

소프트웨어공학

프로젝트 일정 계획 기법/구성 계획

- 1 PERT vs CPM
- 2 간트 차트
- 3 프로젝트 조직 구성 계획 개요
- 4 프로젝트 팀 구성/구성 종류

01

PERT vs CPM

구분	PERT	CPM
명칭	Program Evaluation and Review Technique	Critical Parh Method
배경	프로젝트 종료시점을 계산하기 위해 1958년 미국 해군의 미사일 개발 프로젝트를 통해 개발된 사업관리 기술	개발일정 추정에 관심을 두며 듀폰사의 화학공장 건설 프로젝트를 통해 개발된 프로젝트 통제 기법
목적	각 단계별 종료 시기를 결정	소요기간 예측
특징	<ul style="list-style-type: none"> PERT는 프로젝트에 필요한 전체 작업의 상호 관계를 표시하는 네트워크로 각 작업별로 낙관적인 경우, 가능성이 있는 경우, 비관적인 경우로 나누어 각 단계별 종료 시기를 결정하는 방법이다. 과거에 경험이 없어서 소요 기간 예측이 어려운 소프트웨어에서 사용한다. 노드와 간선으로 구성되며 원 노드에서 작업을 간선(화살표)에는 낙관치, 기대치, 비관치를 표시한다. 	<ul style="list-style-type: none"> CPM은 프로젝트 완성에 필요한 작업을 나열하고 작업에 필요한 소요 기간을 예측하는데 사용하는 기법이다. CPM은 노드와 간선으로 구성된 네트워크로 노드는 작업을 간선은 작업 사이의 전후 의존 관계를 나타낸다. 간선을 나타내는 화살표의 흐름에 따라 각 작업이 진행되며 전 작업이 완료된 후 다음 작업을 진행할 수 있다. 각 작업의 순서와 의존 관계, 어느 작업이 동시에 수행될 수 있는지를 한눈에 볼 수 있다.

1. 간트 차트 : Gantt Chart

- 프로젝트의 각 작업들이 언제 시작하고 언제 종료되는지에 대한 작업 일정을 막대 도표를 이용하여 표시하는 프로젝트 일정표
- 시간선(time-line) 차트
- 특징
 - ① 중간 목표 미달성 시 그 이유와 기간을 예측 가능
 - ② 사용자와의 문제점이나 예산의 초과 지출 등도 관리
 - ③ 자원 배치와 인원 계획에 유용하게 사용
 - ④ 다양한 형태로 변경하여 사용
 - ⑤ 작업 경로는 표시할 수 없으며, 계획의 변화에 대한 적응성 부족
 - ⑥ 계획 수립 또는 수정 때 주관적 수치에 기울어지기 쉬움
 - ⑦ 간트 차트는 일정표, 작업 일정, 작업 기간, 산출물로 구성
 - ⑧ 수평 막대의 길이는 각 작업(task)의 기간 표시

02

간트 차트

1. 간트 차트 : Gantt Chart

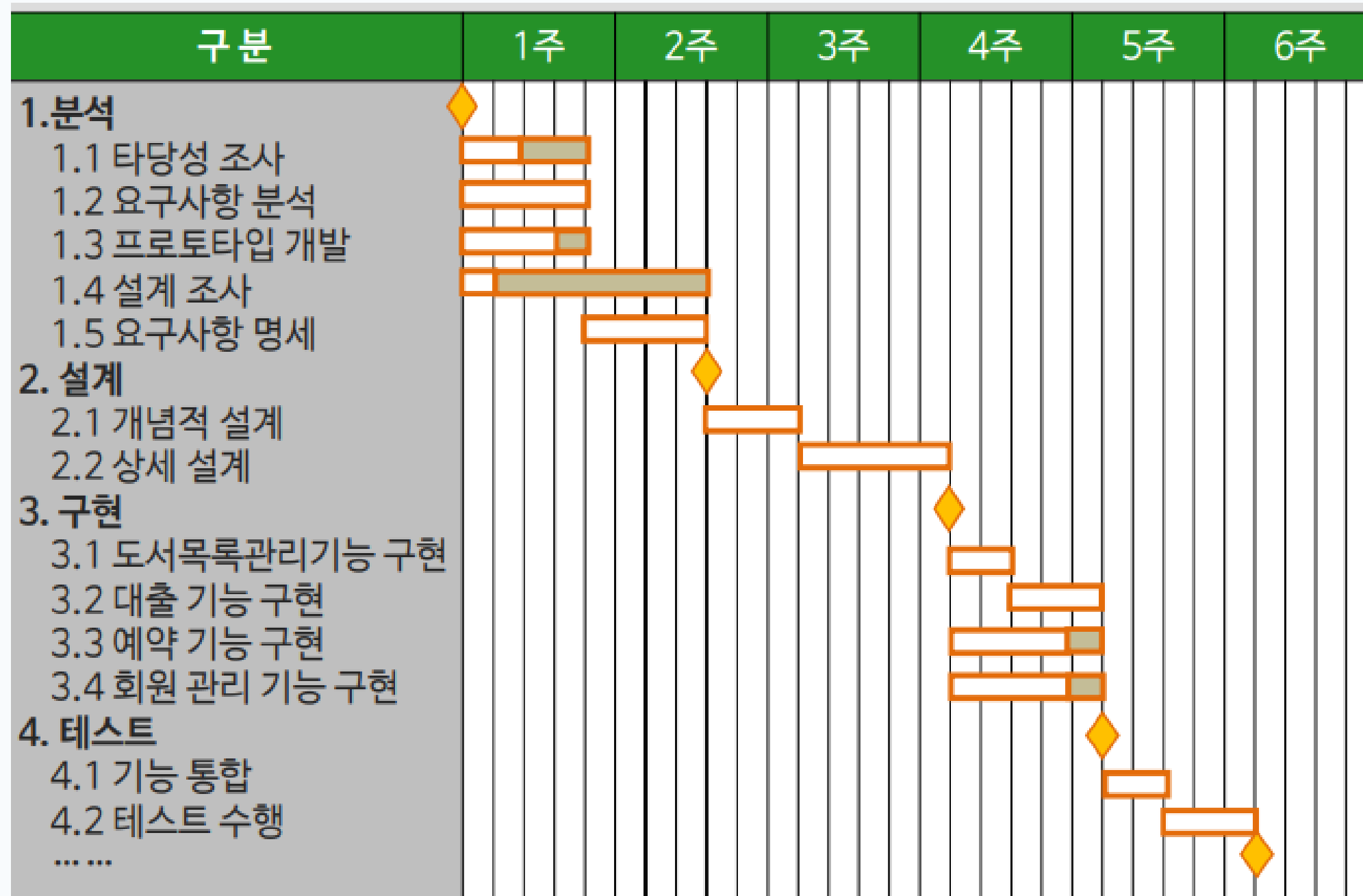
- 예시

작업 일정 작업 단계	이정표												산출물
	1	2	3	4	5	6	7	8	9	10	11	12	
계획	■												시스템 계획서, 프로젝트 정의서
분석		■											요구 분석 명세서
기본 설계			■										기본 설계서
상세 설계				■	■	■							상세 설계서
사용자 지침서			■	■									사용자 지침서
시험 계획					■	■	■						시험 계획서
구현							■	■	■				원시 코드
통합 테스트										■			시스템 통합 계획서
시스템 테스트											■		시스템
인수 테스트												■	개발 완료 보고서

02 간트 차트

1. 간트 차트 : Gantt Chart

- 예시



프로젝트 조직 구성 계획 개요

1. 프로젝트 조직 구성 계획

- 프로젝트를 수행하기 위해 참여하는 각 구성원들의 역할을 할당하고 서로 어떤 방법을 통해 협력할 것인가를 정의
- 프로젝트를 완성하기 위해서는 프로젝트 단위로 팀을 구성하여 수행
- 프로젝트 수행 기간, 작업의 특성, 팀 구성원 사이의 의사 교류 횟수에 의해 팀 구성 방법 차이

03

프로젝트 조직 구성 계획 개요

2. 프로젝트 팀 구성의 종류

- 프로젝트 팀 구성은 의사결정권이 누구에게 있느냐에 따라...
 - 분산형 팀 구성
 - 중앙 집중형 팀 구성
 - 계층적 팀 구성

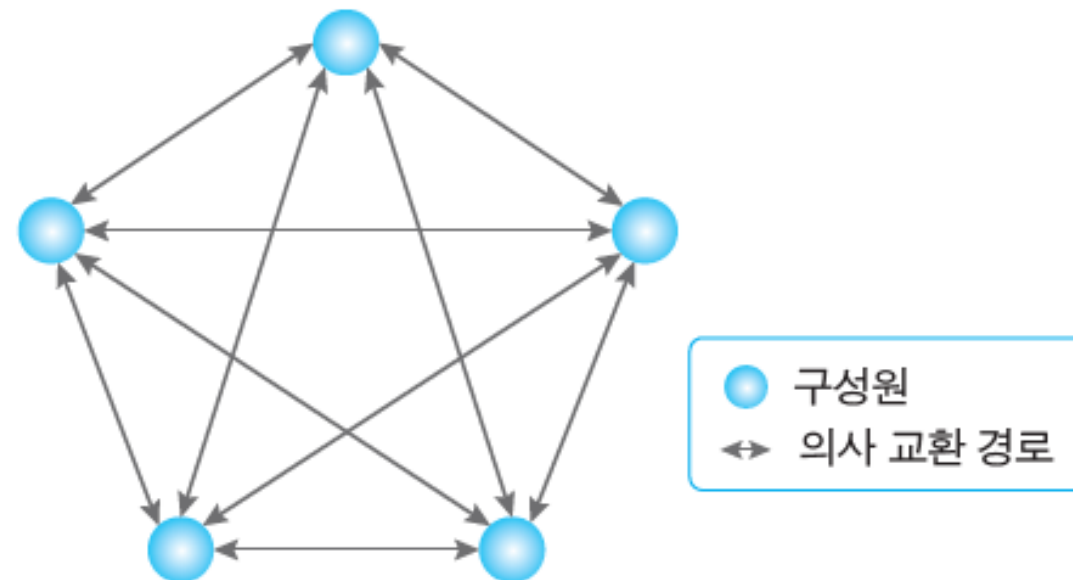
03

프로젝트 조직 구성 계획 개요

3. 프로젝트 팀 구성

분산형 팀 (민주주의 식 팀)

- 팀원 모두가 의사 결정에 참여하는 비이기적인 구성 방식
- 의사 결정을 민주주의식으로 하며 팀 구성원의 참여도와 작업 만족도를 높이고 이직률 낮게함
- 팀 구성원 각자가 서로의 일을 검토, 다른 구성원이 일한 결과에 대하여 같은 그룹의 일원으로 책임, 장기 프로젝트 개발에 적합함
- 팀원 협력심 지원 되지 않으면 낮은 책임감이 문제
- 의사소통 경로가 많아 비생산적일 수 있음
- 의사 소통 경로의 수 = $n*(n-1)/2$ (n : 팀원 수)

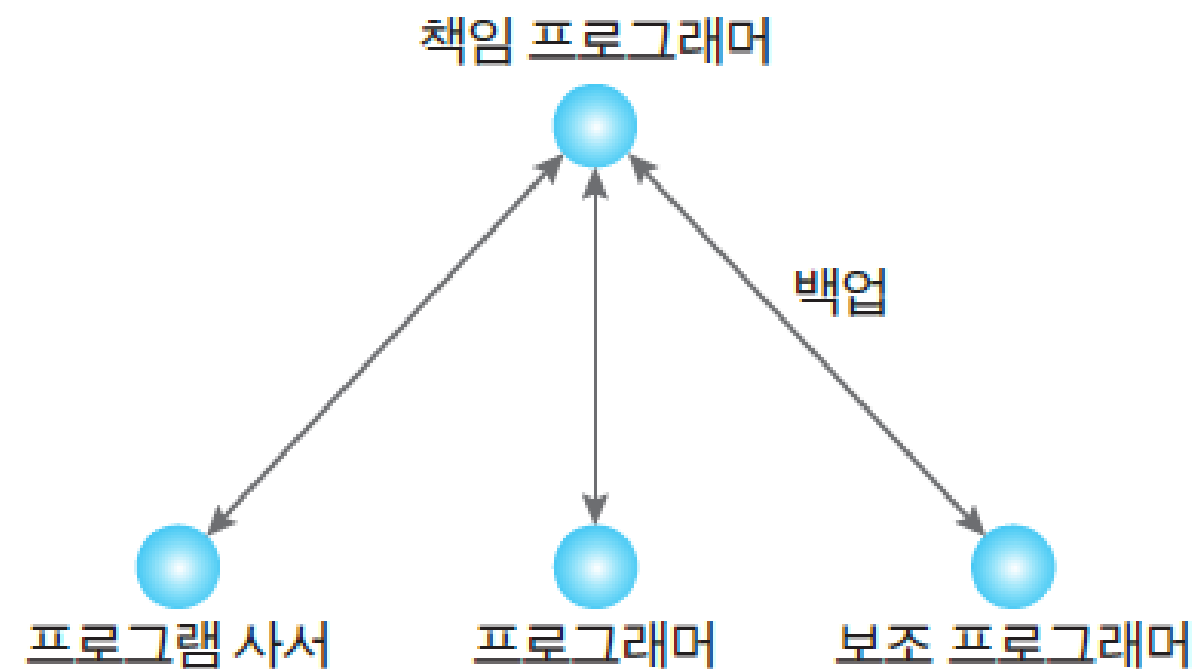


프로젝트 조직 구성 계획 개요

3. 프로젝트 팀 구성

중앙 집중형 팀 (책임프로그래머 팀)	<ul style="list-style-type: none"> 한 관리자가 의사 결정을 하고, 팀 구성원들은 그 결정을 따르는 구성 방식 프로젝트 수행에 따른 모든 권한과 책임을 한 관리자에게 위임하고, 기술 및 관리 지원을 위해 인력을 투입하는 형태 의사 결정이 빠르고, 의사 교환 경로를 줄일 수 있음
-------------------------	---

구성원	역할
책임 프로그래머	요구 분석 및 설계 중요한 기술적 판단 프로그래머에게 작업 지시 및 배분 등
프로그래머	책임 프로그래머의 지시에 따른 원시 코드 작성 테스트, 디버깅, 문서 작성 등
프로그램 사서	프로그램 리스트, 설계 문서, 테스트 계획 등의 관리
보조 프로그래머	책임 프로그래머의 업무 지원 여러 가지 기술적인 문제에 대한 자문 사용자와 품질보증 담당자 등의 섭외 책임 프로그래머 감독하에 분석, 설계, 구현 담당



프로젝트 조직 구성 계획 개요

3. 프로젝트 팀 구성

계층적 팀 (혼합형 팀)

- 분산형 팀 구성과 중앙 집중형 팀 구성을 혼합한 형태 5~7명의 초보 프로그래머를 작은 그룹으로 만들어 각 그룹을 고급 프로그래머가 관리하게 함
- 경험자(고급 프로그래머)와 초보자를 구별함
- 프로젝트 리더와 고급 프로그래머에게 지휘 권한을 부여하고, 의사 교환은 초급 프로그래머와 고급 프로그래머로 분산함
- 기술 인력이 관리를 담당하게 되어 좋은 기술력을 사장시킬 수 있으며, 기술 인력이 업무 관리 능력을 갖춰야 한다는 단점

