



# **소프트웨어 공학 (Software Engineering)**

## **요구 분석 (Requirement Analysis)**



## In this chapter ... (1/2)

요구 분석 (Requirement Analysis)

- 소프트웨어 개발의 실질적인 첫 단계
- 사용자의 요구에 대하여 이해하고 정리하는 작업
- 두 가지 작업
  - 현재의 상태를 파악하고 요구를 정의
  - 명세서(specification) 작성
- 요구의 변경은 파급 효과가 큼 (설계, 구현에 영향을 주기 때문)



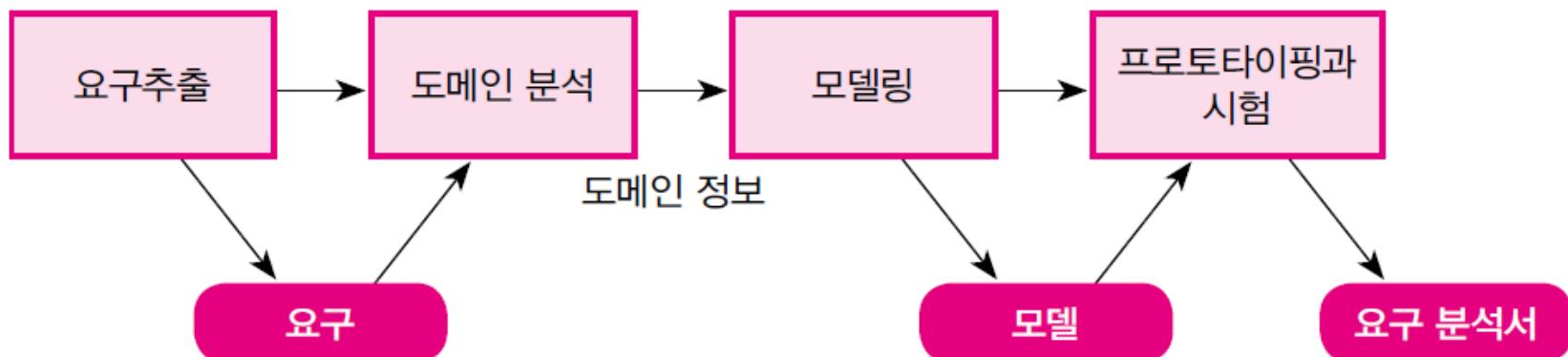
# In this chapter ... (2/2)

요구 분석 (Requirement Analysis)



## 요구 분석 과정

- 요구 추출 – 기능적인 요구와 기능 이외의 조건 추출
- 도메인 분석 – 요구에 대한 정보를 수집하고 배경을 분석
- 모델링 – 도메인 분석을 통해 얻은 자료를 개념화
- 프로토타이핑과 시험 – 분석된 기능적 요구의 타당성시험을 위한 프로토 타입 생성
- 문서화 검토 – 요구 분석서를 작성





# In this chapter ...

요구 분석 (Requirement Analysis)



## 4.1 요구(Requirements)



## 4.2 요구 추출



## 4.3 도메인 분석



## 4.4 사용 사례



## 4.5 요구 분석 명세서



## 4.6 요구 관리 도구



# 요구(Requirements)란?

요구 분석 (Requirement Analysis)



## 요구란

- 시스템이 제공해야 할 역량(capability)
- 외형적으로 나타내는 기능이나 성능



## 현금 인출기의 예

- 외형적 기능: 현금 인출, 잔액 조회, 계좌 이체, 현금 서비스, ...
- 기능 및 성능: 시간당 처리 능력, 반응 시간, 제약 조건  
(습도, 온도 등), ...



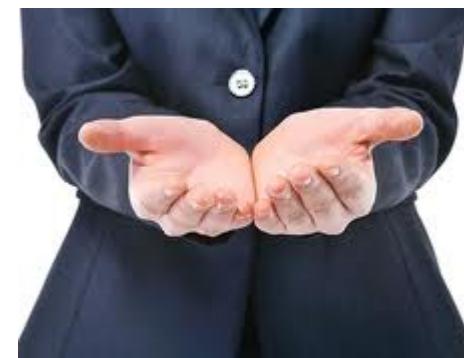


## ① 요구 분석: 무엇을 개발할 것인가를 결정하는 단계

- 잘못 되었을 때 바로 잡기 위한 비용이 큼
- Requirements Engineering (요구 분석가: Requirement Analyzer)

## ② 요구의 종류

- 기능적 요구 (functional requirements)
  - ✓ 소프트웨어의 종류, 사용자, 소프트웨어가 수행되는 시스템에 따라 다름
  - ✓ 시스템이 사용자를 위하여 무엇을 하는가를 거시적으로 기술
- 비기능적 요구 (non-functional requirements)
  - ✓ 성능: 응답 시간, 처리량, 신뢰도, 보안성, 운용 제약
  - ✓ 개발 비용: 투자한계





## 기능 자체와 관련된 질문

- 시스템이 무엇을 하는가?
- 시스템이 언제 그 일을 하는가?
- 시스템 운용될 때 여러 가지 다른 모드가 있는가?
- ...



사례는 [표 4.1] 참조



## 자료와 관련된 질문

- 입력, 출력이 무엇이며 어떤 형태를 갖는가?
- 얼마나 자주 자료를 받고 내보내는가?
- 시스템에 유입되는 자료의 양은 얼마나 되는가?
- 데이터는 어느 기간 동안 보관해야 하는가?
- ...





## 인터페이스(입출력)와 관련된 질문

- 다른 시스템에서 유입, 유출되는 입력은 무엇인가?
- 다른 시스템과 인터페이스하는 프로토콜의 종류는 무엇인가?
- 자료 전달에 사용되는 특정 미디어 형식이 있는가?
- ...



사례는 [표 4.1] 참조



## 사용자와 관련된 질문

- 누가 시스템을 사용할 것인가?
- 사용자가 여러 그룹인가? → 일반 사용자, 운영자, 관리자 등
- 각 사용자 그룹의 컴퓨터 사용 경험은?
- 각 사용자 그룹에 따라 필요한 교육은?
- ...

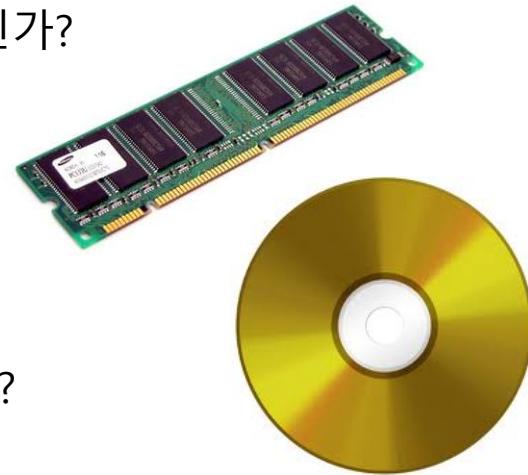


사용자 계정



## 자원과 관련된 질문

- 시스템 구축 및 유지보수에 필요한 자원, 인력은 무엇인가?
- 개발자가 갖추어야 할 기본 요구사항이 있는가?
- 어떤 하드웨어를 사용할 것인가?
- 시스템이 차지할 수 있는 공간은 어느 정도인가?
- 동작 환경(전력, 온도, 습도)에 대한 제약은 없는가?
- 개발된 시스템의 디스크, 메모리 공간의 제약은 없는가?
- ...



## 성능과 관련된 질문

- 시스템의 속도, 반응 시간, 처리율  
(TPS: transactions per second, CPS: calls per second, MIPS, ...)
- 시스템에 의하여 처리되는 자료의 크기
- ...



사례는 [표 4.1] 참조



## 보안과 관련된 질문

- 자료와 시스템에 대한 접근이 통제되어야 하는가?
- 사용자들 사이에 타인에 대한 데이터 및 프로그램 접근 방지
- 시스템의 백업 기간 및 책임자
- 화재, 홍수, 도난 등의 재난을 대비한 방지책은?
- 물리적 보안 대책? (→ 세콤?)
- ...



## 품질과 관련된 질문

- 신뢰성, 가용성, 유지보수성, 보안 등의 품질 특성에 대한 요구
- 시스템이 가동되는 평균 시간
- 시스템의 작업이 중단된 후 다시 복구될 때까지 허용되는 시간
- 설계 변경이 얼마나 용이한가?
- 얼마나 쉽게 위치나 플랫폼 변경이 가능한가? (→ Portability)
- ...



사례는 [표 4.1] 참조



- ① 개발 팀이 응용 도메인에 대하여 충분히 알지 못함
- ② 고객과 사용자가 소프트웨어가 무엇을 하는지 또한 요구를 어떻게 표현할 지 모름
- ③ 공통 배경지식 부족으로, 개발 팀과 사용자 사이의 대화 장벽이 생김
- ④ 소프트웨어 요구에 대한 명세와 구현이 분리될 수 없어 정확히 명시하기 어려움
- ⑤ 요구 추출 작업을 관리자, 사용자, 개발자 모두 과소평가하는 경우가 많음
- ⑥ 비기능적 요구를 파악하고 이해하지 못함
- ⑦ 요구가 계속해서 변경됨



## In this chapter ...

요구 분석 (Requirement Analysis)

 4.1 요구(Requirements)

 4.2 요구 추출

 4.3 도메인 분석

 4.4 사용 사례

 4.5 요구 분석 명세서

 4.6 요구 관리 도구



# 요구 분석가의 역할?

요구 분석 (Requirement Analysis)

- ④ 요구 분석가는 기자의 취재와 유사한 작업을 수행한다.
  - 사용자를 만나 면담을 수행한다.
  - 실제 업무에 참관하거나, 업무를 직접 수행해 보기도 한다.
  - 분석가는 비판적이고 객관적이어야 한다.  
→ 기자가 한쪽 편의 이야기만을 들어서는 되지 않는다.
  - 분석가는 기자와 마찬가지로 6하 원칙에 따라서 정보를 수집한다. → 다음 페이지 참조
  
- ⑤ 기자가 취재를 종합하여 기사를 쓰듯이, 요구 분석가는 다양한 요구 추출을 통하여 요구 분석서를 작성해야 한다.





▣ 요구 추출이란? 사용자가 무엇을 원하는지 결정을 내리는 작업

▣ 요구 추출을 위한 세 가지 단계

- 응용에 대한 정보 출처 파악 – 누가 관련되어 있고 시스템의 범위가 어디까지인가?
- 응용에 대한 정보 취합 – 고객의 발표, 설문, 브레인스토밍 등 정보를 모음
- 요구와 제한 사항의 정의 – 기능, 성능, 제한/제약 사항을 결정





- 1) 절대적으로 필요한 요구 → 전필?
- 2) 요망되나 꼭 필요한 것은 아닌 요구 → 학점 못 채운 경우의 전선?
- 3) 요구로 볼 수는 있으나 제외될 수 있는 요구 → 학점 채운 경우의 전선?

## ❸ 신용카드 대금청구서 예제

- 거래 내역, 총 금액 계산, 납부 기일 명시 → 1)에 해당
- 일시불, 할부 등의 구매 형태 표시 → 2)에 해당
- 취소된 내역과 승인된 내역을 다른 색으로 표시 → 3)에 해당



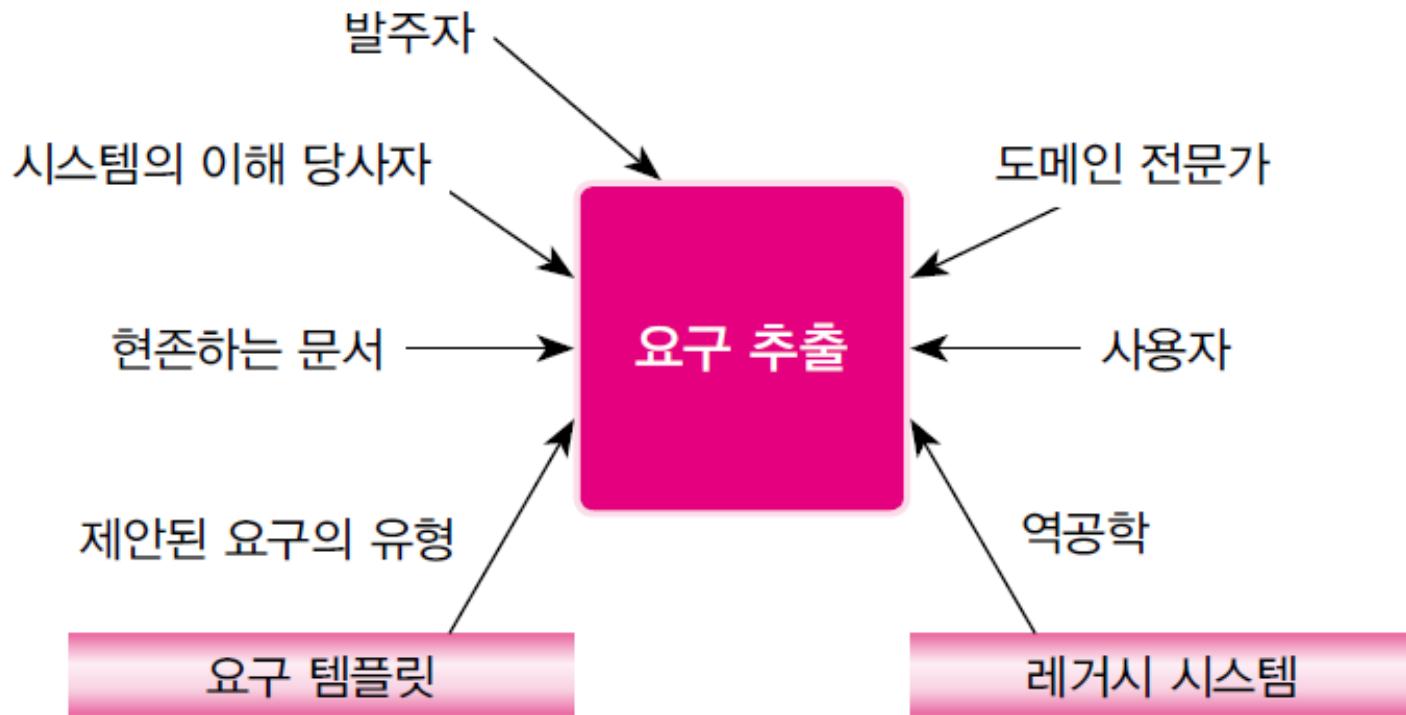


## 정보 출처의 유형

- 고객
- 도메인 전문가 – 비즈니스 도메인을 지원하는 시스템을 구축하기 위하여 필요한 사람  
(예: 회계 시스템을 구축하기 위하여 회계사가 필요)
- 이해당사자(stakeholder) – 시스템 운영으로 인하여 영향 받는 사람  
(예: 창고 운영 시스템의 경우, 주문, 결재 직원이 상품 이동 직원에 비해 관심이 많음)
- 사용자 – 시스템을 직접 사용하는 사람
- 역공학 – 레거시 시스템이나 현재 시스템의 문서로 부터 요구 추출



### ④ 요구 추출을 위한 정보의 출처





- 여러 가지 방법을 동원하여 사용자/소스로부터 요구를 추출함
- 각 방법마다 추출하는 요구의 범위와 효율이 다를 수 있음
- 요구 추출 방법 (정보 수집 방법)
  - 고객의 발표
  - 문헌 조사
  - 업무 절차와 양식 조사
  - 설문
  - 인터뷰
  - 브레인스토밍 회의
  - 프로토타이핑
  - 사용자 스토리





▣ 개발팀이 구축하는 시스템에 대하여 초기에 개념을 잡을 수 있음

▣ 효과적인 가이드 라인

- 고객 업무를 잘 알고 있는 운영 책임자나 관리자가 발표
- 발표하기 전 개발 팀원이 필요한 정보가 있는지 검토
- 의심이 가는 부분을 질문하여 명확히 할 것
- 구현과 관련된 토의는 배제
- 발표 내용의 복사본을 팀원과 공유
- 2시간 이상의 발표회는 지양





- 유사한 프로젝트를 조사

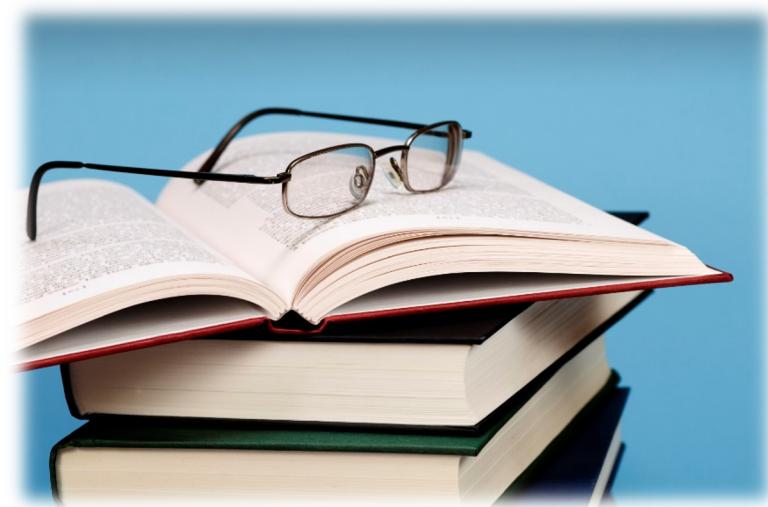
→ 현재 개발할 시스템에 대한 통찰 제공

- 업무 문서나 양식을 조사

→ 현재의 업무나 시스템 정보에 대해 깊은 이해 가능

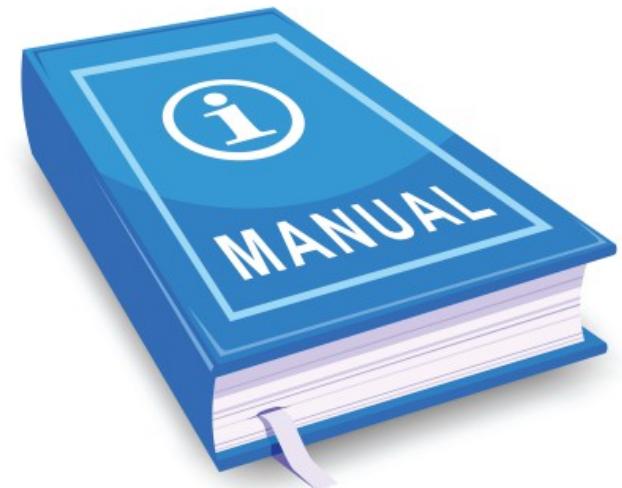
- 산업 및 기업 표준 조사

- 관련 정부 정책/규제 조사





- ④ 업무 관련 문서, 절차, 양식 운영 매뉴얼 조사
- ④ 내부 표준 조사
- ④ 정부, 산업, 기업의 특수 정책이나 규정 조사





- 관리자나 사용자와 같은 이해 당사자를 대상
- 이해 당사자들이 의사결정 과정에 포함

## 무기명 설문

- 이해 당사자들의 관심과 내부정보, 개선 의견 도출
- 감추어진 정보를 끌어내기 쉬움

## 유의사항

- 질문은 간단하고 중요한 이슈에 집중
- 적절하고 잘 기술된 질문

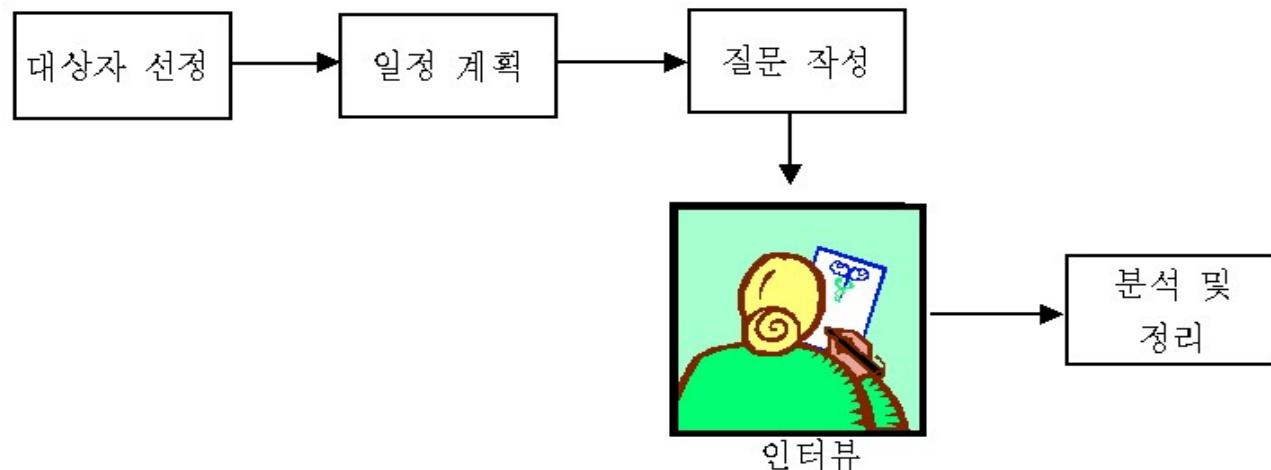




# 인터뷰 (1/2)

요구 분석 (Requirement Analysis)

- 미리 잘 계획하면, 짧은 시간에 많은 정보를 얻을 수 있음
- 가능하면 많은 당사자와 인터뷰
- 관련자 이외의 다른 사람도 인터뷰
  - 예: 경쟁 제품의 사용자, 마케팅 담당자 등
- 인터뷰 시간은 예상보다 여유 있게 시간을 할애하여야 함





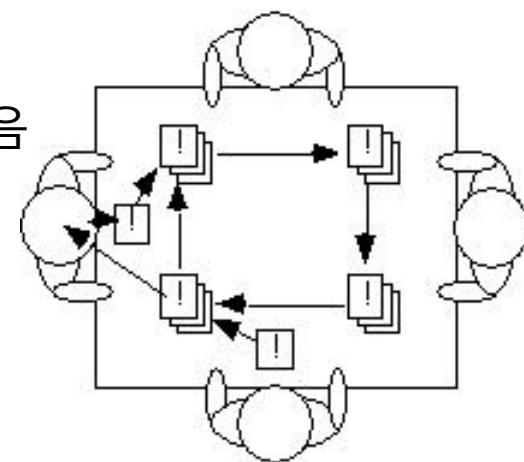
④ 여러 번의 인터뷰를 준비하고, 충분한 질문을 준비함

⑤ 포함시켜야 할 질문 유형

- 구체적인 설명 요구  
예) 최대, 최소, 예외 규칙, 예상 되는 변동 등
- 관련자의 시스템에 대한 미래 비전  
→ 어떤 융통성을 가져야 하는지 제안될 수도 있음
- 다른 아이디어가 있는지 질문 → 구현 대안을 찾을 수도 있음
- 최소한의 허용 가능한 솔루션이 무엇인지 질문  
→ 모든 기능을 아는 것도 중요하지만, 우선 순위를 파악하는 것도 중요함
- 다른 정보원이 없는지 질의 → 다른 인터뷰 대상자를 찾을 수도 있음
- 다이어그램 작성 요구 → 정보 흐름 등을 파악할 수 있음



- 아이디어를 낼 목적으로 여러 명으로부터 정보를 얻기 위한 회의
- 훈련된 요원이 주재하여 브레인스토밍 과정을 정돈(관리)해야 함
- 토론보다는 아이디어를 쏟아놓는 회의, 익명성 보장
- JAD(Joint Application Development) – 집중 브레인스토밍 세션
  - 개발자가 고객(혹은 사용자)과 격리된 장소에서 사나흘 회의함
  - 하나의 그룹으로 작업하거나, 특정 문제에 대해 여러 개의 작은 그룹으로 작업 할 수도 있음
  - 참여자가 다른 업무로 중단 받아서는 안되므로, 학회나 수련회처럼 격리된 장소에서 수행할 필요가 있음





1. 관련자 모두가 참여하는 회의 소집 (5명 ~ 30명 수준)
2. 경험 많은 사람을 회의 주재자로 선정
3. 테이블에 참석자를 배석시키고 종이 준비
4. 토론을 유도할 질문을 정함 (다음 페이지 참조)
5. 질문에 대하여 답을 종이에 적되 한 장에 하나의 아이디어만 적은 후, 참석자에게 돌려 봄
6. 5번 단계를 5~15분간 반복  
(아이디어가 없을 때까지 반복)
7. (원래 아이디어 제안자가 그 내용을) 간단히 설명
8. 모든 아이디어를 칠판에 적은 후 우선순위를 정하기 위하여 투표를 할 수도 있음





- ④ 참석자가 Yes, No 식의 대답이 아닌 짧은 문장의 대답이 나와야 함
- ④ 유도 질문은 회의 소집자, 주재자, 간단한 토론 등을 통해 결정함
- ④ 유도 질문의 예
  - 어떤 기능이 이 시스템에서 중요한가?
  - 미래에 어떤 자료의 출처가 예상되는가?
  - 시스템에서 어떤 출력이 생성되어야 하는가?
  - 도메인 모델에서 어떤 클래스가 필요한가?
  - 인터뷰 질문에서 무엇을 물을 것인가?
  - 고려하지 않을 이슈는 무엇인가?
  - 이 프로젝트의 리스크는 무엇인가?

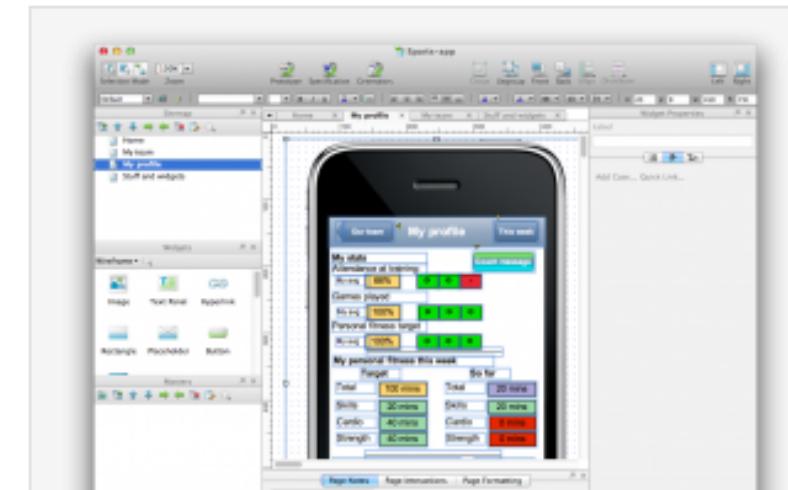


## 프로토타입이란?

최종 시스템의 예상 기능 중 일부를 빠르게 구현한 프로그램

## 가장 단순한 형태: paper prototype

- 시스템이 수행될 때 무엇이 일어날지 설명한 그림을 순서대로 그린 것
- 다수 개발자가 다른 버전을 병행 제작하여, 가장 좋은 것을 선택함



Prototype of a sports iPhone app using Axure



# 프로토타이핑 (2/4)

요구 분석 (Requirement Analysis)

Paper Prototype for leggmasonfunds.com landing page

|  |  |                      |                        |                   |                        |                    |
|--|--|----------------------|------------------------|-------------------|------------------------|--------------------|
| LEGG<br>MASON  | Inst.<br>Asset<br>Mgmt.  | Wealth<br>Management | Legg<br>Mason<br>Trust | Private<br>Client | Legg<br>Mason<br>Funds | Capital<br>Markets |
| Client Log-In  | Legg Mason Funds Banner  |                      |                        |                   |                        |                    |
| Open An Account  |  |                      |                        |                   |                        |                    |
| <u>Search</u> ►►   | tagline ~~~~~  |                      |                        |                   |                        |                    |
| <b>Our Funds</b>   | <b>Global Brief</b><br>content, content ~~~~~<br><u>more ➤ Aug x, xxxx</u>   |                      |                        |                   |                        |                    |
| <b>Knowledge</b>   | <b>Context and Perspective</b><br>content, content ~~~~~<br><u>more ➤ Aug x, xxxx</u>  |                      |                        |                   |                        |                    |
|  | <b>Monthly Market Review</b><br>content, content ~~~~~<br><u>more ➤ Aug x, xxxx</u>  |                      |                        |                   |                        |                    |
| Funds Investor Services<br>1-800-522-5544<br>8:00 AM - 5:30 PM<br>(ET) MON - FRI | <b>What's New!</b><br>> Press Release<br>> Another Press Release<br>> 401k Season<br>> Comments by Bill Miller.<br><br>footer, disclaimer, privacy policy,<br>~~~~~. |                      |                        |                   |                        |                    |

If the user pointed to the "Fund Finder" drop down menu, the full menu (below) would be presented.

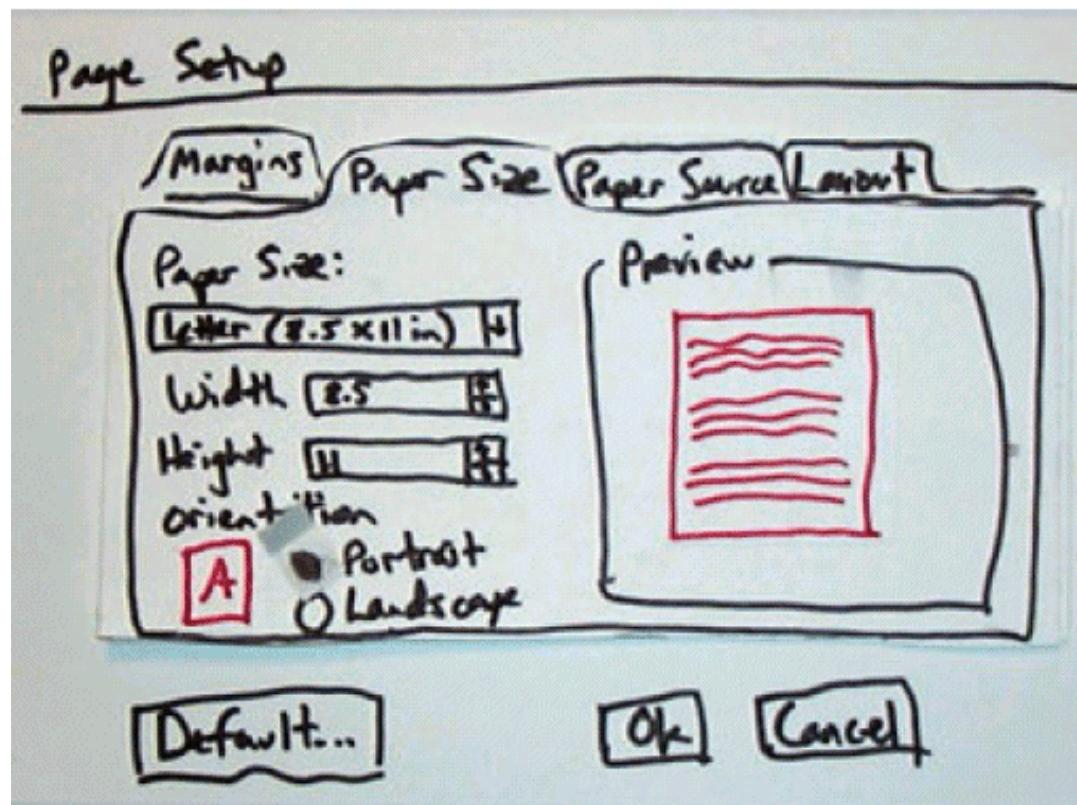
**Fund Finder**

|                        |
|------------------------|
| <b>Select a Fund</b> ▾ |
| American Leading       |
| Balanced Trust         |
| Cash Reserve           |
| Classic Value Fund     |
| Emerging Markets       |
| Europe Fund            |
| Financial Services     |
| Focus Trust            |
| Global Income          |
| High Yield             |
| Int'l Equity           |
| Opportunity Trust      |
| Value Trust            |



## 가장 흔한 형태: 모의 사용자 인터페이스

- 프로토타이핑 전용 언어(rapid prototyping language)로 작성
- 컴퓨팅, 데이터베이스 접근, 다른 시스템과의 상호작용은 불가능





# 프로토타이핑 (4/4)

요구 분석 (Requirement Analysis)

SoCQ GUI

File Schema View Query

Show all Hide all Reduce all 100%

Outline

- Temperature
- Employee
  - Schema
    - Name : STRING
    - Address : STRING
    - Messenger : SERVICE
    - Message : STRING VIRTUAL
    - Sent : INTEGER VIRTUAL
  - BP: sendMessage
- Surveillance

Brice.socq

**Temperature**

- Area : INTEGER
- Temperature : REAL

**Surveillance**

- Area : INTEGER
- Manager : STRING
- Thresold : REAL
- AlertMessage : STRING

**Employee**

- Name : STRING
- Address : STRING
- Messenger : SERVICE
- Message : STRING VIRTUAL
- Sent : INTEGER VIRTUAL

**sendMessage**

**Messenger** : SERVICE

- Address : STRING
- Message : STRING VIRTUAL
- Sent : INTEGER VIRTUAL

Properties

| Property | Value   |
|----------|---------|
| Name     | Sent    |
| Type     | INTEGER |
| Virtual  | True    |



## ❶ 애자일 방법에서 요구 취합을 위해 많이 사용되는 방법

### ❷ 사용자 스토리의 특징

- 사용자와 개발 팀이 함께 작성
- 사용자들이 시스템에 바라는 역량을 간단히 기술한 것
- 내부 사람이 만들기 때문에 효과적

### ❸ 병원 진료 기록 시스템의 사용자 스토리 사례

- 간호사가 환자를 위한 진료 기록을 만들어야 한다.
- 간호사가 환자의 생일이나 이름으로 진료 기록을 찾아야 한다.
- 간호사가 환자의 검사 기록을 추정하여야 한다.
- 의사가 환자와의 대화를 문서로 남겨야 한다.
- 의사가 환자의 정보를 찾기 위해 대화를 검색하여야 한다.
- ...



## In this chapter ...

요구 분석 (Requirement Analysis)

 4.1 요구(Requirements)

 4.2 요구 추출

 4.3 도메인 분석

 4.4 사용 사례

 4.5 요구 분석 명세서

 4.6 요구 관리 도구

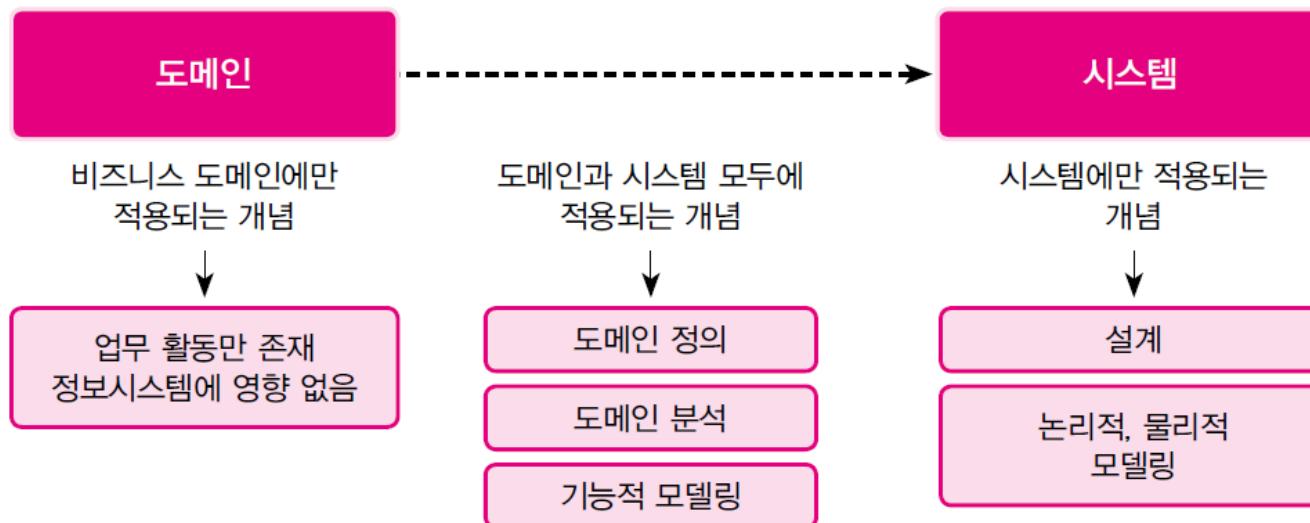


## ▣ 도메인이란?

- 요구의 배경을 말한다.
- 문제 자체도 중요하나 문제가 어디에 놓여있는가, 즉 배경을 이해하여야 한다.

## ▣ 설계 모델링에 필요한 여러 개념과 비즈니스 룰을 파악

## ▣ 응용 분야에 존재하는 개념을 잘 정의하고 분석하여 시스템에 존재하는 개념으로 정립하는 단계





## ① 업무 작업 영역을 파악하고 범위를 규정하는 작업

- 어떠한 정보 시스템을 구축할지의 업무 범위를 결정하는 작업
- 정보 시스템을 구축하는데 필요한 개념적인 프레임워크를 제공함

## ② 넓은 범위의 개념을 좁은 범위의 지식들로 체계화하는 작업

## ③ 사례: 의료 정보 시스템을 위한 도메인 정의

| 도메인     | 범위   |
|---------|--|
| 환자 관리   | <ul style="list-style-type: none"><li>• 진료예약 일정</li><li>• 접수</li><li>• 처방</li><li>• 의료비 청구</li></ul> |
| 조제      | <ul style="list-style-type: none"><li>• 약 보관</li><li>• 약 조달</li></ul>                                |
| 의료 및 검사 | <ul style="list-style-type: none"><li>• 의료 기기 구매</li><li>• 의료 기기 보관</li><li>• 의료 기기 유지보수</li></ul>   |
| 입원병동    | <ul style="list-style-type: none"><li>• 청소</li><li>• 세탁</li><li>• 식사</li></ul>                       |



- ▣ 도메인 분석은 요구의 배경을 알아보는 것임
- ▣ 도메인의 배경 파악을 위해 필요한 세 가지 단계
  - **도메인 개념** 찾기
  - **도메인 사전** 작성
  - **비즈니스 규칙** 정리





## 도메인의 목적, 구조, 동작을 구성하는 객체, 프로세스, 사람, 규칙 같은 것들을 의미함

- 프로젝트 초기에 정확한 개념을 세우기 위해 용어를 바르게 정의하여야 함
- 하나의 도메인에도 개념이 넘쳐나며, 넘쳐나는 개념의 바다에서 시스템에 관련된 개념을 찾는 것이 쉽지 않음

## 도메인 개념 발견을 위해 주의할 사항

- 요구의 핵심을 발견해야 한다.
- 요구가 해결될 것 같은 문제를 발견해야 한다.
- 문제의 요소를 발견한다.  
(예: 의료 시스템에서 예약, 접수, 입원, 기록 퇴원, 진료비 청구 등)
- 관련된 도메인의 개념을 발견한다.  
(직접 관련된 영역만 아니라 연관되는 다른 도메인도 고려)



### 의료 정보 시스템의 도메인 개념은 다음과 같은 것이 파악되어야 함

- 예약: 병원 직업이 환자를 위해 적절한 절차를 거쳐 환자의 정보를 입력한다.
- 접수: 진료 서비스를 받기 전에 환자의 프로파일을 갱신하고, ID를 발급하고, 수납을 위한 정보를 입력한다.
- 진료와 검사: 의사, 간호사, 검사원이 적절한 의료 서비스를 제공한다.
- 입원: 의사 결정에 따라 입원을 진행하며, 입원 후에는 상태 체크, 투약 등의 의료 서비스를 제공한다.
- 퇴원: 의료 서비스가 끝나면 환자를 퇴원시킨다.
- 수납: 환자에게 진료비를 청구하고, 보험공단에도 진료비를 청구한다.



## ▣ 도메인 개념을 조직화한 결과물

- 각 항목은 용어가 사용될 때는 언제든 같은 의미로 통하게 하는 간결한 정의
- 쉽게 말해, 도메인 분석 과정의 용어들을 정의하는 사전임

## ▣ 요구, 인터뷰, 매뉴얼로부터 추출하여 정의

## ▣ 도메인 정의에서 표현된 문장, 어절, 제목에 초점을 두고 개념 추출

## ▣ 의료 정보 시스템에서 도메인 개념 추출

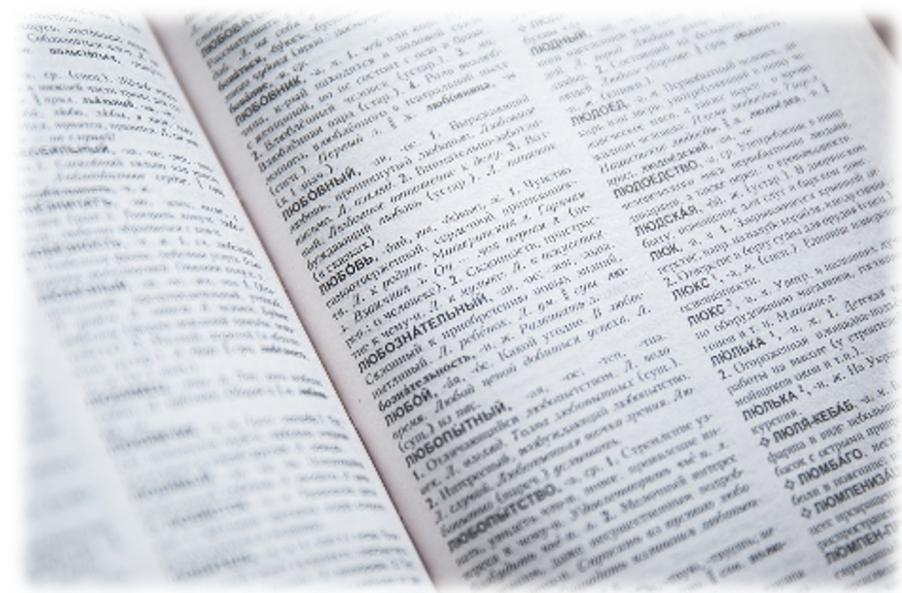
- (정의) 진료와 검사: 의료 서비스의 성격에 따라, 의사, 간호사, 검사원이 환자에게 적절한 예약된 의료 서비스를 제공한다.
- 도메인 개념: 의료 서비스, 의사, 간호사, 검사원, 환자, 예약
- 결국, 도메인 사전에는 도출된 개념에 대한 설명이 있어야 함



## ▣ 사전 양식은 간단한 표로 구성

## ▣ 사전에 포함되는 세 가지 항목

- 명칭: 개념을 식별하기 위한 고유한 단어
- 타입: 객체 지향 관점에서, 프로세스, 객체, 역할, 기능 등으로 구분
- 설명: 명칭에 대한 설명이며, 비즈니스 규칙 혹은 공식으로 설명할 수 있음



 의료 시스템의 도메인 사전 예제

| 명칭       | 타입   | 설명  |
|----------|------|---|
| 예약       | 프로세스 | 환자가 <u>의료 서비스</u> 를 받는 일정. <u>예약 담당 직원</u> 에 의하여 처리됨  |
| 예약       | 객체   | 환자에게 <u>의료 서비스</u> 를 제공하기 위하여 약속된 날짜와 시간  |
| 예약 담당 직원 | 역할   | 환자를 위하여 <u>예약</u> 한다.   |
| 의료 서비스   | 객체   | 의료진이 <u>환자</u> 에게 제공하는 의학적 서비스의 총칭: 진료, 처방, 조제, 검사 등  |
| 의료 서비스   | 기능   | 의료진이 <u>환자</u> 에게 의학적 서비스를 제공하는 행위  |
| ...      | ...  | ...   |
| 예약       | 프로세스 | 의료 서비스를 제공하기 이전에 이루어짐. 프로세스는 새로 오는 환자 또는 기존 환자의 개인 정보와 보험 정보를 수집한다. 이 과정에 병원 <u>진료 카드</u> 가 발급된다. <u>예약 담당 직원</u> 이 수행한다. |
| 예약 담당 직원 | 역할   | 예약을 실시한다.   |



## 업무에서 지키기로 한 규정

- 기업이 운영되는 자세한 정책, 규정, 절차, 가이드라인, 표준의 집합
- 사용자에게 요구해도 준비된 전체 목록을 받기 어려움

## 비즈니스 규칙 종류

- 사실(fact): 개념이 무엇인지 설명
- 추론(inference): 다른 사실로 부터 얻은 사실  
(예: 20세 이하이면 주류를 판매할 수 없다)
- 액션 구동자(action enabler): 조건이 일치되면 액션이 수행  
(예: CD 재고가 없으면, 당일 배송 옵션을 끈다)
- 제약(constraints): 시스템이나 외부 요소가 수행할 또는 수행하지 않을 제약을 가하는 규칙
- 계산(computation): 공식이나 알고리즘



## 의료 정보 시스템의 비즈니스 규칙 예제

| ID                             | 정의   | 타입 | 소스                  |
|--------------------------------|--|----|---------------------|
| 001                            | 환자는 질병으로부터 고통받는 사람이며 의사나, 응급실, 다른 의료기관에서 진료를 의뢰받는다.  | 사실 | 도메인<br>사전           |
| 002                            | 19세 이하의 환자는 보호자나 응급 센터와 동행하여야 한다.  | 제약 | 병원 정책               |
| 예약할 때는 환자에 대한 다음 정보가 보관되어야 한다. |  |    |                     |
| 003                            | ID<br>데이터<br>004-01 이름<br>004-02 주소<br>004-03 전화번호1<br>004-04 전화번호2<br>004-05 주민등록번호<br>004-06 의료카드 ID | 제약 | 도메인<br>전문가와의<br>인터뷰 |
| 005                            | 접수 직원은 환자의 개인 정보를 입력, 수정하고 필요하면 의료 카드를 발행한다.   | 사실 | 도메인<br>사전           |
| 006                            | 의료비 청구가 30일 이내에 수납되지 않으면 연체료를 물린다.   | 유추 | 병원 정책               |



## In this chapter ...

요구 분석 (Requirement Analysis)

4.1 요구(Requirements)

4.2 요구 추출

4.3 도메인 분석

4.4 사용 사례

4.5 요구 분석 명세서

4.6 요구 관리 도구



# 사용 사례 (Use Case)

요구 분석 (Requirement Analysis)

- ▣ 도메인 분석과 모델링 사이의 관문이 되는 작업
- ▣ 도메인 분석의 결과를 액터, 사용 사례, 관계들로 구성된 시스템 명세로 매핑하는 작업

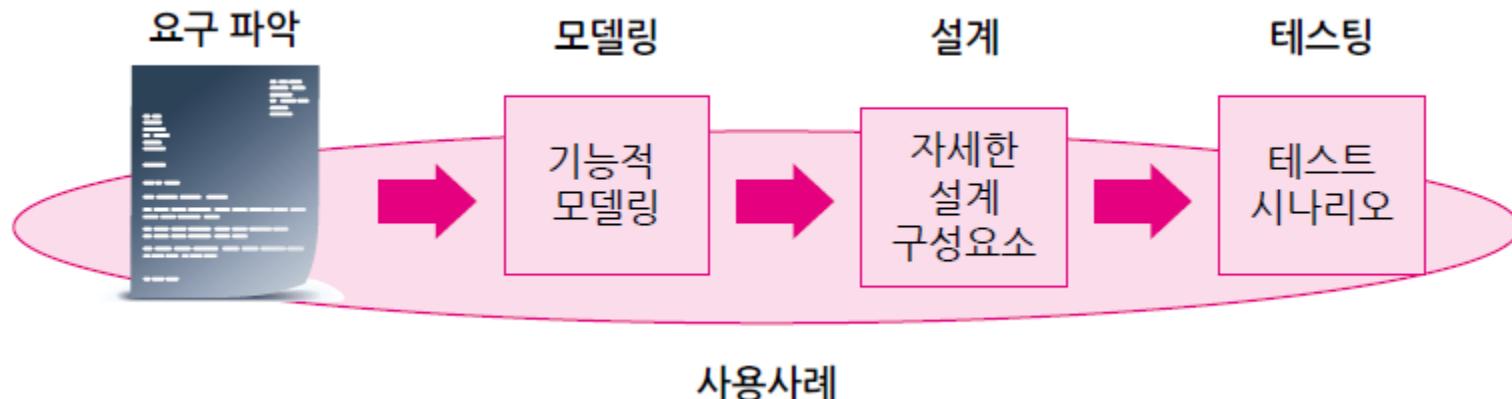




▣ (정의) 사용 사례는 시스템의 사용자에게 서비스를 제공하기 위한 상호 작용의 단위이다.

▣ 사용 사례가 중요한 이유

- 사용자 또는 외부 시스템이나 기타 요소들이 시스템과 상호작용 하는 디아일로그를 모델링
- 시스템 설계자/테스트 프로그래머들이 의사 교환하는데 유용
- 소프트웨어 개발자와 이해 당사자 간의 계약





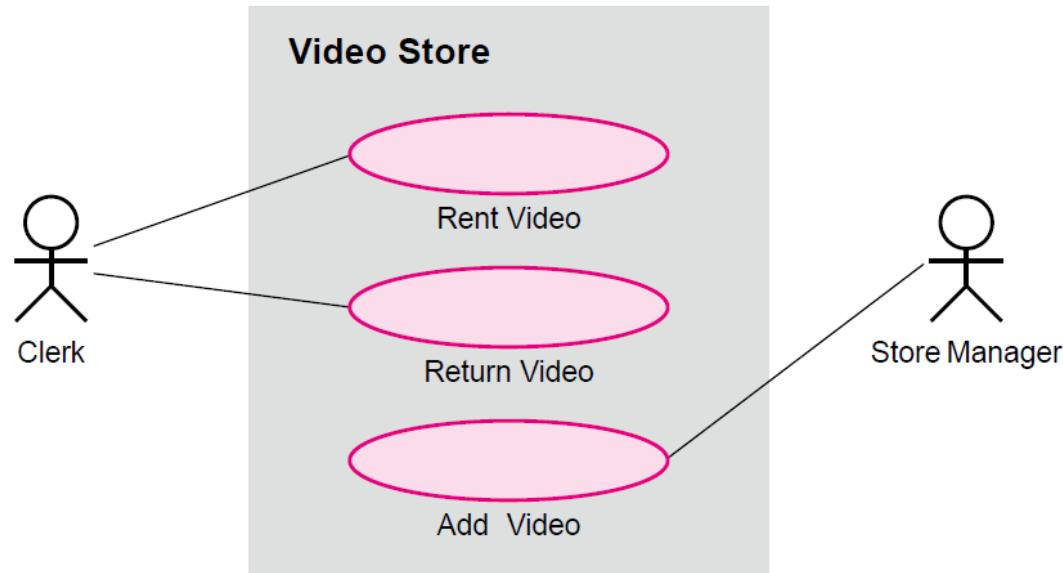
### 사용 사례를 작성하는 과정

- 액터 찾기
- 사용 사례 찾기
- 사용 사례 사이의 관계 찾기





- 시스템의 기능을 나타내기 위하여 사용자의 요구를 추출하고 분석하는데 사용됨
- 사용 사례 다이어그램의 구성
  - 액터(actor): 시스템과 상호작용을 하는 것(사용자, 시스템)
  - 사용 사례(use case): 시스템의 기능





## 액터

- 시스템과 상호 작용하는 외부 엔티티
- 구별되는 이름과 설명이 필요

## 사용 사례

- 액터의 입장에서 본 시스템의 동작
- 액터가 볼 수 있는 결과를 내는 이벤트의 집합
- 다른 사용 사례를 가동시킬 수 있음



## 액터는 시스템과 작용하는 외부 엔티티임

- 액터는 사람이 될 수도 있고 외부 시스템이 될 수도 있음
- 액터는 역할을 추상화한 것으로 꼭 사람과 결부시킬 필요는 없음



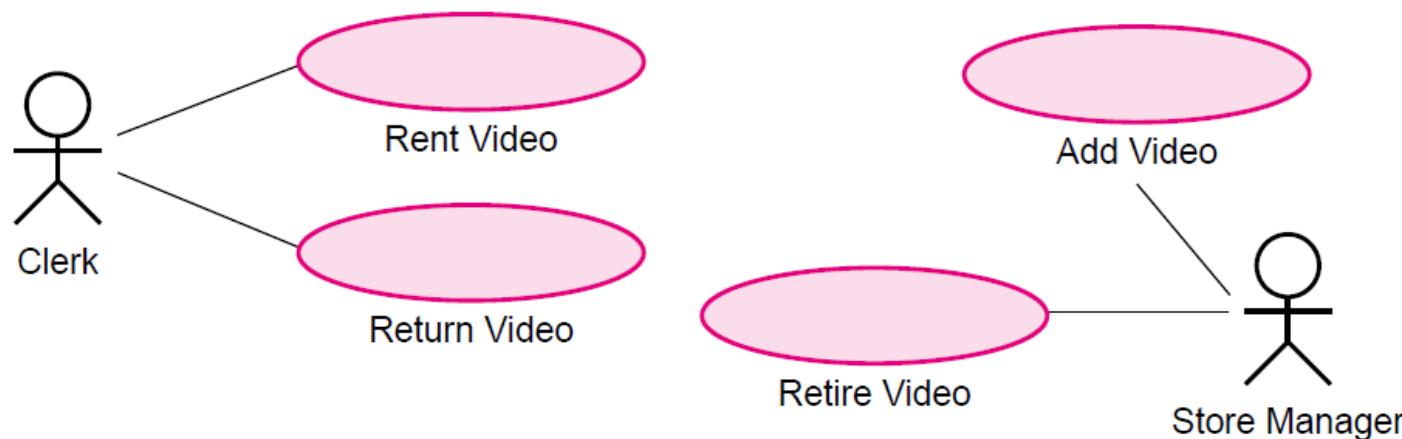
## 액터를 찾기 위한 질문

- 어떤 사용자 그룹이 작업을 수행하기 위하여 시스템의 지원을 받는가?
- 어떤 사용자 그룹이 시스템의 주요기능을 사용하는가?
- 어떤 사용자 그룹이 유지 보수와 관리 등의 부수적 기능을 사용하는가?
- 시스템이 다른 외부 하드웨어나 소프트웨어 시스템과 동작하는가?



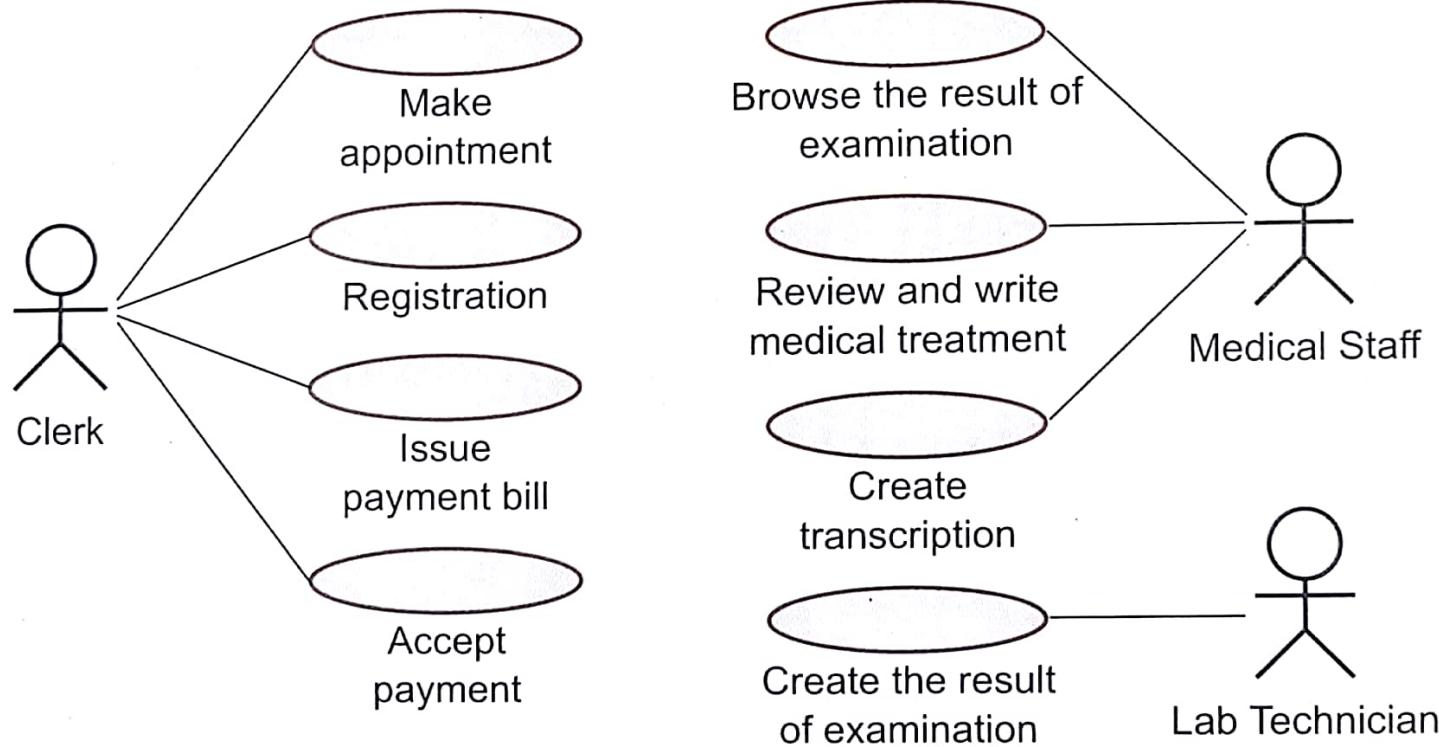
## 비디오 대여점의 액터

- Clerk: 점원으로 비디오 대여 업무를 수행
- Store Manager: 신규 비디오 추가, 오래된 비디오 제거 등 업무 수행



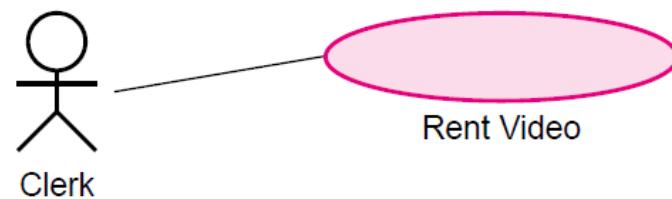


## 의료 정보 시스템의 액터





- 사용 사례는 컴퓨터 시스템을 사용할 때 사용자가 무엇을 경험/수행하는지를 글로 표현한 것임
- 사용 사례는 여러 개의 시나리오를 뮤어서 형성됨
  - 정상적인 흐름 뿐 아니라 오류, 예외 케이스 시나리오도 고려
  - 시나리오는 사용 사례의 인스턴스라 볼 수 있음
- 두 개의 시나리오로 구성된 사용 사례의 예제



## 시나리오 #1

대여 메뉴 선택  
전화번호 키(1234) 입력  
고객 대여 기록 화면에 디스플레이  
대여 테이프, DVD 스캔  
대여기록 추가 및 디스플레이

## 시나리오 #2

대여 메뉴 선택  
전화번호 키(4321) 입력  
전화번호 키를 가진 고객을 찾지 못하고 오류 메시지를 화면에 디스플레이



## 情景構成

- 개발자와 사용자가 함께 작성
- 현재의 응용 도메인에 대하여 기술한 여러 문서를 이용(지침서, 매뉴얼 등)

## 情景構成을 위한 질문

- 시스템이 어떤 작업을 수행하기를 액터가 원하는가?
- 액터가 원하는 정보는 무엇인가?
- 누가 데이터를 생성하는가? 데이터는 조작, 삭제될 수 있는가? 이런 작업이 누구에 의하여 행해지는가?
- 액터가 시스템에 정보를 알리는데 필요한 것은? 얼마나 자주 또 언제 이런 작업이 일어나는가?
- 액터가 시스템으로부터 정보를 알아내는데 필요한 이벤트는? 이런 사건의 빈도는?



## 완성된 사용 사례의 예

|            |   |
|------------|---|
| 사용 사례 이름 : | RentVideo   |
| 참여 액터 :    | User에 의하여 구동됨   |
| 시작 조건 :    | 스캐너를 이용   |
| 사건의 흐름 :   | <ol style="list-style-type: none"><li>User가 터미널에서 “비디오 대여” 기능을 활성화 시킨다. 시스템이 고객 ID 입력 양식을 화면에 제시하여 반응한다.</li><li>점원인 User가 비디오를 대여하려는 고객에게 전화번호의 끝 네 자리를 물어 입력한다.</li><li>입력한 네 자리로 찾은 이름들을 화면에 보여주고 맞는 것을 선택하도록 한다.</li><li>연체료가 있다면 화면에 출력하고 없으면 스캐너를 이용하여 대여하려는 비디오 ID를 입력한다.</li><li>비디오 ID를 이용하여 비디오 정보를 찾아 화면에 출력하고 대여중인 비디오 데이터베이스에 기록한다. 대여할 비디오 가 더 있으면 반복한다.</li><li>User가 대여료를 받고 테이프를 건네준다.</li></ol> |
| 종료 조건 :    |   |



### 의료 정보 시스템의 사용 사례

- 교재의 예제 4.8
- 표 4.9와 표 4.10 참조



관계를 이용하여 모형의 복잡도를 줄이고 이해도를 높임

관계 종류

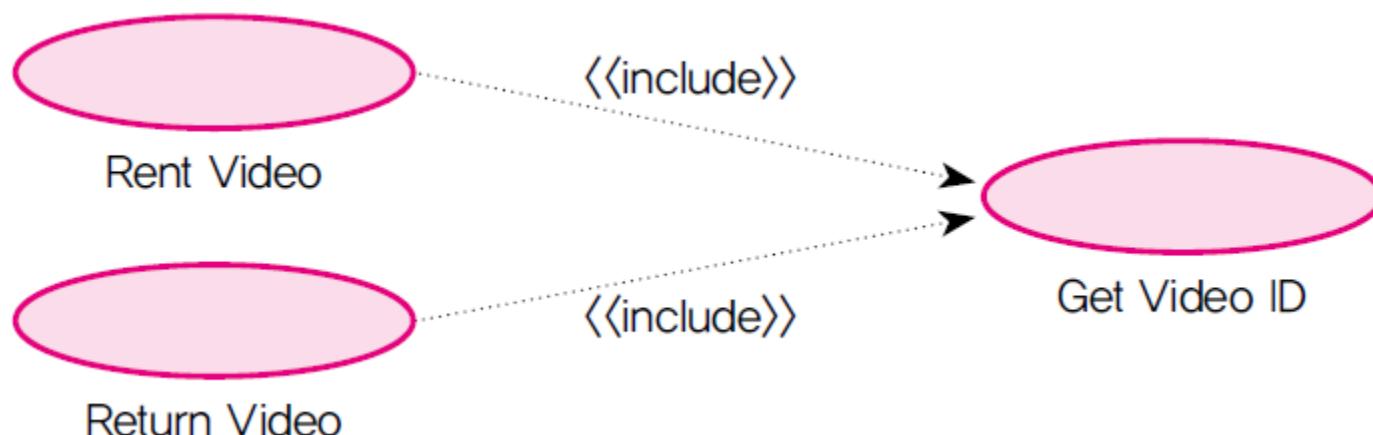
- 포함(include): 사용 사례 사이의 공통점을 찾아 중복을 줄임
- 확장(extend): 정상 이벤트와 예외 이벤트를 분리함

**include**  
**extend**



## 포함 관계

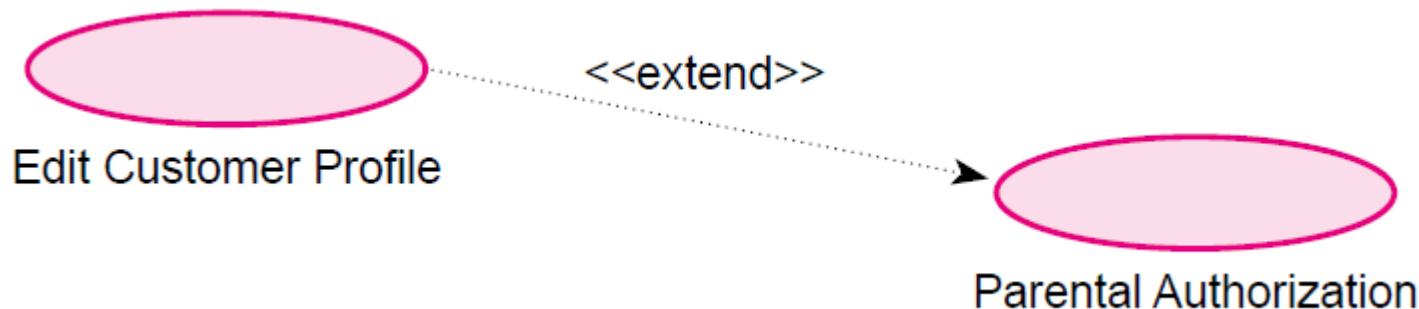
- 사용 사례 사이의 중복을 제거함
- 어떤 사용 사례(들)가 다른 사용 사례를 포함하는 관계
- 공통된 동작을 빼어낼 수 있음
- 예제: 비디오 대여와 반환 사례는 ID를 스캔하는 사례를 포함한다.





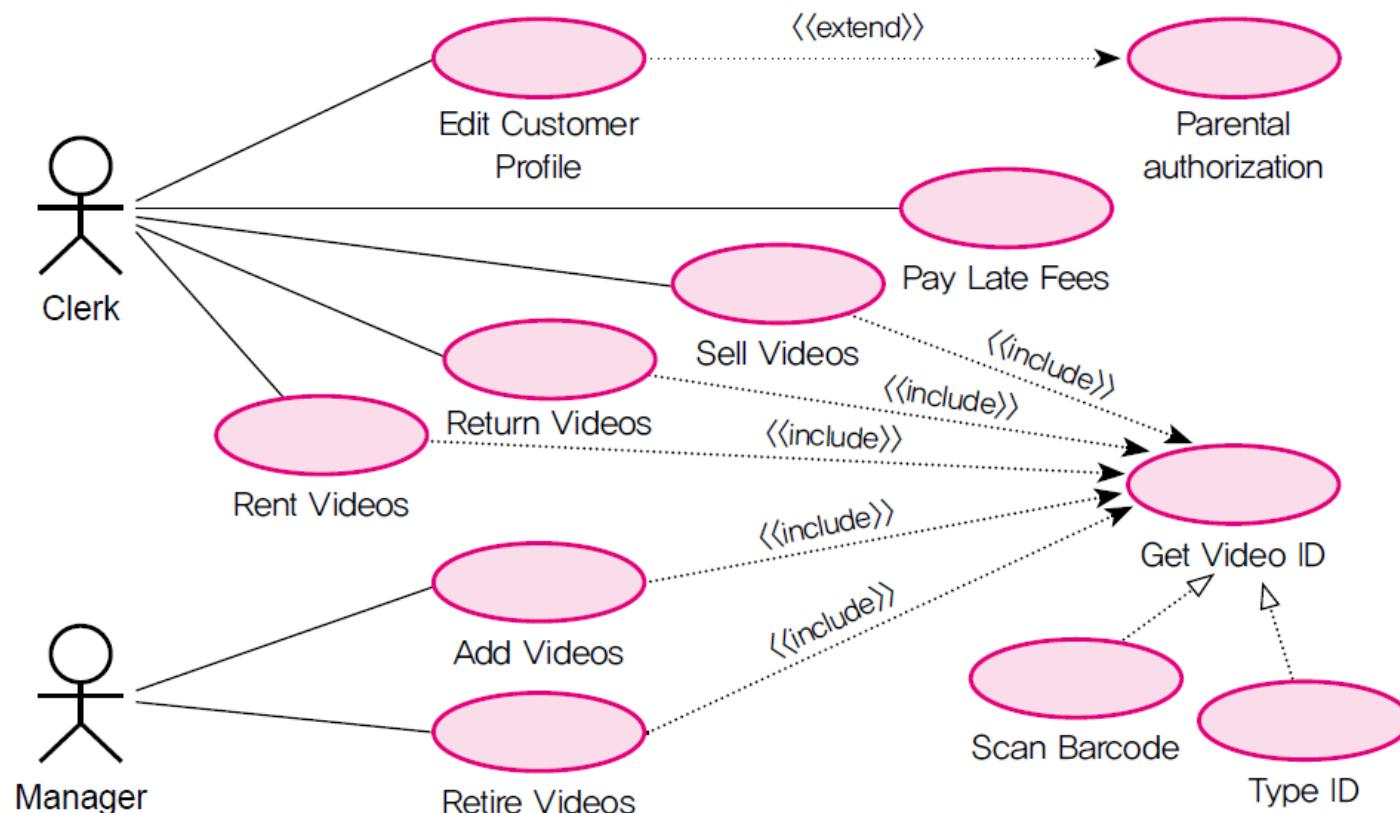
## 확장 관계

- 사용 사례가 일정 조건 아래 확장된 동작을 포함한다면 다른 사용 사례를 확장하는 관계에 있음
- 예외적인 이벤트를 처리하기 위한 사례를 독립적으로 표현
- 예제: 사용자 정보 입력 사례 중 미성년자를 위하여 부모 허락을 받는 사용 사례가 확장되는 경우





## 포함/확장 관계가 적용된 사용 사례 다이어그램





## In this chapter ...

요구 분석 (Requirement Analysis)

4.1 요구(Requirements)

4.2 요구 추출

4.3 도메인 분석

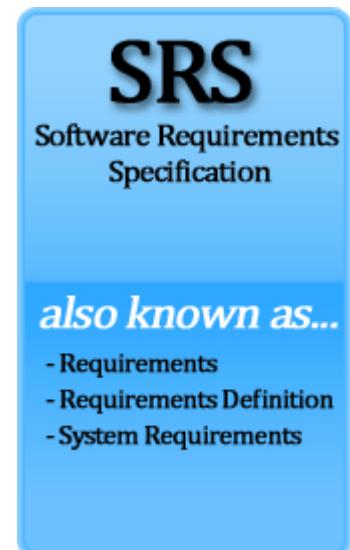
4.4 사용 사례

4.5 요구 분석 명세서

4.6 요구 관리 도구



- ▣ 시스템의 기능을 정확하고 완벽하며 일관성 있게 작성한 것
- ▣ 소프트웨어에 포함될 기능과 제약 조건들을 나열
- ▣ 기능에 대한 자세한 설명과 예외처리 기술
- ▣ 시스템 성능과 관련된 사항, 속도, 정확성, 사용 용이성 포함
- ▣ 과정
  - 명세서 작성
  - 명세서 검토





## ④ 사용자와 개발자 간의 이해를 돋기 위함

### ⑤ Gilbert가 제안한 요구 분석서 작성 시 주의 사항

- 사용자와 개발자 모두가 쉽게 이해할 수 있도록 작성해야 한다.  
→ 무릇 모든 문서/발표는 독자/청중의 수준에 맞추어야 한다.
- 기술된 조건은 개발자와 사용자 모두가 동의한 것이어야 한다.  
→ 자기의 이익을 앞세운 조건이 들어가면 계약이 될 수 없다.
- 제안된 시스템에 의하여 수행될 모든 기능을 정확히 기술하여야 한다.  
→ 어떤 방법으로 제공하는가 보다는 무슨 기능을 제공하는가에 초점
- 제안된 시스템에 영향을 주는 모든 제약 조건을 기술하여야 한다.  
→ 목표 하드웨어, 하드웨어 용량, 동작 환경 등
- 시스템의 인수를 위한 테스트 기준을 제공하여야 한다.  
→ 성능(1000 TPS), 용량(100만 가입자), ...
- 시스템의 품질과 상대적인 중요도 및 품질 측정 방법이 기술되어야 한다.  
→ 기능/성능보다 안전성(우주선), 편리성보다 정확성(현금 지급기), ...



## 1. 개요

### 1.1 시스템의 목적

### 1.2 범위

### 1.3 정의, 약어

### 1.4 참조

## 2. 기능적 요구

### 2.1 외부 인터페이스 요구

#### 2.1.1 사용자 인터페이스

#### 2.1.2 하드웨어 인터페이스

#### 2.1.3 소프트웨어 및 통신 인터페이스

### 2.2 기능 요구

#### 2.2.1 기능 #1(사용 사례 #1)

#### 2.2.2 기능 #2(사용 사례 #2)

...

## 3. 기타 요구 및 제약 사항

### 3.1 성능 요구(반응 시간, 처리 소요 시간, 처리율)

### 3.2 H/W 요구(기억 장치 규모, 통신수용도)

### 3.3 예외 조건 및 이의 처리

### 3.4 자원, 인력에 대한 제약 조건

## 4. 인수 조건

### 4.1 기능 시험 및 성능 시험

## 5. 참고 자료



## <정수에 대한 사칙연산의 예제>

- |                       |                          |
|-----------------------|--------------------------|
| 무결성(correctness)      | "+"에 대한 정의가 바르게 되어야      |
| 완벽성(completeness)     | 네 가지 연산을 모두 정의하여야        |
| 일관성(consistency)      | 모두 정수에 대해 정의하여야...       |
| 명확성(unambiguous)      | '±' 등으로 한꺼번에 나타내지 않아야... |
| 기능성(functionality)    | 레지스터가 어쩌고...가 아니어야       |
| 검증 가능성(verifiability) | 연산이 바른지 확인할 수 있어야        |
| 추적 가능성(traceability)  | 연산이 어디에 사용되었는지...        |
| 변경 용이성(modifiability) | 연산의 변경이 가능해야...          |



## In this chapter ...

요구 분석 (Requirement Analysis)

4.1 요구(Requirements)

4.2 요구 추출

4.3 도메인 분석

4.4 사용 사례

4.5 요구 분석 명세서

4.6 요구 관리 도구



## ④ 요구 변경은 소프트웨어 프로젝트에서 매우 흔한 일임

- 새로운 요구의 추가
- 기존 요구의 삭제/변경

## ⑤ 요구 관리란?

- 요구 변경과 관련된 모든 이슈를 다루는 메커니즘
- [그림 4.19] 요구 관리 작업

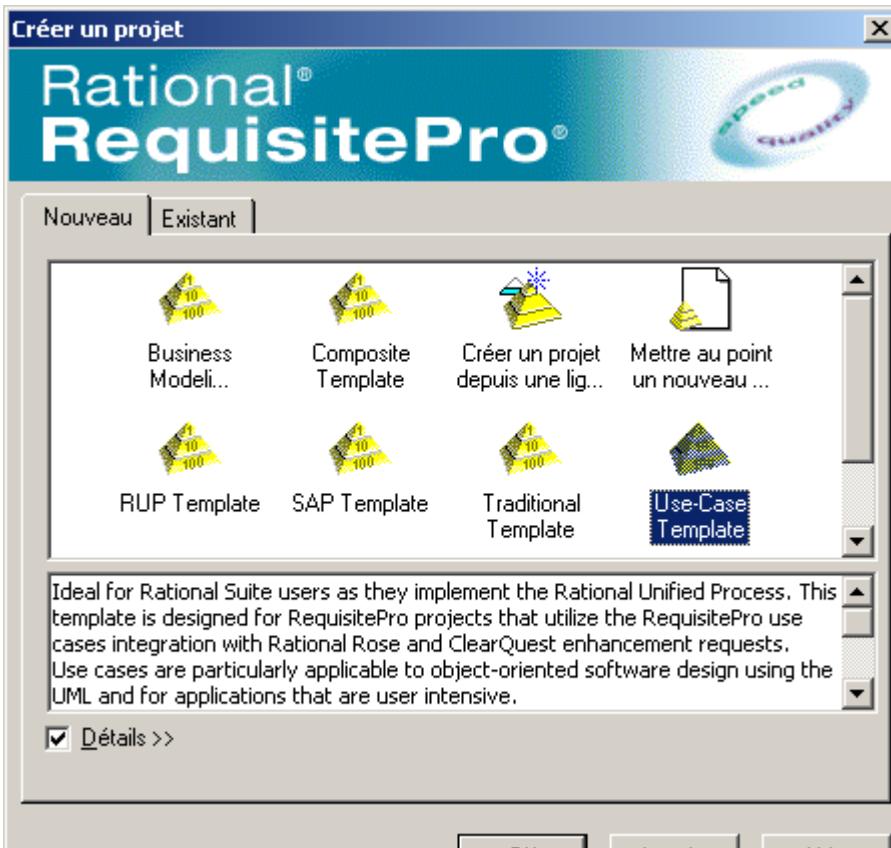




# 요구 관리 도구 사례 (1/2)

요구 분석 (Requirement Analysis)

## IBM Relational의 RequisitePro



The application window has a title bar 'RequisitePro - DemoForIntegration - [SUS: ALL]'. The menu bar includes 'Requirement', 'Traceability', 'Tools', 'Window', and 'Help'. The toolbar contains icons for New, Open, Save, Print, Find, Replace, and others. The left pane shows a tree view with nodes like 'Analysis', 'Design', 'Requirements', 'Requests', 'Stories', and 'Management Plan'. The right pane displays a table titled 'Requirements' with the following data:

| Requirements      | Traced-to | RootTag# | CQ Requirem | TM TestCase |
|-------------------|-----------|----------|-------------|-------------|
| SUS1: SUS1        | 1         | 33629991 | RAPID000015 |             |
| SUS2: SUS2        | 2         | 33629995 | RAPID000015 |             |
| SUS3: SUS3        | 3         | 33629999 | RAPID000015 |             |
| * <Click here...> | pending   |          |             |             |

At the bottom of the right pane, a message says 'Multiple requirements selected.' and '3 requirements'.



## Eclipse에서 사용 가능한 오픈소스 ProR

The screenshot shows the RMF ProR Standalone application window. The menu bar includes File, Edit, Navigate, Search, Project, Run, ProR, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Print, and search. The central workspace has three tabs: Project Explorer, Requirements Document, and Outline. The Requirements Document tab is active, showing a table with columns Description and Link. The table contains requirements: 1. 앱 적용 기술 확정, 2. 메뉴 확정, 2.1 주메뉴, and 2.2 부메뉴. The Outline tab shows a hierarchical tree of specifications and objects, including 'Specifications' and 'SpecObjects' nodes. The Properties tab at the bottom displays requirement details: Type is '부메뉴' (Submenu) under Requirement Type, and Value is '부메뉴' under Description. The status bar at the bottom shows 'Standard Attributes | All Attributes'.

|     | Description | Link |
|-----|-------------|------|
| 1   | 앱 적용 기술 확정  |      |
| 2   | 메뉴 확정       |      |
| 2.1 | 주메뉴         |      |
| 2.2 | 부메뉴         |      |



# In this chapter ...

요구 분석 (Requirement Analysis)

4.1 요구(Requirements)

4.2 요구 추출

4.3 도메인 분석

4.4 사용 사례

4.5 요구 분석 명세서

4.6 요구 관리 도구