

블로그

삶의 모든 순간을 즐겨라!

Sequence Diagram

2007.07.04 13:21

2 Jordan 3

UML

댓글쓰기

1. Sequence Diagram 개요

※ UML은 기존에 제공하지 못했던 객체간 동적 상호 작용을 제공한다.

이를 Interaction이라 하는데 UML에는 Sequence Diagram과 Collaboration Diagram이 Interaction Diagram에 속한다.

① 정의 : 문제해결에 필요한 객체를 정의하고 객체간 동적 상호관계를 시간순서에 따라 정의.

② 작성목적

- * 객체간 동적 상호작용을 시간적 개념을 중시하여 모델링
- * 객체의 오퍼레이션과 속성을 상세히 정의
- * Usecase를 실현
- * 프로그래밍 사양 정의

③ 작성시기 : 유즈케이스 다이어그램 정의 후부터 프로그램 코딩전

④ 작성순서

- * 작성대상 선정 : 유즈케이스를 선정하고 유즈케이스 정의서 분석
- * 액터 위치시킴 : 액터는 좌측부터 위치..
- * 클래스 위치시킴 : 유즈케이스에 참여하는 클래스 위치.
- * 객체간 메시지정의 : 시간순서대로 객체간 메시지 정의
- * 객체 추가정의 : 요구사항 처리를 위해 필요한 객체가 정의되지 않았으면 추가 정의.

2. Sequence Diagram 구성요소

① Thing

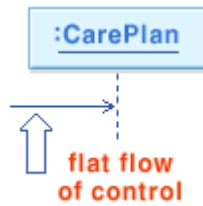
- * Actor : Usecase에서의 actor.
- * Object : 클래스의 인스턴스. 클래스 타입으로 선언된 변수형태로 존재.

홍길동:사원
사번=16009 이름=홍길동 급여=4000 성별=남 직위=과장

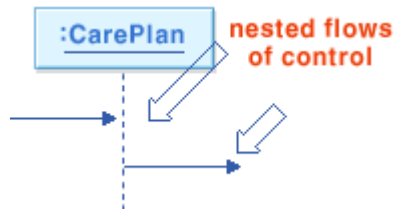
② Relationship

- * Message : 객체와 객체가 통신하는 유일한 수단

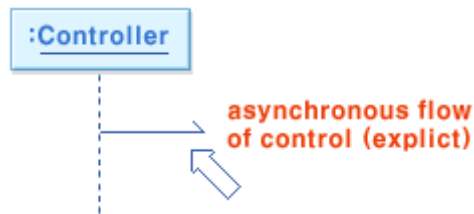
– Flat Flow of Control : 가장 일반적 메시지



– Nested Flow of Control : 메시지가 중첩 시 메시지가 모두 돌아와야 다음 처리 진행



– Asynchronous Flow of control : 메시지의 결과를 기다리지 않고 다음 처리 진행

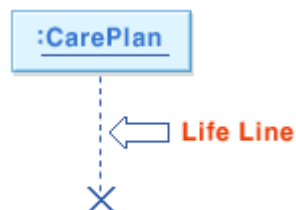


– Return Flow : 메시지를 처리한 결과. 필요한 경우에만 사용.



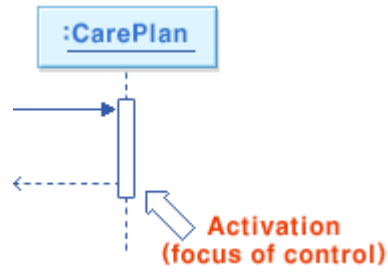
③ etc

* Life Line : 객체의 생존기간. 점선에 X표시가 객체가 소멸하는 시점.



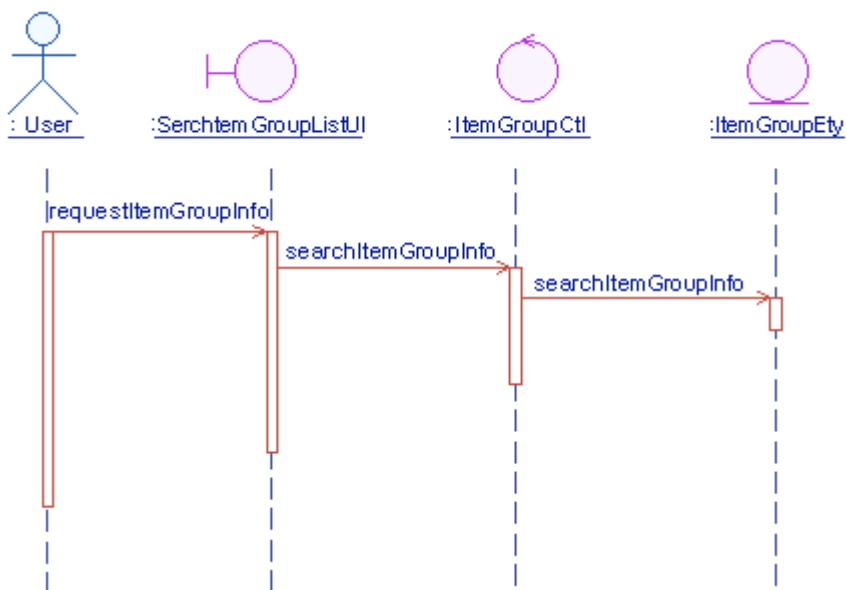
* Activation : 객체가 활성화 되어있는 기간. -점선표기

객체가 외부 메시지를 받고 보낸 메시지를 기다리는 기간.-좁고긴시각형

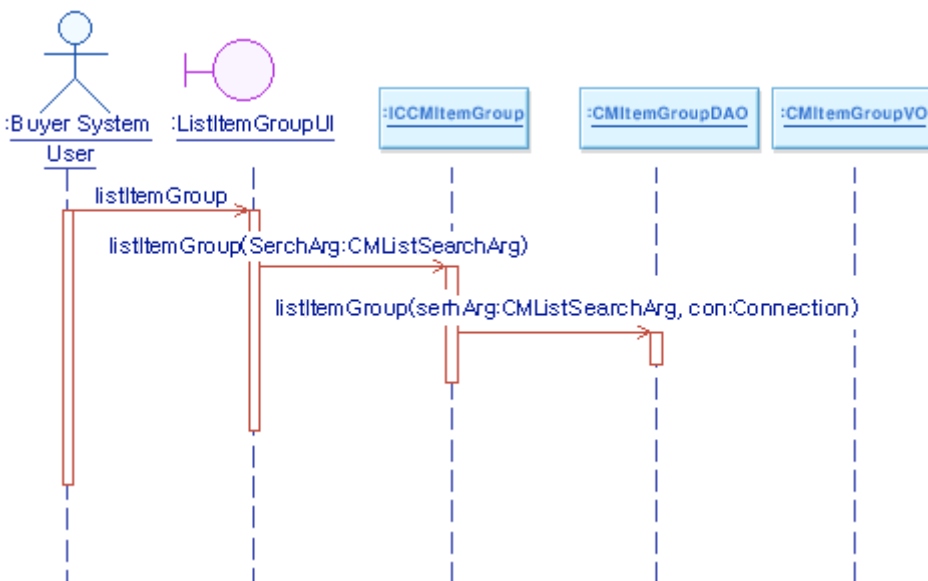


3. Sequence Diagram 사례

① 설계단계 Sequence Diagram



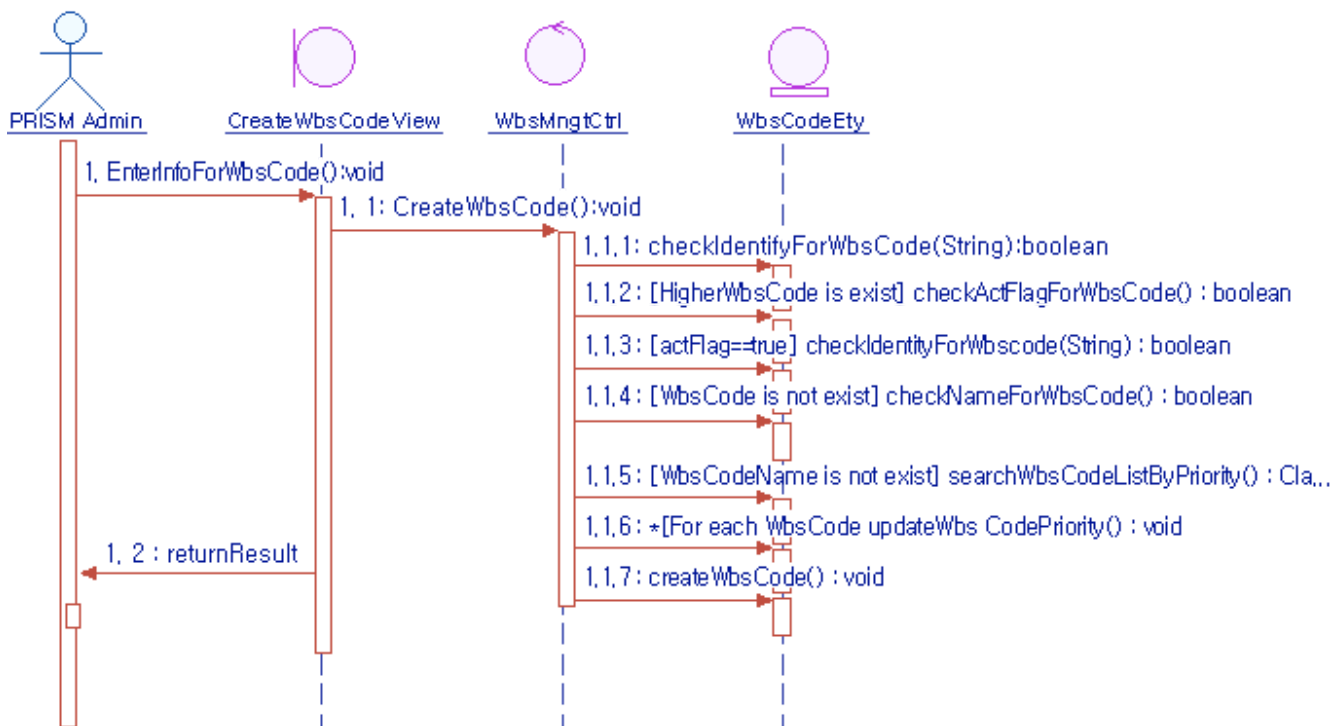
② 상세설계단계 Sequence Diagram



단계	설계단계	상세설계단계
Actor	User	Buyer System User
Boundary 객체	SearchItemGroupListUI	ListItemGroupUI
Control 객체	ItemGroupCtl	ICCMItemGroup(<<interface>>)
Entity 객체	ItemGroupEty	CMItemGroupDAO(<<DAO>>), CMItemGroupVO(<<VO>>)
	requestItemGroupInfo	listItemGroup
메시지	searchItemGroupInfo	listItemGroup (searchArg:CMListSearchArg)
	searchItemGroupInfo	listItemGroup (searchArg:CMListSearchArg, con:Connection)
ex)		

《uses》

WEB코드 목록조회



모듈. Use case Diagram, Use case/ 6. 유즈케이스 정의서의 사례/ 사례 2를 시퀀스

다이어그램으로 작성한 것입니다.

액터는 Prism Admin입니다. 설계단계에서 나온 것으로 객체들은

Boundary, Control,

Entity가 도출되었으며, 아이콘으로 표기되었습니다.

액터로부터 이벤트를 받는 역할은 Boundary인 'CreateWbsCodeView'

객체가 담당

하고, Control인 'WbsMngtCtrl' 객체가 비즈니스 로직을 담당하는 역할을

하고 있음을

알 수 있습니다.

메시지 1.1.6:*[For each Wbs Code]updateWbsCodePriority():void에서

몇 가지를

살펴 보겠습니다.

-1.1.6 : 메시지의 흐름 순서를 표기(필수는 아님)

-* : 메시지가 반복되는(looping) 것을 표기

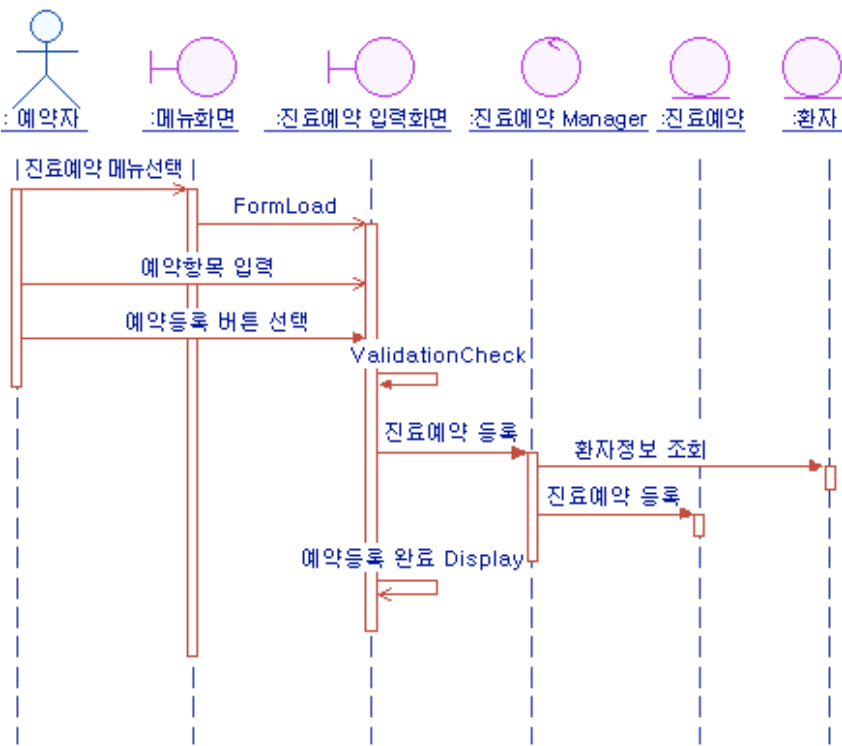
_ [For each Wbs Code] : 메시지가 실행되기 위한 조건(Guard Condition)

-updateWbsCodePriority() : 메시지 명

-void : Return 값

ex) 앞서 언급한 회계 시스템 시퀀스 다이어그램

■ 문제영역에 대한 시퀀스 다이어그램



댓글쓰기

--	--	--	--	--	--	--

11월 4주 이 블로그 인기글

- [Node JS] 이미지 업로드 및 리사이즈
- [펌] Eclipse에서 SVN 충돌시 해결 방법

윗 글 Collaboration Diagram
아랫글 Class Diagram

홈 | 방명록

맨위로