

# 경영정보시스템 12주차 강의

## 교재 제 12장. 정보시스템의 구축

### 12주차 수업내용

#### 1. 시스템의 개발

조직변화와 시스템 개발

시스템 개발의 주체

시스템 개발방법론들의 비교

#### 2. 시스템 수명주기 방법

#### 3. 프로토타이핑 방법

#### 4. 기타 시스템개발 방법론

# 1. 시스템의 개발

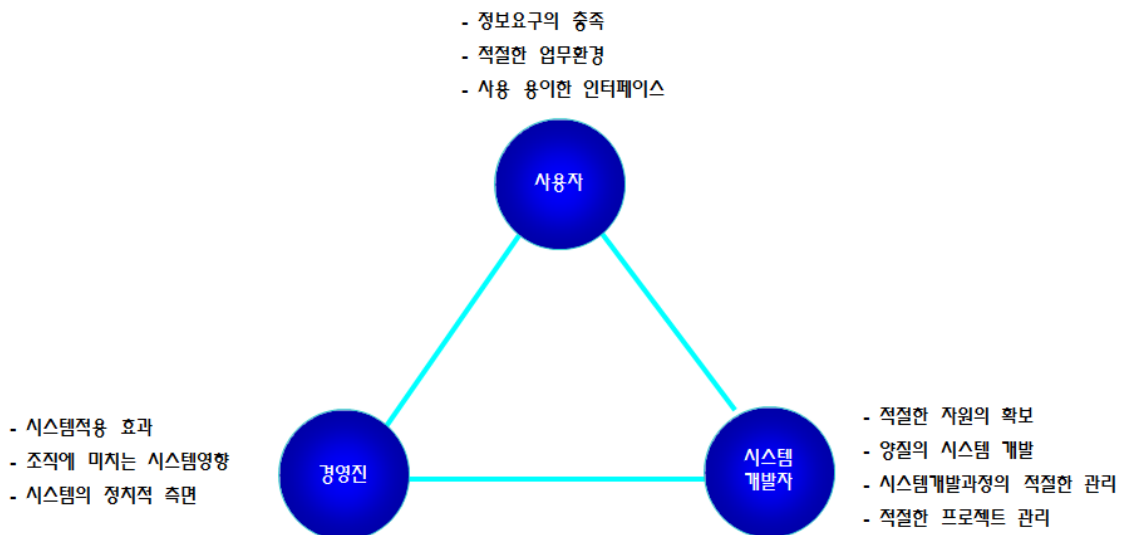
## 조직변화와 시스템 개발

1) 정보시스템의 도입은 조직전반에 걸쳐 큰 영향을 미치므로 '조직 변화' (organization transformation) 관점에서 개발 필요

2) 정보시스템 개발의 네 가지 측면:

- 기술적 기능의 정확한 구현
- 인간측면에 기초하여 사용자 인터페이스 설계
- 시스템이 조직에 미치는 전반적인 영향을 고려하여 개발 추진
- 경영관리자의 시각에서 시스템 설계 및 구현

## 시스템 개발의 주체



법문사 교재 그림 12-1 참조

## 시스템 개발방법론

개발방법	주요 특성
SDLC 방법을 통한 내부개발	<ul style="list-style-type: none"> <li>• 단계적인 시스템 개발 (분석, 설계, 구현)</li> <li>• 거래처리시스템과 같은 전사적인 시스템 구축에 적합</li> <li>• 변하는 사용자 요구에 대해 유연한 대응이 어려움</li> </ul>
프로토타이핑 방법을 통한 내부개발	<ul style="list-style-type: none"> <li>• 사용자의 피드백을 토대로 초기 프로토타입을 반복적으로 수정·보완함으로써 시스템 개발</li> <li>• 신속하고도 용이하게 개발 프로젝트를 완료할 수 있음</li> <li>• 자칫하면 체계없는 시스템으로 진화되어 시스템 유지보수가 어려워질 수 있음</li> </ul>
고속 애플리케이션 개발 (RAD)	<ul style="list-style-type: none"> <li>• 그래픽중심 사용자 인터페이스를 신속, 효율적으로 개발할 수 있음</li> <li>• RAD 툴을 이용해서 반복적으로 최종시스템을 확장시켜 나감</li> <li>• 대규모 팀에 의해 복잡한 시스템을 구축하여야 하는 경우에는 적합하지 못함</li> </ul>
객체지향 개발	<ul style="list-style-type: none"> <li>• 프로그램과 데이터를 하나의 객체에 캡슐화시켜 객체단위로 프로그램을 개발하고 관리</li> <li>• 웹 프로그램을 개발하는데 가장 이상적임 (예: 자바 애플릿)</li> <li>• 기존 코드를 재사용할 수 있으므로 시스템의 유지보수가 효율적이고 용이함</li> </ul>
최종사용자지킴이등을 통한 내부개발	<ul style="list-style-type: none"> <li>• 사용자들이 자기부서의 사용을 위해 스스로 시스템 개발</li> <li>• 정보시스템 부서의 인력을 소모하지 않으며 이용자 만족도 높음</li> <li>• 체계적인 개발방법의 부재로 인해 위험요인이 따를 수 있음</li> </ul>
소프트웨어 패키지의 구입	<ul style="list-style-type: none"> <li>• 범용패키지를 구입하여 조직용도에 맞게 커스토마이징함</li> <li>• 적은 비용으로 신속하게 업무에 적용하는 것이 가능함</li> <li>• 업무 프로세스가 시스템의 도입으로 크게 변화할 수 있으므로 프로세스 혁신(process innovation)과 병행하여 추진하는 것이 좋음</li> </ul>
시스템통합업체 위탁을 통한 외부개발	<ul style="list-style-type: none"> <li>• 외부 전문업체(System Integrator: SI)에 시스템 개발을 위탁</li> <li>• 정보시스템 부서 인력의 업무부담 경감</li> <li>• SI업체의 다양한 경험 활용(기술력, 문제해결능력 등)</li> <li>• 신기술에 대한 조직학습 효과를 크게 기대하기 어려움</li> </ul>

### 법문사 교재 표 12-1 참조

## 2. 시스템 수명주기 방법

### SDLC의 개념 및 특징

- 1) SDLC (System Development Life Cycle)라고도 불림
- 2) 사용자 요구에서부터 완성된 시스템에 이르기까지의 전반적인 개발과정을 정형화된 일련의 단계들을 통해 추진하는 방법
- 3) 산업공학의 프로젝트관리 기법에 기초한 구조적인 방법론
- 4) 총체적 접근방법(comprehensive approach)
- 5) 다양한 분석/설계 도구
  - DFD(data flow diagrams)
  - flow chart
  - CASE(computer-aided system engineering) 소프트웨어
- 6) 상당한 자원소요 (시간 및 비용 소모적)
- 7) 전략정보시스템이나 대규모 전사적 시스템 등 개발에 적합
  - 정보요구의 사전 도출이 비교적 수월한 경우에만 SDLC방법 적용 타당

## SDLC의 단계 및 내용

- 1) 시스템 조사 및 분석 : 타당성 조사, 요구사항분석
  - 결과물 : 시스템 제안서, 요구사항 분석서
  - 시스템의 필요성/목적 결정
  - 기존시스템의 문제점을 조사하고 새 시스템의 전략적 기회제공의 가능성 탐색
  - 기 수립된 정보시스템 계획(IS plan)에 기초하여 수행
  - 조직 환경, 조직, 기존 시스템에 대해 조사
    - 현 업무절차와 기존시스템의 구체적 문제점 파악
    - 인터뷰, 설문, 관찰, 문서검토 등의 방법을 통해 정보 수집
  - 사용자의 정보 요구 결정
    - 사용자를 시스템분석 과정에 최대한 참여시키는 것이 바람직함
  - 사용자 요구에 근거하여 시스템 능력/기능 결정
  
- 2) 시스템 설계 : 개념설계, 세부설계
  - 결과물 : 프로그램 및 DB의 개념 설계, 세부 스펙
  
- 3) 제작 : 프로그래밍/테스트
  - 시스템 완성 및 매뉴얼
  
- 4) 구현 : 설치
  - 소프트웨어 프로그래밍 : 코딩, 디버깅 및 테스트
  - 하드웨어 구입
  - 시스템 전환 : 이전 시스템을 새 시스템으로 교체
    - (병행방식, 맞교환, 단계적전환, 실험적 전환)
  - 사용자 교육 : 시스템 효과, 사용방법
  
- 5) 시스템 운영 및 보수 : 성능평가, 사용자 Feedback, 개선/보완, 유지보수
  - 개선사항 도출

## 타당성 조사

### 1) 경제적 타당성

- 시스템이 기업에 전략적 우위나 경제적 이익을 가져다 줄 것인가? 비용을 감당할 수 있는가?
- 비용/효익 분석(cost/benefit analysis): 비용과 효익 항목 간의 비교

### 2) 기술적 타당성

- 시스템의 개발과 운영에 필요한 기술 및 인력 및 노하우를 갖추고 있는가?
- 우리가 할 수 없다면 아웃소싱을 통해서도 개발이 가능하겠는가?

### 3) 행태적 타당성

- 시스템이 사용자에게 어떠한 영향을 미치겠는가?
- 사용자가 큰 어려움 없이 시스템에 적응할 수 있겠는가?

## 경제적 타당성의 예

시스템 비용항목	시스템 효익항목
<b>고정비용 (초기 투자)</b>	<b>비용절감 효과</b>
- 하드웨어 구입	- 인력비용 절감
- 소프트웨어 개발/구입	- 하드웨어 비용 절감
- 데이터베이스 제작	- 구매비용 절감
- 사용자에게 대한 교육	- 재고비용 절감
- 새로운 업무절차 개발	
<b>운영비용 (지속적 투자)</b>	<b>수입증대 효과</b>
- 인적자원 비용 (운영 및 보수유지)	- 기존시장에서의 매출증대
- 시설 관리비, 소모품구입비 등	- 시장 확대

법문사 교재 표 12-4 참조

### 3. 프로토타이핑 방법

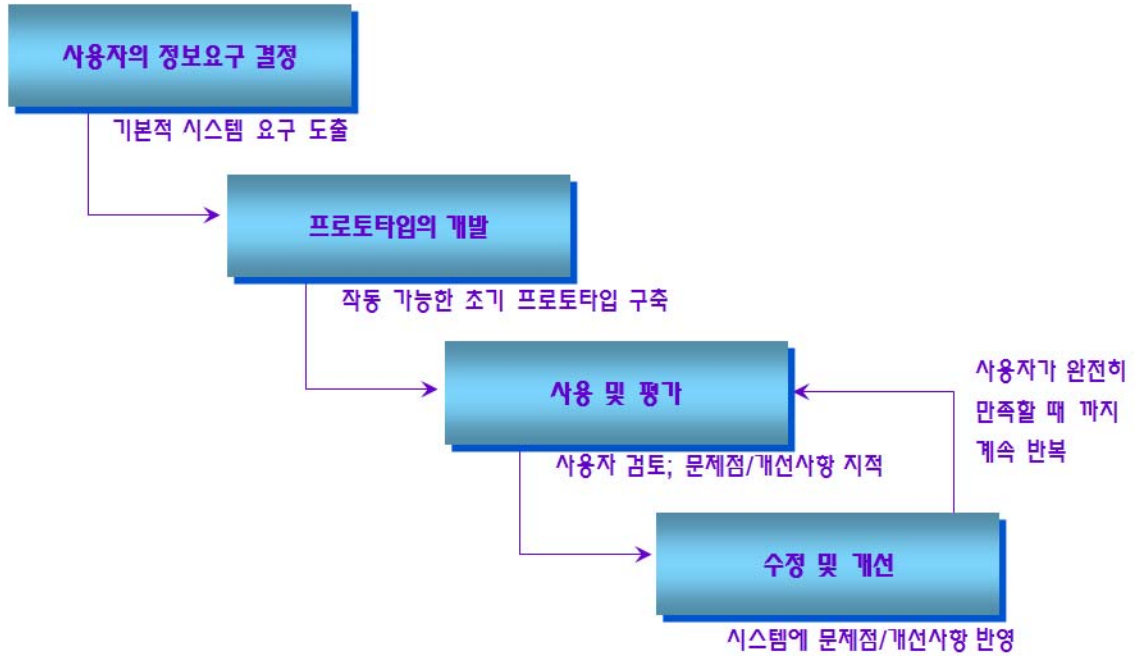
#### 프로토타이핑 방법 사용 시 전제

- 1) 시스템요구를 미리 결정하는 것은 매우 어려울 수 있다.
- 2) 사용자는 시스템개발에서 중심적 역할을 담당해야 한다.  
(시스템설계에의 참여 필요성)
- 3) 사용자는 제안된 상상적 시스템에 대해 장단점을 평하는 것보다는 이미 존재하는 시스템에 대해 더욱 정확하게 평할 수 있다.
- 4) 시스템개발은 실험적으로 진행될 때 더 큰 효과를 기대할 수 있다.

#### 프로토타이핑 방법 개념 및 특징

- 1) 작은 prototype(working model)를 개발하여 반복적으로 프로토타입시스템 확장/개선
- 2) 수정시마다 사용자요구를 수렴하여 시스템에 반영  
(가능한 한 한번에 큰 변화를 구현함)
- 3) 신속한 개발 / 소규모자원 소요 / 복잡성 감소 / 사용자 만족도 증대
- 4) 새시스템에 대한 교육효과
- 5) 주로 4GL과 함께 사용
- 6) DSS, ES, EIS 등의 개발에 적합

## 프로토타이핑 단계 (법문사 교재그림 12-5 참조)





# SDLC Vs Prototyping 비교

## SDLC

### 1) 장점

- 시스템요구가 미리 파악 가능한 경우에 적합 (거래처리시스템과 같이 일상업무지원을 위한 시스템을 개발할 경우)
- 복잡한 대형시스템 개발 프로젝트를 효과적으로 관리가능

### 2) 문제점

- 많은 자원/시간 소요
- 프로젝트 관리의 복잡성

## Prototyping

### 1) 장점

- 사용자가 자신이 제시한 피드백의 구현 결과를 즉시 확인가능
- 아이디어를 신속하게 테스트할 수 있어 시스템을 잘 못 설계할 위험부담 감소
- 사용자가 시스템 개발에 참여할 수 있도록 동기 부여

### 2) 단점

- 전사적인 대형시스템의 개발에 부적합
- 완성이 되어도 곧 업무환경에 적용하기가 현실적으로 어려움
- 대량데이터를 통한 시스템 테스트 미흡 매뉴얼 등 문서작성 소홀

## 4. 기타 시스템개발 방법론

고속 애플리케이션 개발(RAD)

객체지향 개발

전문 패키지

최종사용자컴퓨팅을 통한 내부개발

정보시스템 아웃소싱