

5 주차

목차

1. 프로젝트 관리 필요성
 2. 프로젝트 관리 기법
 3. 프로젝트 조직
 4. 프로젝트 관리 계획서
 5. 프로젝트 지원 도구
-

1. 프로젝트 관리 필요성

SW개발 프로젝트가 실패하는 이유

1. 부족한 소프트웨어 마인드 - 하드웨어와 달리 물리적으로 존재하기 않기 때문에 언제든지 변경이 가능하다고 생각한다.
2. 소프트웨어 공학 기술의 활용 미흡 - 업무 증가와 일정 지연 등의 이유로 공학 기법 적용을 미룬다.
3. 부족한 프로젝트 관리 기술 - 프로젝트 고려 요소가 기하급수적으로 증가, 개발 기술의 고도화

프로젝트 관리의 정의

개발자 또는 개발 팀이 프로젝트 목표를 효율적이고 효과적으로 달성하는 데 필요한 내적 환경 요소들을 준비하고 유지하는 활동

프로젝트 관리 단계

1. **계획 수립** - 소프트웨어 개발의 목적, 필요한 자원, 정보의 흐름, 소요 인력, 산출물 등을 정의한다.
2. **자원 획득** - 계획 단계에서 예측한 다양한 장비 및 시설 확보, 팀 구성 등과 같은 일을 수행하여 실제로 프로젝트가 수행될 여건을 마련한다.
3. **실행** - 프로젝트 수행 계획서에서 정의한 일정에 따라 SW 개발 활동을 수행
4. **모니터링** - 프로젝트 관리 계획서에서 정의한 **베이스라인**에 근거하여 개발 활동에 대한 진척도를 관리하고 품질을 모니터링한다.

프로젝트 관리의 실패 원인

부정확한 요구사항, 사용자 환경 이해 부족, 프로젝트 관리 기술 부족, 부족한 자원 등등... etc

2. 프로젝트 관리 기법

일정 관리 기법, 비용 관리 기법, 위험 관리 기법 - 세 가지로 나뉨

1. 일정 관리 기법

베이스 라인

어떤 변경이 발생했을 때, 모든 관련자의 합의를 거쳐 변경사항을 결정한 다음, 후속 작업을 수행하기 위한 출발점이라고 정의한다.

베이스 라인 관리

1. **요구사항 베이스 라인**: 요구사항을 수집하고 프로젝트 계획을 수립한 후에 정의하는 베이스라인
2. **기능 베이스 라인**: 요구사항 분석 단계 후에 설정하는 베이스 라인
3. **제품 베이스 라인**: 소프트웨어 개발을 완료한 후에 정의하는 베이스 라인이며 개발 과정의 모든 산출물이 이 베이스 라인 기준선에 포함된다.

작업 분할도(WBS: Work Breakdown Structure)

프로젝트의 전체 목표를 중간의 세부 목표들로 쪼개어 나타낸 작업 목록이다. 프로젝트에서 수행해야 하는 모든 종류의 작업이 나타나게 된다.

칸트 차트

프로젝트의 스케줄링, 예산산정, 자원 계획을 수립하기 위해 사용하는 일정 표현 기법
슬랙 타임: 작업을 완료해야 하는 마지막 시간

퍼트 차트

■ 파트 차트(PERT Chart)

- 태스크 A, B와 D는 병행으로 수행 가능, 태스크 C는 반드시 A가 종료되어야 수행
- 임계 경로에 있는 작업 일수의 합 45일
- Latest Start Date(F) = $45 - (5 + 15 + 7) = 45 - 27 = 18$

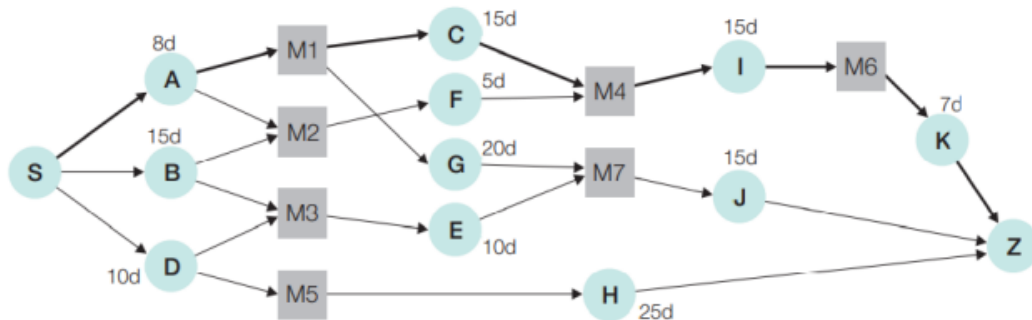


그림 5-6 프로젝트 일정을 나타내는 PERT/CPM 차트의 예

프로젝트에서 수행되어야 하는 태스크 간의 **의존성**을 표현하는 차트
프로젝트 전체 수행 일정에서 핵심적인 수행 경로를 식별 관리한다.

- 임계 경로: 각 경로 상에 있는 태스크 수행 일수의 합이 가장 많은 핵심 경로 → 임계 경로를 기준으로 프로젝트 수행 기간을 정하면 된다.

2. 비용 관리 기법

개발 비용을 산출하는 기법들

1. COCOMO/COCOMO 2 - 개발 대상 소프트웨어에 대한 KDSI를 유사 소프트웨어를 비교 분석하여 개발 소프트웨어 복잡도에 근거하여 개발 기간과 필요 인원수를 산정한다.
2. 전문가의 판단(WBD: Wide Band Deliphi) - 다수의 전문가가 모여서 소프트웨어 요구사항을 기준으로 SW 크기를 예상
3. 파킨슨의 법칙 - 어떤 방법으로 소프트웨어 개발 비용을 산정다더라도 조직에서 확보한 인력이 해당 소프트웨어를 개발하기에 충분한지 여부와 무관하게 조직의 인력에 개발 일정이 맞춰져 진행된다.
4. 기능 점수 산정법 - SW 시스템이 제공하는 기능의 필요 정도와 이의 복잡도를 기준

3. 위험 관리 기법

재작업, 의사소통 증가, 개발자의 능력부족과 같은 SW 개발 계획 단계에서 식별되는 리스크를 정리하고 대책을 마련한다.

3. 프로젝트 조직

프로젝트 팀 구조

1. 중앙 집중형 팀 구조

프로젝트에서 수행해야 할 작업 목록이 단순하거나 충분히 이해된 경우에 적합한 팀 구성 방식이다.

유능한 프로젝트 리더가 필요하다.

문제 해결이 신속하게 이루어질 수 있고, 의사소통의 패턴이 매우 단순하다.

2. 분산형 팀 구조

프로젝트의 주요 의사결정이 팀 구성원의 합의에 의해 이루어지는 민주주의적 팀 구성이다.

문제가 복잡하여 해결 방안을 모색해야 하는 경우에 적합하다.

대규모 구성원을 포함하는 프로젝트에는 적합하지 않을 수 있다.

3. 하이브리드 팀 구조

중앙 집중형 팀 구조와 분산형 팀 구조를 통합한 계층형 구조이다.

프로젝트 관리자(PM)는 각 팀의 리더와 의사 결정을 위한 중앙집중형 구조를 취하고, 팀 내부의 운영은 분산형 구조를 채택하여 의사소통한다.

팀원을 소규모로 구성해 팀별 문제 해결이 용이하다.

팀 구조 선정 전략

프로젝트의 기간, 문제의 복잡성, 참여 인력 등의 요소를 고려하여 어떤 구조의 프로젝트 팀을 구성할 것인지 결정한다.

4. 프로젝트 관리 계획서

계획서 작성시 유의사항 ← 대충 읽음 중요도 하

1. 문서의 목적 : 해당 문서가 왜, 어떤 목적으로 작성되며, 어떠한 내용을 포함하는 지 설명한다.
2. 관련 문서: 해당 문서와 관련된 공식적인 이전 베이스라인 산출물, 관련 표준등을 다룬다.
3. 3.1 팀 구조 목차에서는 집중형 팀 구조, 분산형 팀 구조, 계층형 팀 구조 중에서 선정한다.
4. 5.3의 품질 관리 적용 기법 목차에서는 리뷰, 인스펙션, 테스트와 같은 품질 관리 활동을 정의한다.

5. 프로젝트 지원 도구

프로젝트 관리 기능

1. 계획 기능: 팀 구성, WBS 생성, 태스크 할당, 일정 계획 생성, 마일스톤을 정의해주는 기능을 제공한다.
2. 진도 관리 기능: 태스크를 개별적으로 수행하고 수행 완료 및 진도 정보를 입력해주는 기능을 제공한다.
3. 협업 기능: 협업 목표 설정, 협업 공간 생성, 상호 의사소통, 자료공유 기능을 제공한다.
4. 배포 기능: 산출물을 저장과 이를 위한 Check-in/Check-out, 검색 기능을 제공한다.
5. 보고서 생성 기능: 태스크 목록, 작업 일정 보고서, 종합 보고서와 같은 다양한 보고서 생성 기능을 제공한다.
6. 통합 기능: 프로젝트 차원에서 활용 가능한 기능을 제공한다.

PMO 도구 ← 안 나올듯

- | | |
|-------------------|------------------|
| 1. 일정 및 진도 관리 기능 | 5. 산출물 품질 관리 기능 |
| 2. 인적 물적 자원 관리 기능 | 6. 이슈 및 위험 관리 기능 |
| 3. 의사소통 관리 기능 | 7. 성과 관리 기능 |
| 4. 조직의 능력 관리 기능 | |