킨다.
	A02	선택한 옷장이 기본 옷장인경우 기본옷장은 삭제할 수 없다는 메시지를 화면에 출력한뒤B03을 실행한다.
예외 흐름	E01	선택한 옷장이 유일한 옷장인 경우 에러메시지를 화면에 출력하고 UC011로 돌아간다.
시나리오	SN001	B01-B02-B03
	SN002	B01-A01-B03
	SN003	B01-A02-B03
	SN004	B01-E01-B03

## 유스케이스 기술 - 카테고리 관련

UC020: 카테고리 선택한다

설명	선택한 카테고리 안에 있는 옷들의 목록을 보여준다.	
관련 액터	사용자	
사전 조건	카테고리를 선택한다.	
사후 조건	-	
기본 흐름	B01	사용자는 원하는 카테고리명을 선택한다.
	B02	시스템은 해당 카테고리 안에 있는 옷을 보여준다.
대안 흐름	A01	B02에서 닫기 버튼으로 이전 화면으로 돌아간다.
예외 흐름	E01	B02에서 해당 카테고리에 옷이 없을 경우 없음을 알려줌.
시나리오	SN001	B01-B02
	SN002	B01-B02-E01-B02-A01

UC021: 카테고리 목록을 보여준다

설명	선택한 카테고리 안에 있는 옷들의 목록을 보여준다.	
관련 액터	사용자	
사전 조건	카테고리를 선택한다.	
사후 조건	-	
기본 흐름	B01	사용자는 원하는 카테고리명을 선택한다.
	B02	시스템은 해당 카테고리 안에 있는 옷을 보여준다.
대안 흐름	A01	B02에서 닫기 버튼으로 이전 화면으로 돌아간다.
예외 흐름	E01	B02에서 해당 카테고리에 옷이 없을 경우 없음을 알려줌.
시나리오	SN001	B01-B02
	SN002	B01-B02-E01-B02-A01

UC014: 카테고리를 지정한다

설명	옷의 종류와 특성별로 목록을 나열한다(위의 정보에 기반)	
관련 액터	사용자	
사전 조건	-	
사후 조건	이전 화면으로 돌아간다.	
기본 흐름	B01	사용자는 해당 기능에 접근한다.
	B02	시스템은 카테고리 창을 띄워준다.
	B03	사용자는 원하는 옷의 종류나 특성별로 카테고리를 선택하여 옷을 나열시킨다.
대안 흐름	A01	B02이후 취소 버튼으로 이전 화면으로 돌아갈 수 있다.
시나리오	SN001	B01-B02-B03

## 유스케이스 기술 – 기타 기능

### UC008 : 입을 옷을 추천한다

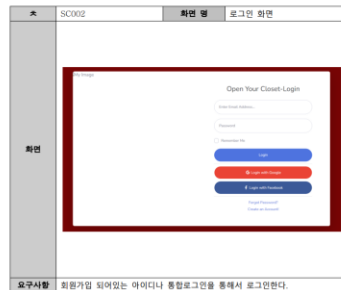
설명	사용자가 선택한 옷장들에 존재하는 옷들을 기반으로 입을 옷을 코디해 준다	
관련 액터	날씨 정보 시스템	
사전 조건	UC013	
사후 조건		
기본 흐름	B01	입을 옷 추천을 클릭한다
	B02	오늘의 날씨를 조회한다.
	B03	B02의 결과값과 '날씨' 정보가 같은 옷들을 조회한다.
	B04	카테고리별로 하나씩, 색의 조화를 고려하여 옷의 조합을 만든다.
	B05	최대 3개의 코디 조합을 화면에 출력한다.
	B06	사용자의 선택을 받는다.
	B07	프로세스를 종료한다.
대안 흐름	A01	카테고리 별로 하나씩의 옷이 없는 경우 완전한 코디를 추천할 수 없기 때문에 옷만 추천한다.
	A02	날씨 조회의 결과값에 해당하는 옷이 없는 경우 날씨를 고려하지 않는다.
	A03	조화되는 색이 없는 경우 색을 고려하지 않는다.
예외 흐름	E01	날씨 정보를 불러오지 못했다면, UC013_E01 를 실행한다
시나리오	SN001	B01-B02-B03-B04-B05-B06-B07
	SN002	B01-B02-B03-B04-A01-B06-B07
	SN003	B01-B02-B03-A02-B04-B05-B06-B07
	SN004	B01-B02-B03-B04-A03-B05-B06-B07
	SN005	B01-B02-E01-B07

### UC013: 오늘의 날씨를 조회한다

설명	날씨 정보 중 옷과 관련된 정보를(온,습도, 우천 여부, 풍속, 체감 온도 등)표시하고 날씨에 맞는 옷을 추천한다	
관련 액터	날씨 정보 시스템	
사전 조건	-	
사후 조건	이전 화면으로 돌아간다.	
기본 흐름	B01	사용자는 날씨 조회 기능에 접근한다.
	B02	시스템은 날씨 정보 시스템에서 날씨를 불러와 필요한 정보를 가공하여 표시해주며 날씨에 맞는 옷을 추천해준다.
대안 흐름	-	-
예외 흐름	E01	B02에서 정보 불러오기 실패 시 실패했다고 알려준다.
시나리오	SN001	B01-B02
	SN002	B01-B02-E01

# 사용자 인터페이스 요구사항, 화면 목록 예시

ID	화면 명	관련 유스케이스 ID	설명
SC002	로그인 화면		로그인은 팝업창을 통해 사이트 아이디 혹은 카카오톡, 네이버, 구글 통합로그인으로 한다
SC003	회원가입 화면		사이트 아이디를 만들기 위한 회원 가입 화면
SC004	홈 화면		로그인을 했을 때 나오는 프로그램의 홈 화면
SC005	옷장 화면		내가 만든 옷장속에 분류 되어있는 옷들을 볼 수 있는 화면
SC006	카테고리 화면		내가 분류한 카테고리별로 옷들을 볼 수 있는 화면
SC007	전체 옷 목록		내가 등록한 모든 옷들을 볼 수 있는 화면
SC008	옷장 목록		내가 만든 옷장들을 볼 수 있는 화면
SC009	옷 상세 정보		등록 되어 있는 옷을 클릭시 나오는 등록할 때 입력한 옷들의 정보들을 보여주는 화면
SC010	옷 등록 화면		옷을 등록 할 수 있는 화면

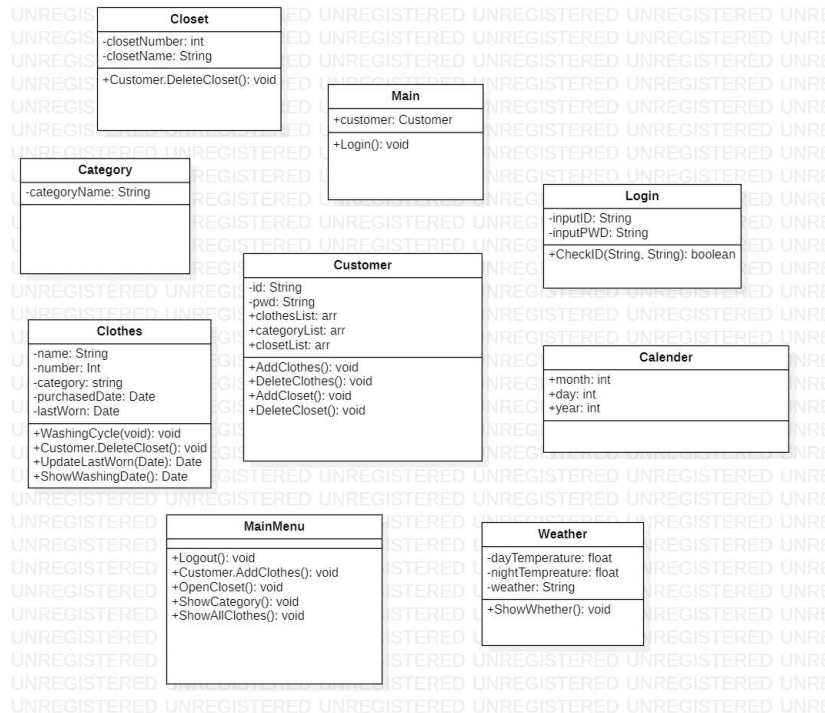


### 3. 클래스 다이어그램

# Entity 클래스

회원 정보, 옷장, 옷, 날씨 정보 등  
전반적인 데이터들을 저장, 삭제,  
수정하는 클래스들

- Closet
- Main
- Category
- Customer
- Clothes
- MainMenu
- Weather
- Login
- Calendar

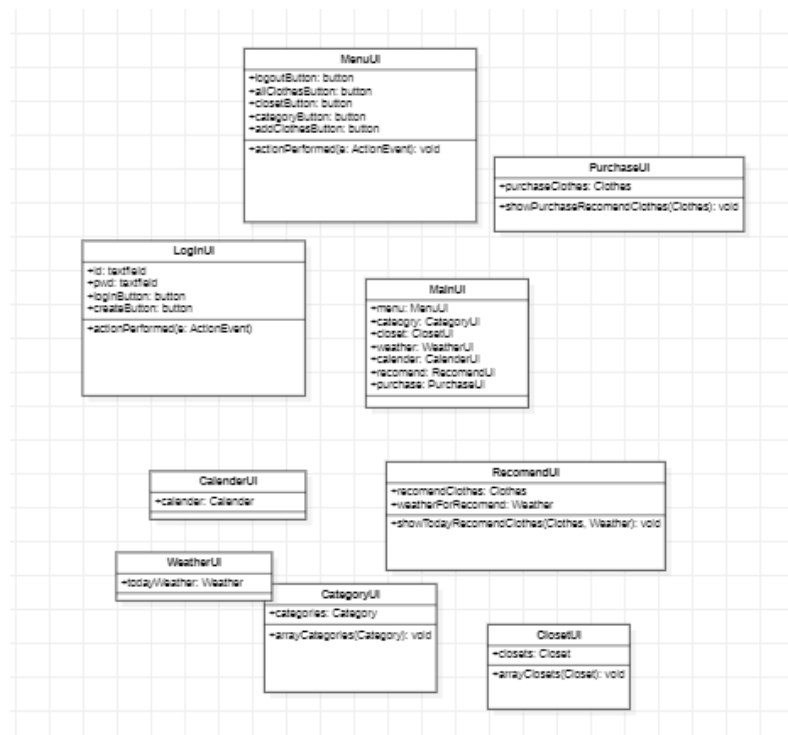




## 경계 클래스 – 메인 화면

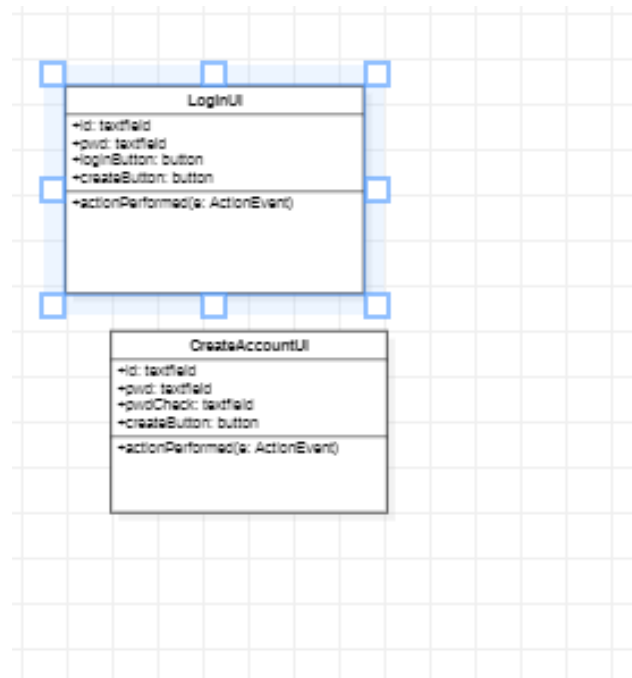
사용자에게 보여지는 UI들과  
그에 대한 이벤트를 담당

- MainUi
- MenuUi
- LoginUi
- **PurchasUi**
- CalenderUi
- WeatherUi
- CategoryUi
- ClosetUi
- **RecomendUi**



## 경계 클래스 - 로그인 화면

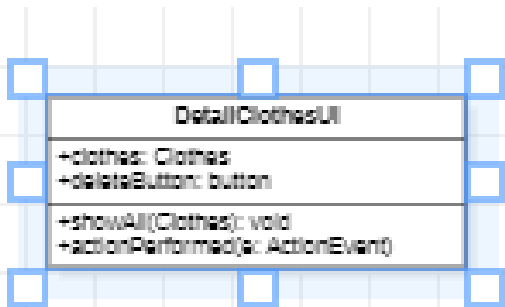
- LoginUi
- CreateAccountUi



## 경계 클래스 - 상세 옷, 옷 추가 화면

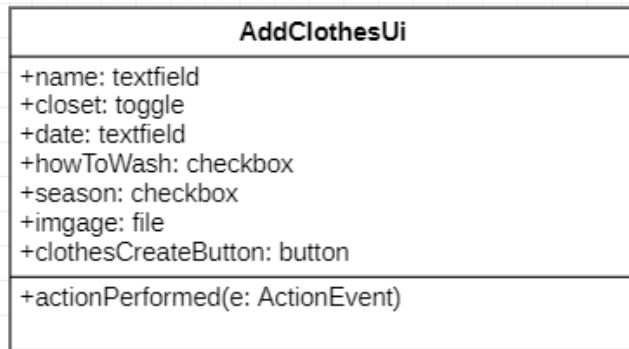
### (1) 상세 옷 화면

#### - DetailClothesUi



### (2) 옷 추가화면

#### - AddClothesUi



## 제어 클래스 - 메인 컨트롤러, 로그인 컨트롤러

### (1) 메인 컨트롤러

- mainController

mainController
+mainUi: MainUi +menu: Menu +closet: Closet +weather: Weather +calender: Calender +recomend: Recomend +purchase: Purchase
+showMenu(menu: Menu): void +showCloset(closet: Closet): void +showWeather(weather: Weather): void +showCalender(calender: Calender): void +showRecomend(recomend: Recommen): void +showPurchase(purchase: Purchase): void

### (2) 로그인 컨트롤러

- createAccountController
- loginController

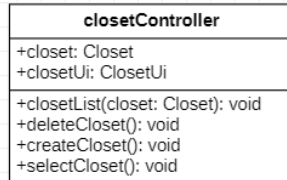
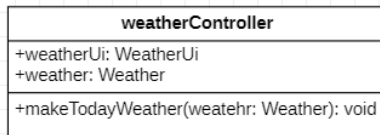
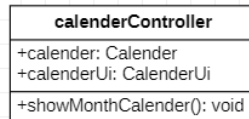
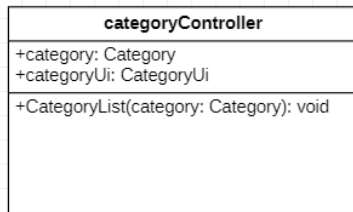
createAccountController
+createAccountCustomer: Customer +createUi: CreateUi
+checkPwd(pwd: String, checkPwd: String): void +checkId(id: String): void +createAccount(): Customer

loginController
+loginCustomer: Customer +loginUi: LoginUi
+isLogin(id: String, pwd: String): void +login(loginCustomer): void

경계 클래스와 Entity 클래스  
사이에서 흐름 제어 및 통제 역할

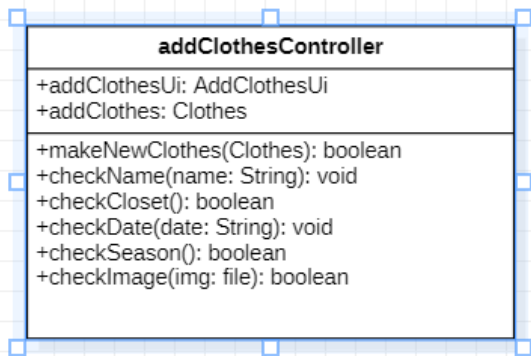
## 제어 클래스 - 메인화면 컨트롤러

- categoryController
- calenderController
- weatherController
- closetController

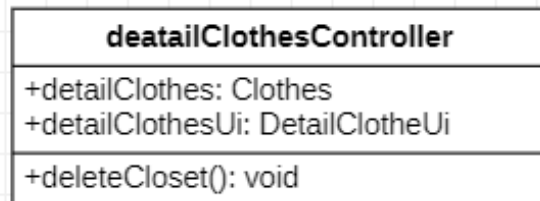


## 제어 클래스 - 옷 추가, 옷 상세 컨트롤러

### (1) 옷 추가 컨트롤러



### (2) 옷 상세 컨트롤러

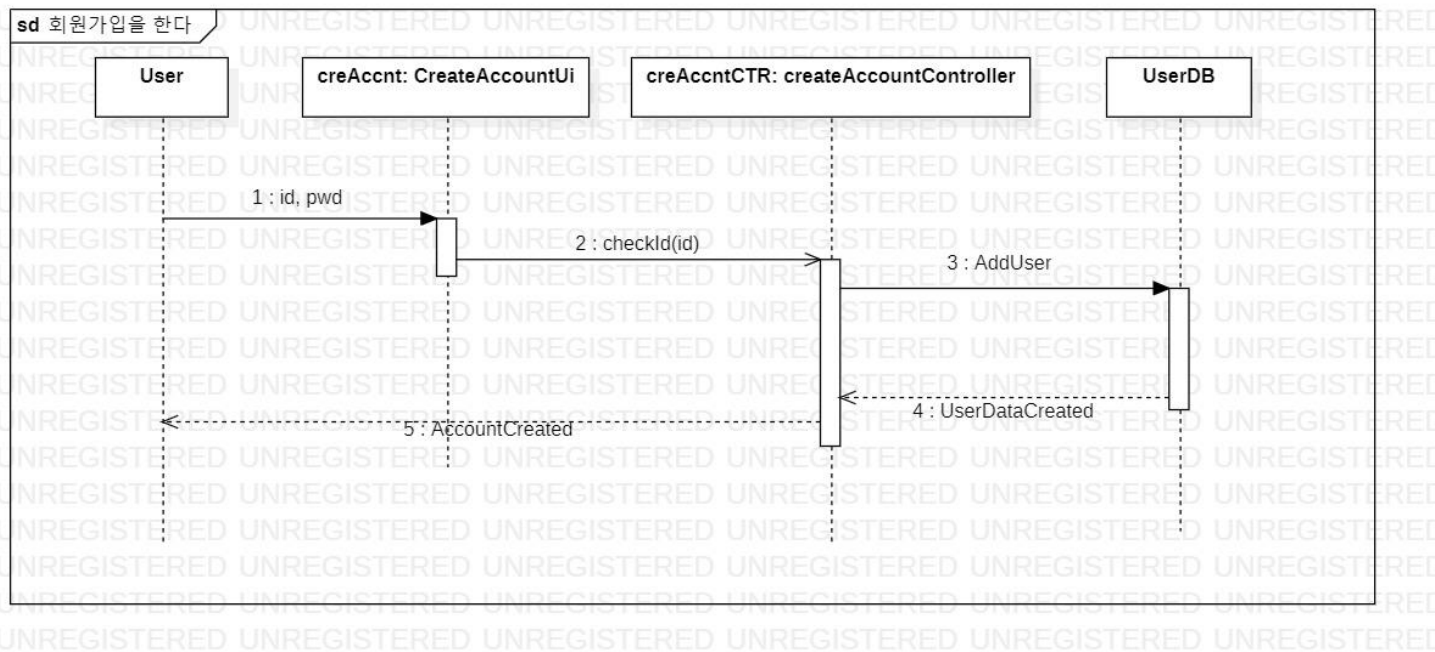




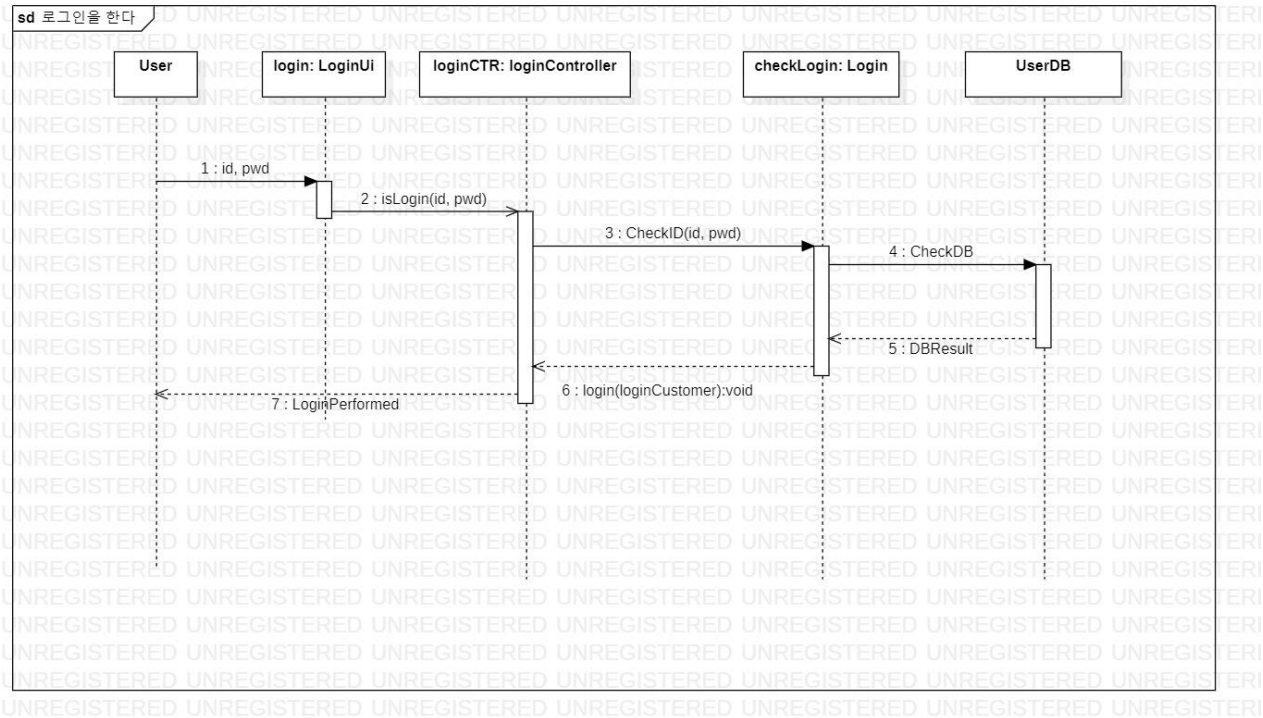
## 4. 시퀀스 다이어그램



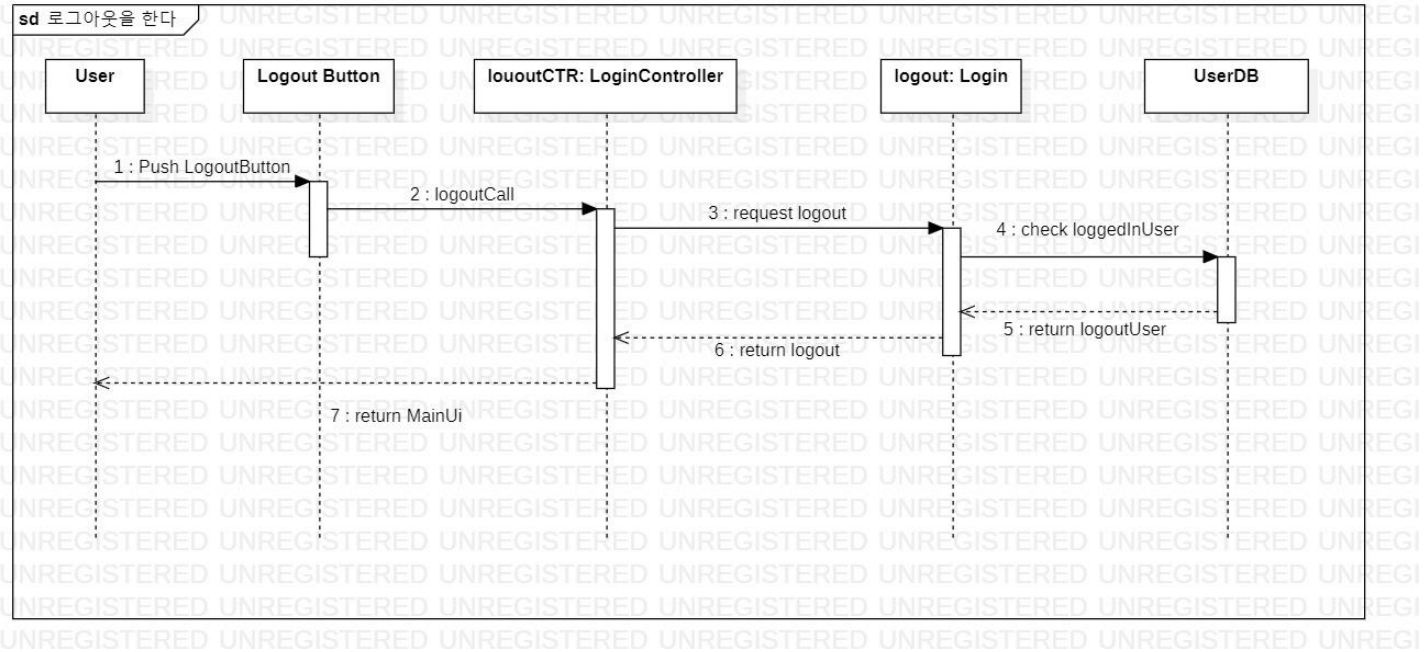
## 시퀀스 다이어그램 - 회원가입을 한다



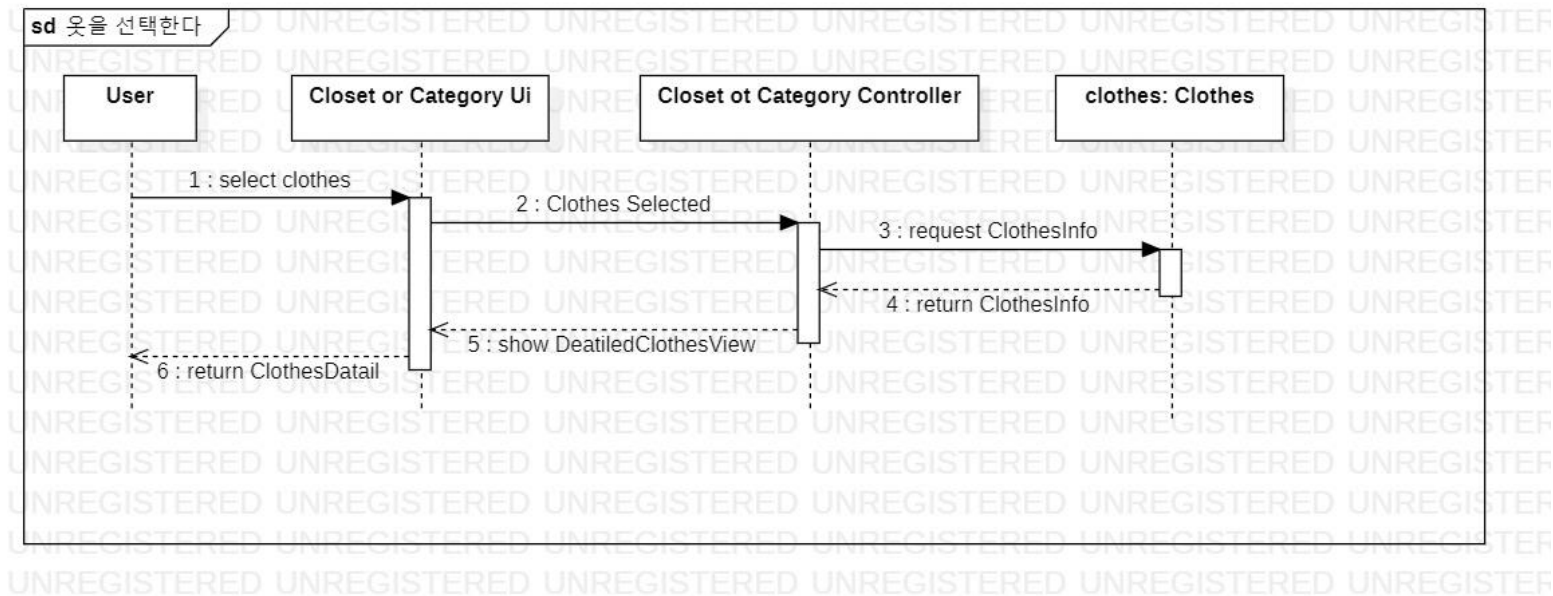
## 시퀀스 다이어그램 - 로그인을 한다



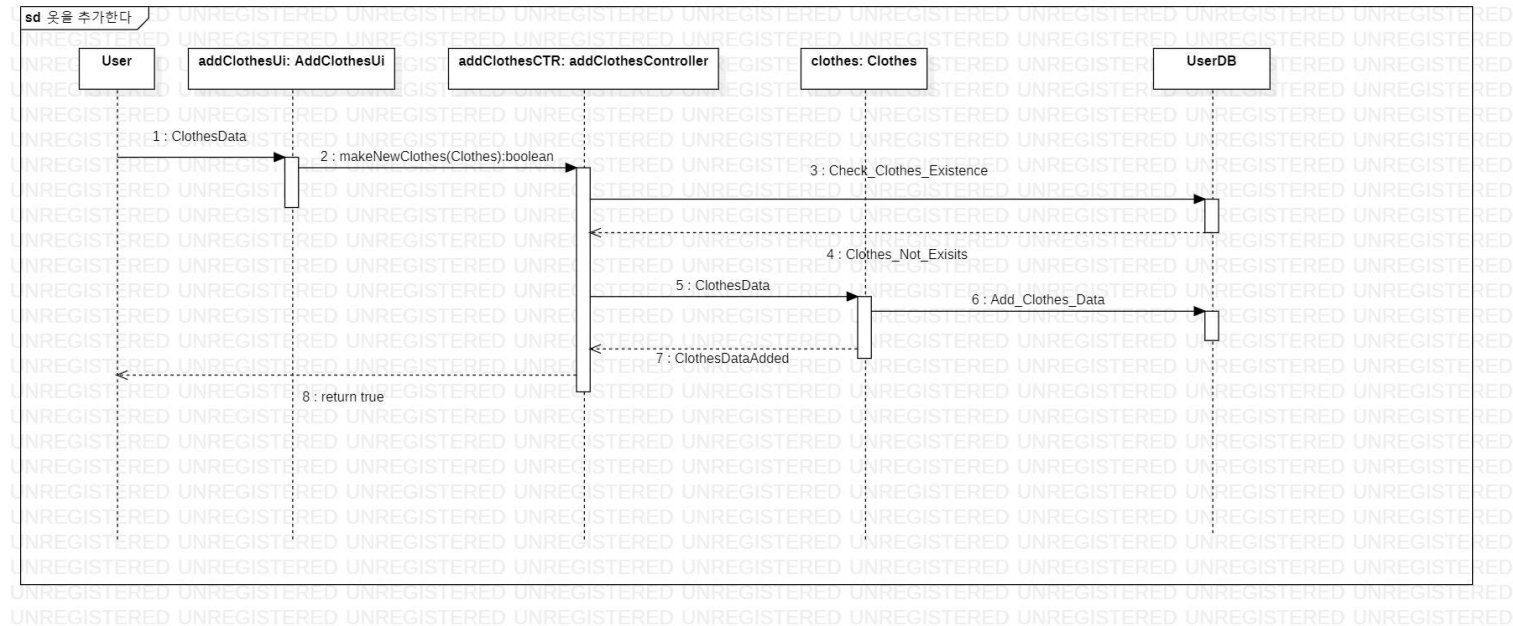
## 시퀀스 다이어그램 - 로그아웃을 한다



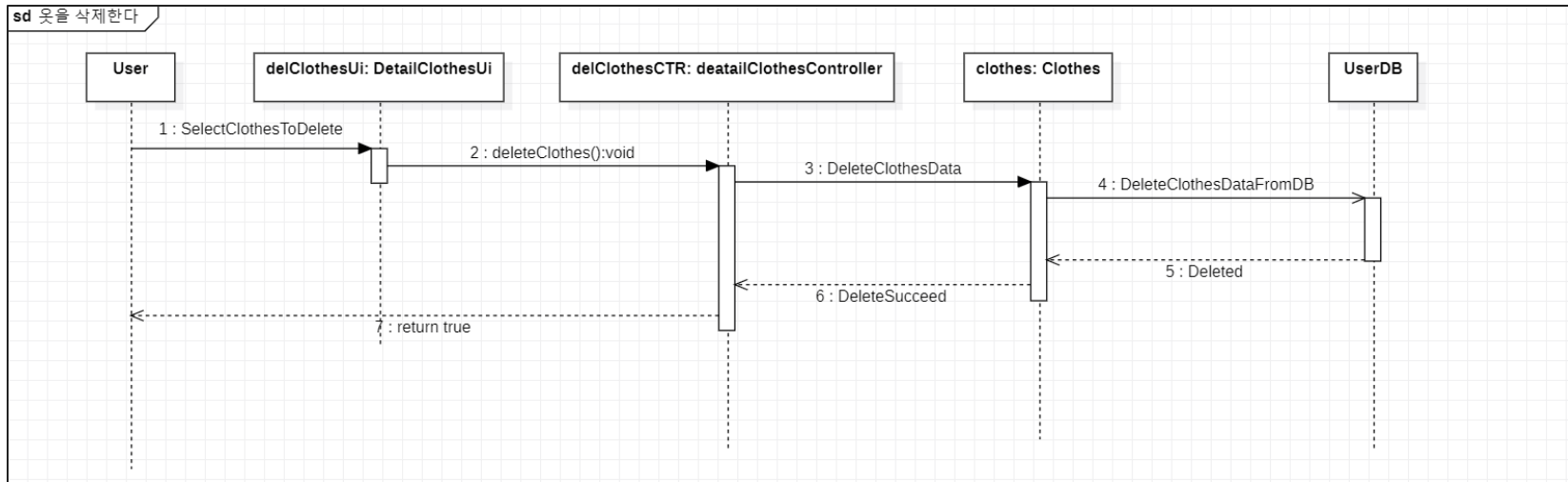
## 시퀀스 다이어그램 - 옷을 선택한다



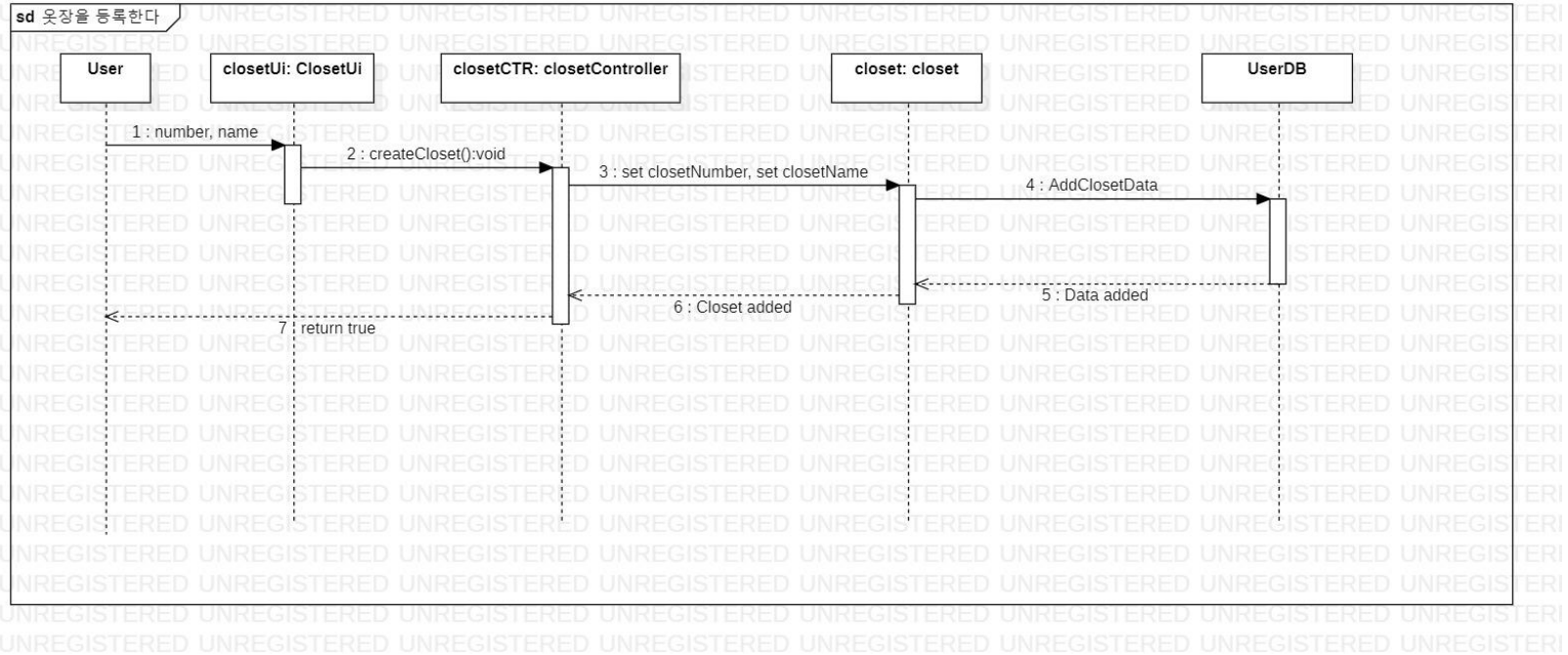
## 시퀀스 다이어그램 - 옷을 추가한다



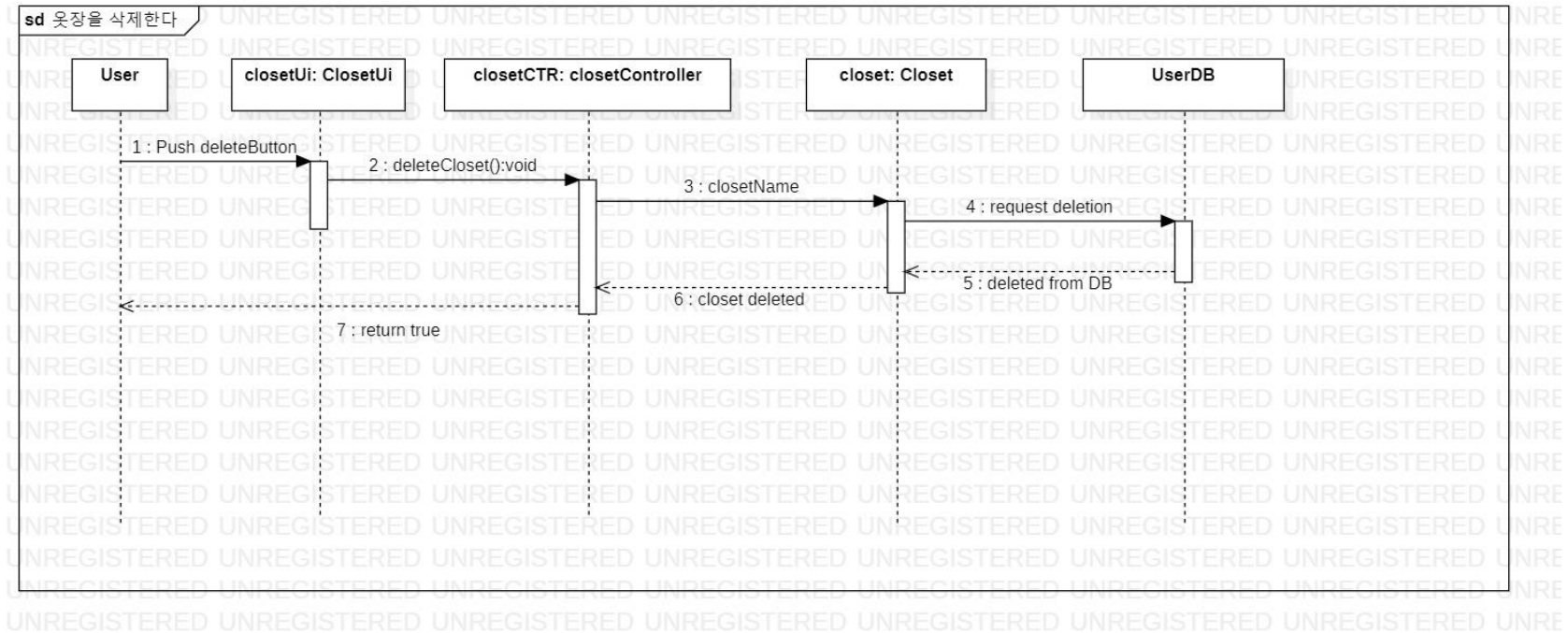
## 시퀀스 다이어그램 - 옷을 삭제한다



## 시퀀스 다이어그램 - 옷장을 등록한다

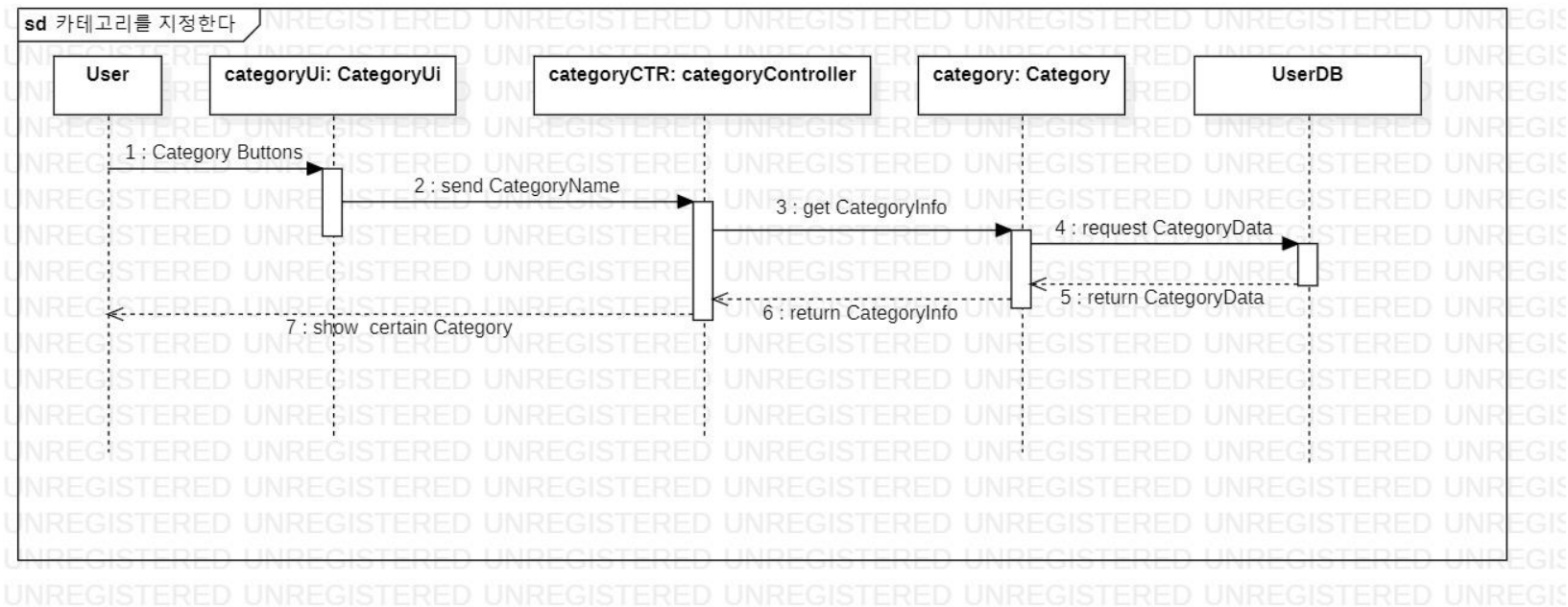


## 시퀀스 다이어그램 - 옷장을 삭제한다

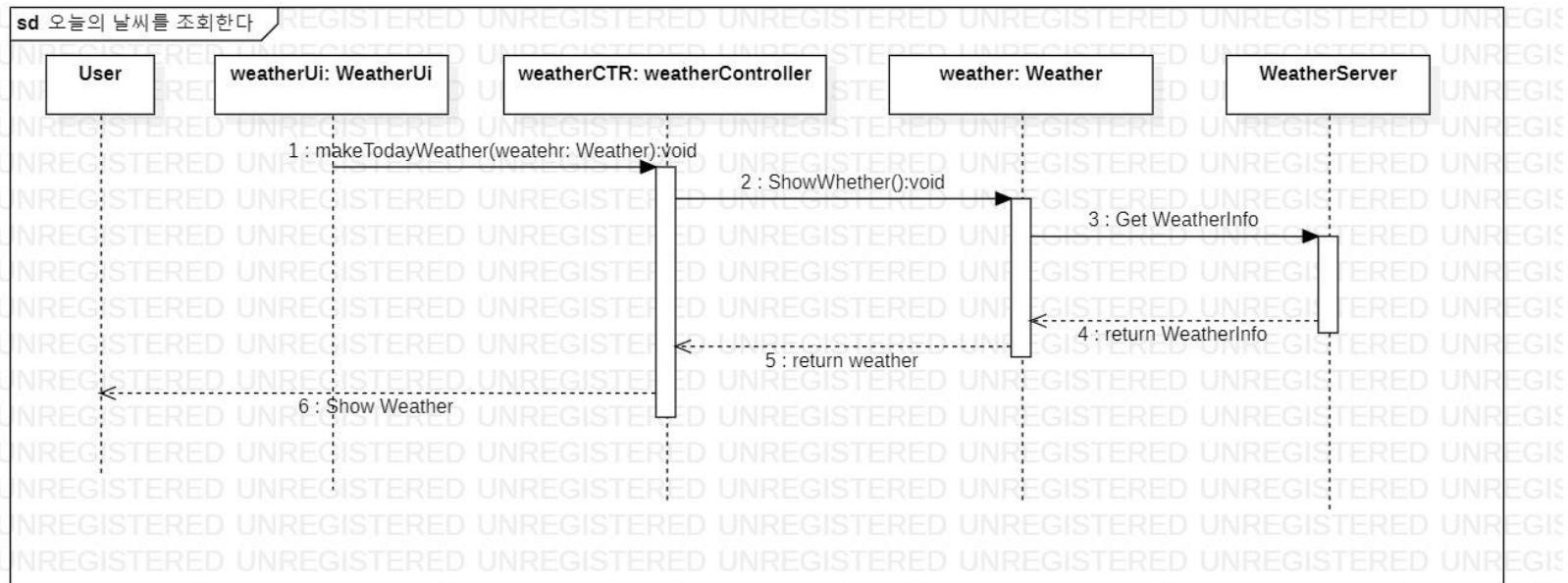




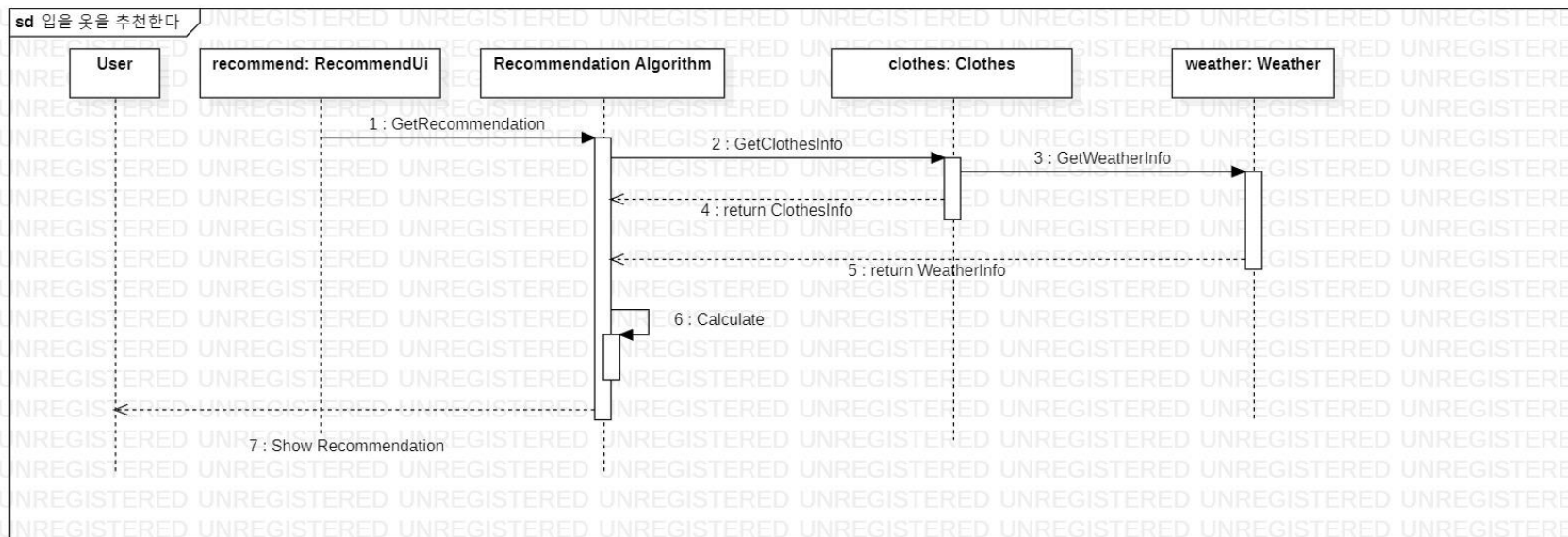
## 시퀀스 다이어그램 - 카테고리를 지정한다



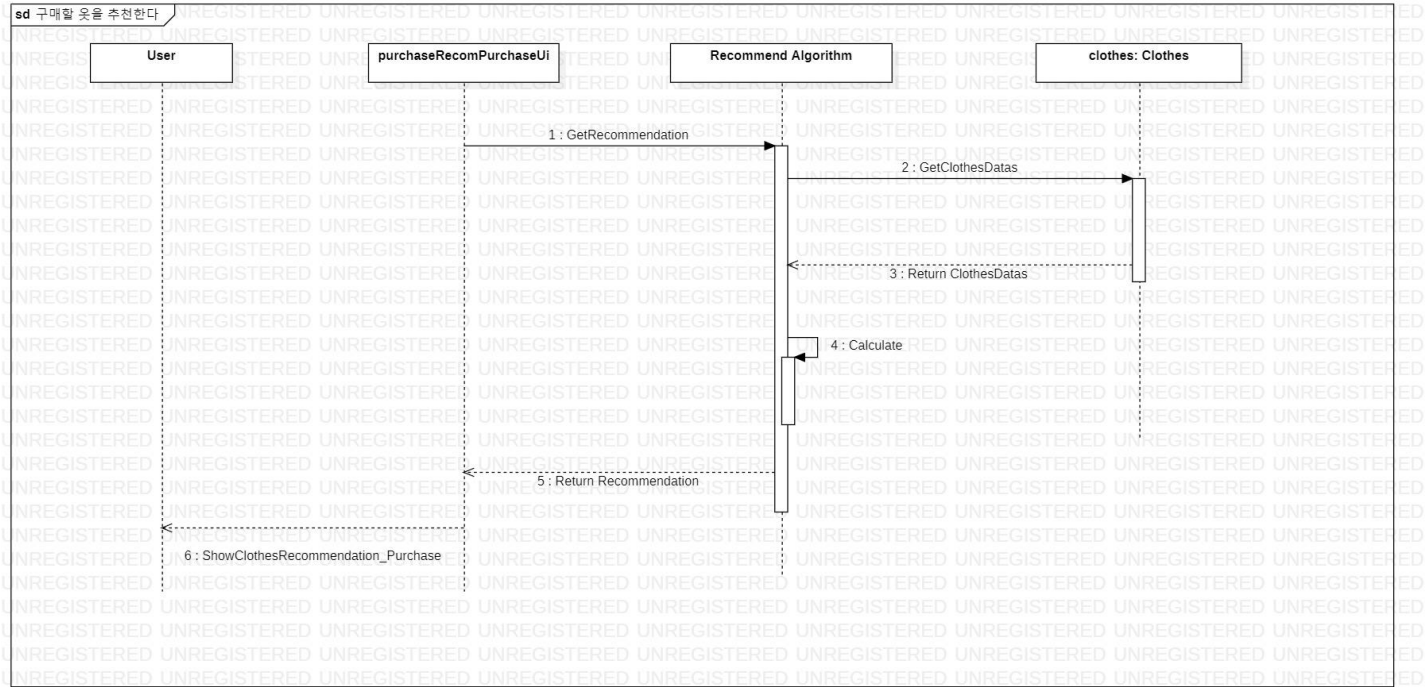
## 시퀀스 다이어그램 - 오늘의 날씨를 조회한다



## 시퀀스 다이어그램 - 입을 옷을 추천한다



## 시퀀스 다이어그램 - 구매할 옷을 추천한다



## 5. 어려웠던 점, 배운 점

## 어려웠던 점 & 개선점

### 프로젝트 중 어려웠던 점

- 새로운 언어를 습득하는 과정에서 예상보다 일정이 지연되었다.
- 일정 관리 미흡으로 처음 생각했던 기능들을 전부 다 구현하지는 못했다.

### 프로젝트 개선점

- 옷 추천 기능 제작 도중에 발표를 하게 되어 이 부분은 조금만 더 손보면 완성 가능할 것이라 생각한다.
- 현재 있는 기능 외에도 캘린더와 연계하여 더욱 다양한 기능들을 추가할 수 있을 것으로 보인다.

## 배운 점

### 1. 요구사항 명세서의 필요성

프로그램의 제작에 있어 막연하게 제작하기보다는 고객의 요구 사항을 명확하게 인지하고 목적에 맞추어 명세서를 작성하는 것이 큰 프로젝트일수록 필요하겠다고 느꼈다.

### 2. 소프트웨어 개발 모델 이용

소프트웨어 개발 생명 주기 모델을 간략하게나마 체험해보고 이를 통해 프로젝트를 진행하여 소프트웨어 개발 모델에 대해 이해하게 되었다.

### 3. 협업 톨 및 일정 관리의 중요성

프로젝트의 진행에 있어 예상 외의 변수가 많아 일정을 체계적으로 관리하고 공유하는 것이 중요하며, 협업 톨과 주기적인 기록들로 프로젝트를 수월하게 진행할 수 있는 방법을 배웠다.

## 팀원 역할, 기여 비율

