

## 요구사항의 유형

### 1. 사용자 요구 사항

시스템을 사용하는 주체인 사용자 관점으로 시스템이 제공할 것을 요구하는 서비스 사양과 제약 사항

### 2. 시스템 요구 사항

전체 시스템이 만족시켜야 하는 요구 사항

### 3. 기능 요구 사항

소프트웨어로 수행하고자 하는 기능에 관해 기술한 것으로, 목표 시스템이 반드시 수행해야 하거나 목표 시스템을 이용하여 사용자가 반드시 수행할 수 있어야 하는 기능을 서술한 것

### 4. 비기능 요구 사항

SW 시스템이 기능 요구 사항을 만족시키면서 정상적으로 작동하기 위한 시스템 내부 또는 외부의 제약 조건

---

## 비기능적 요구 사항에 포함될 수 있는 사항

### 1. 사용성

학습의 용이함, 쉬운 사용, 오류 방지 및 복구, 상호 작용의 효율성, 접근 용이성 등을 작성

### 2. 성능

구체적인 성능 요구 사항을 기술하여 작성. 서로 다른 기능적 요구 사항이나 특성이 각기 다른 성능 요구 사항을 갖는다면 각 기능적 요구 사항에 상응하는 성능 목표를 구체화하여 작성

### 3. 보안성

구체적인 제품에 대한 접근이나 사용을 제한하는 보안이나 개인 정보 보호 이슈에 관한 모든 요구 사항을 구체화하여 작성한다.

### 4. 안정성과 제약 요구 사항

제품 사용 시 발생할 수 있는 손실이나 손상, 피해에 대한 요구 사항을 구체적으로 설명

# 기능 요구 사항 분석 기법 중 유스케이스(Use Case)와 사용자 스토리

## [유스케이스]

시스템 경계 밖에 위치한 액터가 특정 목적을 달성하기 위해 시스템이 제공하는 기능을 이용하여 시스템과 일련의 상호 작용을 주고받는 시나리오로 나타낸 것

## [사용자 스토리]

시스템의 사용자나 고객과 같이 신규 기능을 필요로 하는 사람의 관점에서 서술하는 기능에 대한 간략하고 간단한 설명

---

## 유스케이스(Use Case)의 요소

시스템 경계 : 시스템의 범위 표시

액터 : 시스템과 상호 작용하는 시스템 외부 객체

유스케이스 : 시스템에 의해 수행되는 행위 또는 기능

연관 : 액터와 유스케이스 간의 상호 작용 관계

---

## 요구사항 도출 기법

- 인터뷰, 문서분석, 워크숍, 시나리오

---

## 사용자 별 요구사항 도출 기법

(사용자의 전문적 지식 수준, 개발조직의 경험 여부에 따라 나누어짐)

- 사용자의 전문적인 지식수준이 높은 경우 : 인터뷰, 문서분석
  - 사용자의 전문적인 지식수준이 낮은 경우 : 워크숍, 시나리오
  - 개발조직의 경험이 많은 경우 : 문서분석, 시나리오
  - 개발조직의 경험이 적은 경우 : 인터뷰, 워크숍
-

## 요구 사항 유형별 문서의 종류

비즈니스 요구 사항 문서, 비즈니스 규칙 문서, 사용자 요구 사항 문서, 기능적 요구 사항 문서, 품질 속성 문서, 외부 인터페이스 문서, 제약 조건 문서, 데이터 요구 사항 문서, 솔루션 아이디어 문서

---

## 요구 사항 도출 프로세스 진행 시의 고려 사항

1. 방법이 아니라, 무엇을 해야 하는지에 초점을 맞추어 기술한다.
  2. 적절한 문법과 철자법으로 완전한 문장을 작성한다.
  3. 능동적인 어법을 사용한다.
  4. 용어집에 정의된 용어를 일관성 있게 사용한다.
  5. 모호하지 않은 수준까지 분해하여 작성한다.
  6. 가시성을 높이기 위해 가능한 한 목록, 테이블 및 그래프를 사용한다.
  7. 주관적이거나 모호한 용어는 피한다.
  8. 대부분의 요구 사항은 비즈니스 문제로부터 시작된다.
  9. 대부분의 요구 사항은 현 시스템을 기반으로 한다.
- 

## 요구 사항 정리 기록 보관 방법

요구 사항에 관련된 자료는 조직 내에서 매우 중요하게 관리해야 하며, 해당 자료에 대해 권한 그룹을 생성하고, 문서·정보의 보안이 이루어질 수 있도록 한다.

---

## 요구 사항 명세서 작성 시 검토해야 하는 속성 및 내용 항목

- 정확성 - 명세화된 요구 사항이 실제 시스템 구현 시 필요한 것인지 알 수 있도록 정확해야 한다.
- 명확성 - 요구 사항을 혼동하지 않도록 한 가지 단어로 해석되어야 한다.
- 완전성 - 시스템이 구현될 때 필요한 요구 사항이 빠짐없이 모두 반영되어야 한다.
- 일관성 - 요구 사항들 간의 충돌이 없어야 한다.
- 수정 용이성 - 구조와 스타일의 일관성이 유지되면서 요구 사항 수정이 용이해야 한다.
- 추적성 - 각각의 요구 사항들이 관련 있는 요구 사항들과 유기적으로 연결되어야 한다.