

소규모 소프트웨어 업체를 위한 간략화된 프로젝트 관리 모델에 관한 연구

이정은 · 박주철[†]

울산대학교 산업정보경영공학부

Research on a Streamlined Software Project Management Model for Small-sized Software Enterprises

Jeong Eun Lee · Ju Chull Park

Department of Industrial Engineering, University of Ulsan, 680-749, Korea

Effective management of a project is a crucial issue to software companies. There are various international standards, such as CMM, ISO9000, and SPICE, applied to project management. But, small scale companies have some difficulties in accepting those standards because of its complexity and enormous manpower requirements.

This study proposes a project management approach streamlining the CMM by which a small scale software company can perform the tasks with ease. A prototype system has also been developed to show the capability of the proposed model.

Keyword: software project management, CMM, small-sized software enterprise, software process

1. 서론

소프트웨어 개발 프로젝트란, 고객의 요구사항에 맞추어 기능, 품질 등을 충족하고 정해진 기간 내에 소프트웨어 제품을 개발하고 인도하는 사업을 말한다. 과거의 소프트웨어 개발 프로젝트는 소수의 탁월한 능력을 가진 프로그래머에 의해 만들어지고 관리되었다. 그러나 소수의 인원만으로 개발하는 규모를 넘어서면서, 근래에는 소프트웨어 개발의 성패가 그 집단의 프로젝트 관리에 대한 숙달 정도에 크게 좌우되게 되었다. SEI(Software Engineering Institute)의 CMM(Capability Maturity Model)에 의하면 소프트웨어 개발 프로세스가 기관의 특성에 맞도록 잘 정의되고 교육 및 관리된다면 개발 과정이 효율적이 될 수 있다고 한다(Song and Lee, 2003). 이는 소프트웨어 프로젝트

관리 프로세스 정립의 중요성을 단적으로 보여주는 말이다. 하지만, CMM과 ISO9000, SPICE와 같은 국제적 표준안들은 내용이 방대하고 쉽게 접근하기가 어려우며, 또한 이를 수용하기 위해서 별도의 많은 인력과 노력이 소요되어 소규모 소프트웨어 업체에게는 적지 않은 부담을 줄 수밖에 없다.

본 연구에서는 프로젝트 계획에서 마감까지의 전 프로세스를 소규모 소프트웨어 업체의 실정에 맞게 간략화하고 효율화한 프로젝트 관리 모델을 제시 하고자 한다. 또한, 관리의 상당 부분을 정형화 하여 시스템으로 대체하고, 최소한의 입력으로 세부 관리항목의 보고서가 생성되어 관리가 가능해지도록 하는 정보 시스템을 활용 하는 접근 방법을 제안한다. 또, 연구된 관리 모델을 기반으로 프로토타입 시스템을 개발하여 연구된 내용이 실제 소규모 소프트웨어 업체에서 쓰일 수 있도록 한

본 연구는 울산대학교 연구비(2002-0187)에 의한 연구임.

[†]연락처 : 박주철 교수, 680-749 울산시 남구 무거동 산 29 울산대학교 산업정보경영공학부, Fax : 052-259-2180,

E-mail : jcpark@ulsan.ac.kr

2008년 02월 접수, 1회 수정 후 2008년 03월 게재확정.

다. 본 연구의 주요 연구 수행단계는 <그림 1>과 같다.

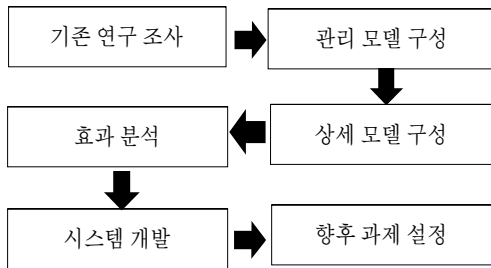


그림 1. 연구 수행 단계

2. 소프트웨어 프로젝트 관리의 일반적인 절차

일반적으로 프로젝트 관리란 프로젝트를 성공적으로 이끌기 위한 계획, 조직화, 인력확보, 지휘, 통제를 제공하기 위한 절차, 기법, 기술, 노하우 등으로 정의된다(Choi, 2001). 그중에서 소프트웨어 프로젝트 관리는 최종 산출물이 소프트웨어이고 합의된 시간과 예산 혹은 자원 내에서 사용자가 만족할 만한 소프트웨어 제품을 개발하는데 필요한 모든 기술적이며 관리적인 업무를 지칭한다(Choi, 2001).

소프트웨어 프로젝트 관리의 단계별 흐름은 계획, 개발, 종료로 나누어질 수 있다. 단계별 관리의 흐름 중 첫 번째인 계획 단계는 프로젝트를 수행하기에 앞서 누가, 무엇을, 언제, 어떻게 할 것인가를 결정하는 과정이다(Choi, 1999; Kim, 1999). 프로젝트 계획은 프로젝트 관리의 핵심이며, 필요한 작업의 종류와 작업의 수, 자원, 예상비용, 작업순서, 작업간의 의존관계, 예상기간, 전체 진행기간 같은 요소를 정확하게 추정해야 한다(Choi, 2003).

프로젝트 개발단계에서는 진도관리, 산출물 관리, 품질관리, 변경관리, 위험관리에 대한 사항들을 관리한다. 프로젝트 개발단계의 관리 목적은 계획한 바에 따라 프로젝트가 잘 수행되어 계획한 개발 기간 내에 계획된 품질로 개발하는 것이다.

프로젝트 종료단계에서는 미결사항에 대한 해결과 프로젝트 완료보고 후 프로젝트 성과물 인도 및 계약종료의 단계를 거치면서 프로젝트 종료를 수행한다. 약속한 품질의 소프트웨어를 약속한 기간내에 신속히 인도하는데 그 목표가 있다.

2.1 CMM을 활용한 프로젝트 관리

소프트웨어 개발 조직의 성숙도 모델인 CMM은 조직의 소프트웨어 프로젝트 관리를 포함한 소프트웨어 개발 전체 프로세스의 단계별 성숙 모형을 제시한다(Lee, 2003). 일반적인 소프트웨어 프로젝트 관리는 물론이고, 프로젝트 관리 방법의 문서화와 정량적인 프로세스 수준의 측정 및 지속적인 프로세스 개선을 포함한다(Park, 2001). CMM은 이러한 성숙도를 5개의 단계로 구조화하였고, 조직들은 이 단계를 통해서 지속적

인 프로세스 개선을 위한 성공적인 기반을 마련할 수 있도록 설계되었다(Song and Lee, 2003). 각 성숙도 레벨은 몇 개의 KPA(Key Process Areas)에 의해 구분된다. KPA는 조직이 해당 성숙도 레벨을 위한 프로세스를 평가할 때 반드시 초점을 두어야 할 영역을 말한다. CMM의 성숙도 레벨별 KPA는 <그림 2>와 같다.

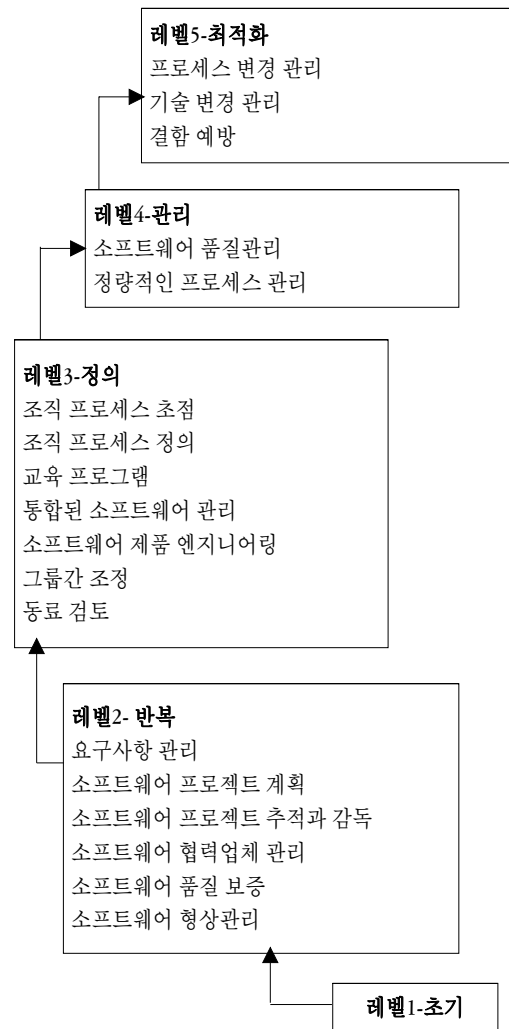


그림 2. CMM의 성숙도 레벨의 KPA

레벨 2에 있는 대부분의 KPA는 프로젝트 관리에 초점을 두고 있는 반면, 레벨 3의 KPA는 프로세스의 제도화와 소프트웨어 엔지니어링을 위한 추가적 프로세스를 목표로 한다. 그리고 레벨 4의 KPA는 프로세스와 프로젝트의 정량적인 관리에 초점을 맞추고 있으며, 마지막으로 레벨 5의 KPA는 결함예방과 기술 도입, 그리고 프로세스 확장을 통한 프로세스 개선에 초점을 두고 있다(Lee, 2004; Lee, 2005).

본 논문에서는 조직적인 측면보다 프로젝트 측면에서 CMM 레벨 4수준의 정량적인 프로젝트 관리가 되는 관리모형의 개발을 목표로 하고 있다. 일반적인 프로젝트 관리의 프로세스

를 확립하고 프로세스와 프로젝트의 정량적인 관리가 되도록 한다. CMM 레벨 4에 해당하는 프로세스를 구현하기 위해 수행 프로젝트 결과 데이터의 영구저장소인 PDB(Process Database)와 프로세스 수행 능력의 기준 척도인 PCB(Process Capability Baseline)를 사용하는데 이는 다음 장에서 설명한다. PDB는 CMM 레벨3의 주요 프로세스 영역인 조직 프로세스 초점과 조직 프로세스 정의에서 요구된다(Choi, 2001). 레벨 4에서 PDB는 프로세스를 분석하고 PCB를 계산하는데 쓰인다. PCB는 레벨 4의 주요 프로세스 영역인 정량적인 프로세스 관리와 소프트웨어 품질관리와 연관이 있다(Jalote, 2000).

2.2 PDB와 PCB

소프트웨어 프로젝트에서 PDB와 PCB는 프로젝트 계획과 관리에 유용하게 쓰일 과거의 경험들을 요약하기 위한 절차의 한 부분이다(Jalote, 2000).

PDB는 완료된 프로젝트의 측정값과 기록을 포함한다. PDB에 포함되어야 할 항목은 다음과 같다.

- 프로젝트의 특성
- 프로젝트의 일정
- 프로젝트의 공수
- 프로젝트의 규모
- 프로젝트의 결합
- 프로세스 자산: 다음 프로젝트에 쓰일 수 있는 산출물

PDB는 완료된 프로젝트의 마감분석으로부터 얻어진 프로세스 수행 데이터의 영구 저장소로서 프로젝트 생산성, 품질 등을 포함하며 다른 프로젝트의 계획, 추정, 분석에 쓰일 수 있다. 여기서 마감분석이란 프로젝트 수행 중 축적된 프로세스 자산을 종합해서 분석하는 활동을 말한다.

PDB가 프로젝트 각각에 초점을 맞춘 데이터를 포함하는데 반해, PCB는 정량적인 프로세스의 능력을 나타낸다. 프로세스의 능력은 프로세스를 수행 했을 때 예상되는 결과의 범위이다. 달리 말해서, 프로세스에 따라 프로젝트를 수행했을 때, 프로세스 능력은 프로젝트의 성공적 수행 확률을 결정하는데 쓰일 수 있다. PCB의 첫 번째 이슈는 어떤 타입의 척도들이 PCB에 포함되어야 하는지에 관한 것이다. PCB에 포함되어야 할 척도들은 다음과 같다(Jalote, 2000).

- 시스템 품질
- 생산성
- 일정 차이
- 공수 분포
- 결합 투입율
- 프로세스에서의 결합제거율
- 결합 분포

위의 정보들은 프로젝트 계획에 유용하게 쓰일 수 있다(Jalote, 2000).

PDB와 PCB는 완료된 프로젝트의 마감분석에서 만들어진 데이터를 기반으로 한다. <그림 3>은 PDB와 PCB사이의 연관성을 나타낸다(Jalote, 2000).

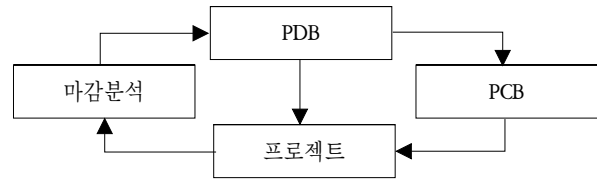


그림 3. PDB와 PCB

본 논문에서는 일반적인 소프트웨어 개발 프로젝트 라이프 사이클 프로세스를 종료한 후 마감분석을 통해 PDB를 구성하고 PCB를 산출함으로써 프로세스 개선 혹은 다음 프로젝트의 계획, 추정, 분석을 용이하게 하도록 한다. 또, 인력관리에 어려움이 많은 소규모 소프트웨어 업체의 실정에 맞게 프로젝트 개발 전체의 라이프사이클을 지원하고 PDB와 PCB 구성을 지원하는 프로토타입 시스템을 개발한다.

3. 소규모 소프트웨어 업체를 위한 프로젝트 관리 모델의 개발

소규모 소프트웨어 개발 업체에서의 프로젝트팀은 주로 개발을 담당하는 개발자들 중심으로 구성되고 관리와 개발을 병행하는 리더가 있어 부분적으로 프로젝트 관리를 담당하는 형태를 띤다. 프로젝트 관리력의 부족으로 효율적인 프로젝트 관리가 어려운 상황이다. 본 논문에서는 최소한의 관리와 시스템에 의한 관리를 통해서 개발 팀을 지원하는 관리 모델을 제시하고자 한다.

본 논문은 CMM의 레벨 4를 목표로 하는 프로젝트 관리 모델을 제시하고자 한다. 아울러 제시된 프로젝트 관리 모델을 지원하는 시스템 구성안을 제시하고자 한다. 이를 통해 본 논문에서 말하는 프로세스의 상당부분을 시스템이 자동으로 처리하도록 하고자 한다.

관리자는 시스템을 통해 적은 노력으로 전체 프로젝트의 세부 계획을 작성한다. 작성된 계획을 기반으로, 일별 작업 계획을 시스템에서 지원받아서 개발자들이 개발을 진행하게 된다. 시스템은 일일보고서, 산출물 정보 등을 관리자에게 제공함으로써 관리가 가능하다. 마지막으로 개발 단계에서 기존의 입력된 내용으로 구성된 PDB를 종료 단계에 넘겨주게 되면 PDB를 통해 분석된 PCB를 관리자에게 제공하고, 계획단계로 피드백 한다. <그림 4>는 본 논문의 프로젝트 관리 모델의 개념도이다.

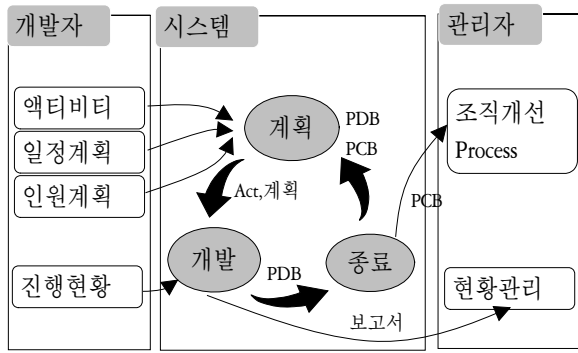


그림 4. 프로젝트 관리모델 개념도

개발단계에서 축적된 일일작업 실적으로 PDB를 형성하고 그 PDB로 종료단계에서는 PCB를 생성하고, 관리자는 PCB를 보고 조직개선 프로세스를 통해 조직의 성숙도를 높이도록 한다. 그와 더불어 PDB와 PCB로 다음 프로젝트의 예측정확도를 높이도록 한다.

3.1 계획 단계에서의 프로젝트 관리 프로세스

소프트웨어 개발 일정의 지연, 비용 초과, 품질저하, 유지비용 증가 등과 같은 문제는 계획의 부재에서 기인되는 경우가 많다(Choi, 2001). 개발 일정을 지키면서 품질 좋은 소프트웨어를 생산하기 위해서는 무엇보다도 세심한 계획이 필요하다.

소프트웨어 개발 과정을 계획하는 것은 소프트웨어가 개발되어 설치될 때까지 어떤 일들이, 누구에 의하여, 언제 행해져야 하는지를 정하는 것을 말한다. 일반적으로 말하는 계획은 다음과 같은 작업으로 구성된다(Choi, 2001).

- 문제를 이해하고 정의
- 필요한 액티비티를 정의하고 순서를 결정
- 일정 계획
- 조직 계획
- 요구분석
- 계획서 및 요구분석서 작성
- 산출물 계획

이상의 작업들은 보통 개발자들에 의해서 문서화 되어 관리된다. 본 논문의 관리 모델은 PDB와 PCB 및 표준 프로세스를 활용한 계획 지원기능을 강화해서 개발자들이 수월하게 정돈 높은 계획을 작성하도록 하는데 초점을 맞춘다. <그림 5>는 본 논문의 계획단계에서의 데이터 흐름을 나타낸다.

본 논문에서 제시하는 계획 단계의 업무절차는 다음과 같다.

- 단계 1: 기존 사례분석
시스템으로부터 기존 사례 분석 보고서를 조회해서 계획 기

본 자료로 활용한다.

- 단계 2: 프로젝트 생성

프로젝트 납기 및 명칭 등의 기본적인 특징을 입력하여 기본데이터를 생성하며 프로젝트를 정의한다.

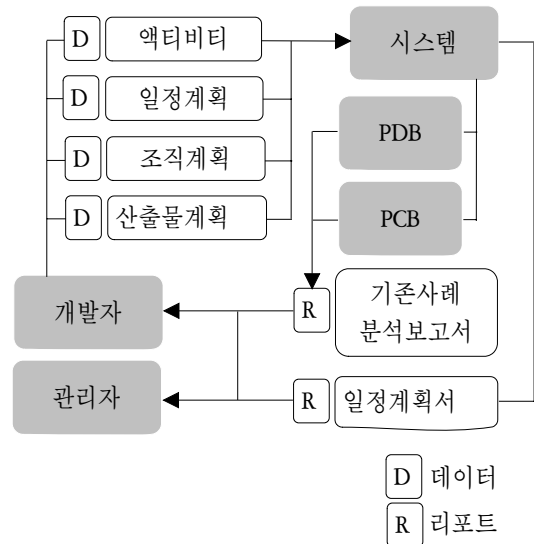


그림 5. 계획단계의 데이터 흐름

- 단계 3: 프로젝트 액티비티 정의

표준화된 액티비티를 바탕으로 프로젝트에 맞게 액티비티를 정의한다. 표준 액티비티를 가져오고 프로젝트에 맞게 새로운 액티비티를 정의한다. 또 정의된, 액티비티 마다 시스템에 입력 되어야 할 내용 외에 입력되어야 할 산출물을 정의한다. 정의된 액티비티의 순서를 지정한다. 액티비티 정의를 위한 각 단계를 지원하는 시스템 기능이 제공된다.

- 단계 4: 요구 사항 정의

요구사항은 개발자들에 의해서 정의되고 시스템에 입력된다.

- 단계 5: 프로젝트 일정 지정

앞서 정의된 프로젝트 액티비티를 기반으로 일정을 지정한다. 일정 계획은 시스템으로 지원되고, 액티비티별 예상 공기를 입력함으로써 계획이 이루어진다.

- 단계 6: 공수 계획

단계 5의 프로젝트 일정 지정과 마찬가지로 앞서 정의된 프로젝트 액티비티를 기반으로 공수를 지정한다. 공수 계획은 시스템으로 지원되고, 액티비티별 예상 공수를 입력함으로써 계획이 이루어진다.

- 단계 7: 조직 계획

기존 인원과 신규로 프로젝트에 투입되는 인원을 바탕으로

액티비티별로 투입될 인원을 지정한다. 지정된 내용을 시스템에 입력한다.

<그림 6>은 시스템 내의 업무처리 흐름이다. 기존 사례가 존재하지 않는 경우 기존의 결과를 사용할 수 없어 대부분의 기초 데이터를 계획단계에 입력을 할 필요가 있다.

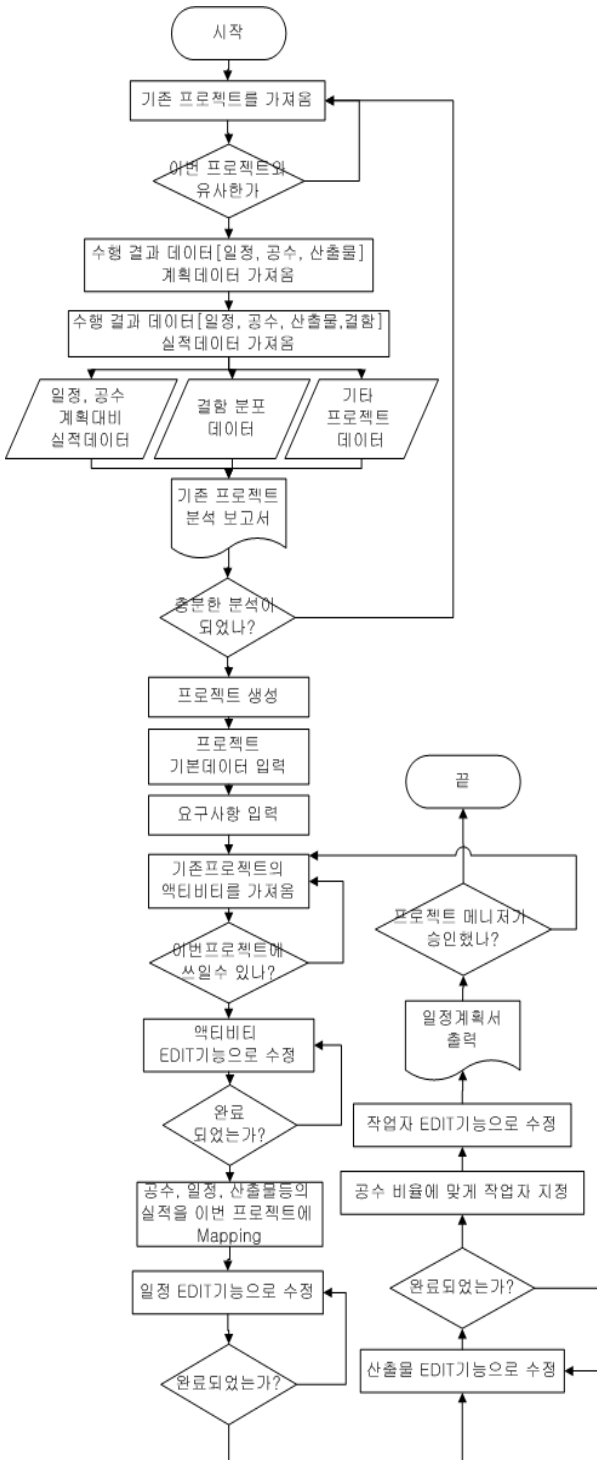


그림 6. 계획단계의 시스템 내 업무처리흐름

PDB와 PCB는 기존 사례분석 보고서의 형태로 표시가 되는데 기존 사례 분석보고서의 양식은 종료단계에서 제공될 마감 분석 보고서와 유사한 양식으로 기 수행한 프로젝트 별로 제공이 되도록 한다. <그림 7>은 기존사례분석 보고서의 양식 샘플이다. 그리고 기존사례 분석 보고서의 양식 중 세부 항목은 다시 그래프 형식으로 표시가 되도록 하여 자세한 분석이 가능하도록 한다. <그림 8>은 그의 한 예인 일정 분석서이다. 그림과 같은 보고서를 통해 프로젝트의 규모와 일정, 공수 분포 등을 참조해 좀더 정도 높은 계획을 할 수 있다. <그림 9>는 이러한 기존사례 분석을 통해 만들어진 신규 프로젝트의 일정계획서 샘플 양식이다.

기존사례 분석 보고서(일정분석)						
1. 일정						
계획 일정	0000년00월00일~0000년00월00일					
실제 일정	0000년00월00일~0000년00월00일					
1.1 일정차이 분석						
단계	착수			완료		
	계획	실적	차이	계획	실적	차이
요구사항분석						
상위수준설계						
상세설계						
코딩						
단위테스트						
통합테스트						
시스템테스트						
설치, 수락테스트						
프로젝트관리						
형상관리						
프로젝트교육						
기타						
총계						
1.2 공기분석						
단계	예측		실제		편차(%)	
	공기비율	공기	공기비율	공기		
요구분석						
설계						
빌드						
테스트						
수락						
관리						
교육						
기타						
총계						

그림 7. 기존사례 분석 보고서(일정분석)

3.2 개발 단계에서의 프로젝트 관리 프로세스

개발단계에서는 설계 및 실제 프로그램 개발이 이루어진다. 설계에서 소프트웨어 내부구조와 자료 구조를 결정하며 설계

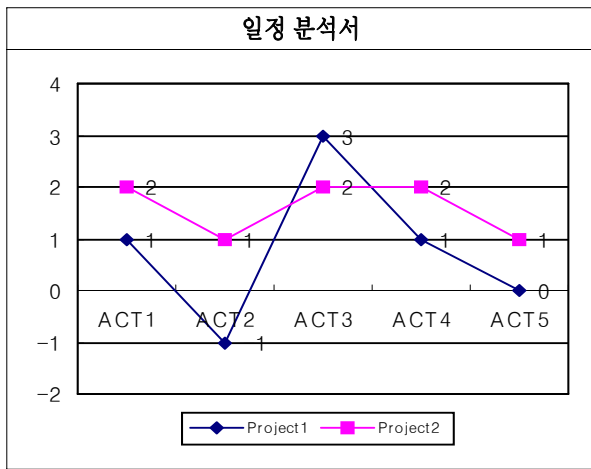


그림 8. 일정 분석서

NO	작업이름	2006/11				2006/12				2007/01				2007/02				2007/03				2007/04			
		WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT	WT				
1	00001-의 구축																								
2	1. 범위 분석 및 시스템 설계																								
3	1.1 요구사항 분석																								
4	요구사항 조사/분석																								
5	제시방안 결의																								
6	1.2 기본 설계																								
7	specification 작성																								
8	단위입력 흐름도																								
9	표준코드 및 규약 결의																								
10	수단입력 흐름도/결의서																								
11	1.3 상세설계																								
12	Entity-Relationship/결의서																								
13	ER Diagram																								
14	인터																								
15	데이터베이스/결의서																								
16	인용문헌과 결의서																								
17	2. 프로젝트 개발 및 실행관리																								
18	공통모듈 및 핵심기능																								
19	공통모듈 및 단위(P/S)개발 지원																								
20	문, 프로그램 개발(포장)																								
21	Data Migration/이동																								
22	단위테스트																								
23	3. 통합테스트 및 사용자 교육																								
24	사용자 인클로저 작성																								
25	통합테스트																								
26	사용자 교육 및 시행																								
27	시스템 인클로저 작성																								

그림 9. 일정 계획서

된 내용을 바탕으로 각 모듈에 대한 원시 코드를 작성한다 (Choi, 2001). 실제적으로 가장 많은 노력이 투입되는 단계이고, 관리가 가장 어려운 단계이다.

본 논문의 프로젝트 관리 프로세스에서는 개발자들에게 실행해야 할 액티비티의 항목을 알려주고, 실제 수행된 일의 양과 함께 각종 관리 데이터를 보고 하도록 함으로서 특별한 관리자 없이 시스템적으로 실제적인 관리가 되도록 한다. 개발자들은 시스템으로부터 당일 수행해야 할 액티비티를 부여 받고, 수행결과 및 코멘트 사항을 입력해서 관리가 되도록 한다. 개발자들이 입력해야 할 항목은 다음과 같다.

- 수행 액티비티 투입 공수
- 액티비티 작업 형태(작업, 검토, 제작업)
- 결함
- 산출물
- 수행 현황(%)
- 작업 특이사항 및 코멘트

위의 개발자가 입력하는 데이터로부터 시스템은 관리 리포트를 생성하여 준다. <그림 10>은 개발단계에서의 데이터 흐름을 나타낸다.

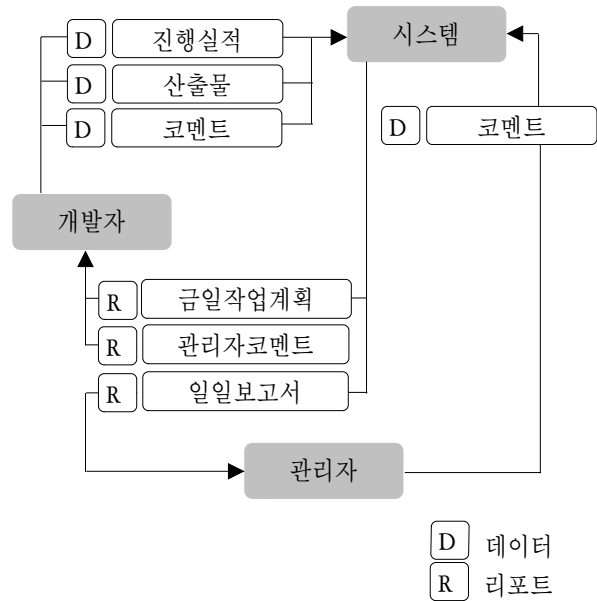


그림 10. 개발단계의 데이터 흐름

위와 같이 시스템으로 입력하는 항목을 바탕으로 관리자에 게 일일 보고서를 제공함으로써 프로젝트 관리가 최소한의 공수 투입으로 가능해진다. 관리가 가능한 항목은 다음과 같다.

(1) 진도 관리

시스템 내에서 일정 계획된 부분과 개발자가 입력한 실행 현황을 비교한다. 계획과 실제의 차이가 발생할 경우 연장 근무 및 추가인원 투입이 고려될 수 있고, 추가 공수 투입으로 계획 내 작업이 불가능 할 경우 계획변경을 고려한다.

(2) 형상 관리

액티비티별로 입력되어야 할 산출물 등이 시스템에 입력되어 관리가 된다.

(3) 위험관리

개발자들은 실행된 사항을 입력할 때 코멘트를 입력하는 부분에 일정 혹은 조직상의 위험이 예상되는 사항을 입력한다. 프로젝트 리더 혹은 프로젝트 관리자는 입력된 사항을 확인하여 적절한 조치를 취하여 관리를 한다.

(4) 품질관리

개발자들은 정의된 액티비티별 산출물을 입력하게 되는데, 각 산출물들은 프로젝트 리더 및 프로젝트 관리자에 의해 검사되고, 요구 분석에서 정의된 요구사항에 부합하는 내용인지 판단하여 재 작업, 수정, 합격이 결정되어 관리가 된다. 이는 코멘트 사항과 함께 시스템으로 저장 된다.

개발단계에서 작업된 사항과 관련된 생산성 정보, 품질 정보, 결함 분포도 등은 개발자가 일일보고를 통해 시스템에 입

력한 내용을 토대로 일일보고서로 정리된다. 일일보고 내용을 바탕으로 종료단계에서 수행할 마감 분석에서 사용될 데이터를 종료단계로 넘겨준다. <그림 11>은 시스템 내에서의 업무 흐름이다. <그림 12>는 산출물중 하나인 일일보고서의 예이다.

3.3 종료 단계에서의 프로젝트 관리 프로세스

종료 단계의 가장 중요한 부분은 프로젝트를 종료하고 난 후 축적된 데이터를 가지고 PDB를 생성하고 PCB를 산정한 후 PDB와 PCB를 기반으로 마감분석 보고서를 리포트 형식으로 개발자와 관리자에게 제공하여준다. <그림 13>은 종료단계에서의 데이터 흐름을 나타낸다.

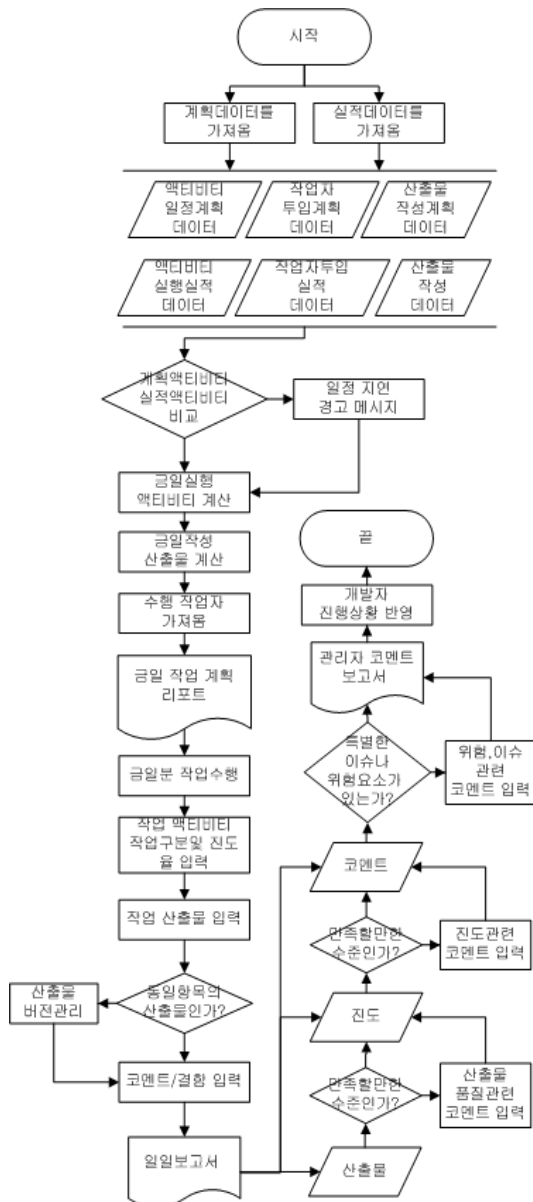


그림 11. 개발단계의 시스템 내 업무처리흐름

[일일 보고서]		0000년/00월/00일	
■ 현재실행 ACTIVITY			
계획	ACTIVITY1		
실적	ACTIVITY2		
■ 일정 [단위:%]			
전체	계획	<div style="width: 75%;"></div>	75
	실적	<div style="width: 72%;"></div>	72
현재실행 ACT	계획	<div style="width: 92%;"></div>	92
	실적	<div style="width: 89%;"></div>	89
■ 공수 [단위:M/H]			
전체	계획	<div style="width: 80%;"></div>	80/100
	실적	<div style="width: 75%;"></div>	75/100
현재실행 ACT	계획	<div style="width: 12%;"></div>	12/15
	실적	<div style="width: 10%;"></div>	10/15
■ 산출물			
금일계획 산출물	금일입력 산출물	버전	품질
PRODUCT1	PRODUCT1	1.0	양호
■ 결함			
금일발생결함	누적 결함수	제거된 결함	현재 결함수
0	3	3	1
■ 코멘트 사항			
MEMBER1	COMENT1		
MEMBER2	COMENT2		

그림 12. 일일 보고서 양식 샘플

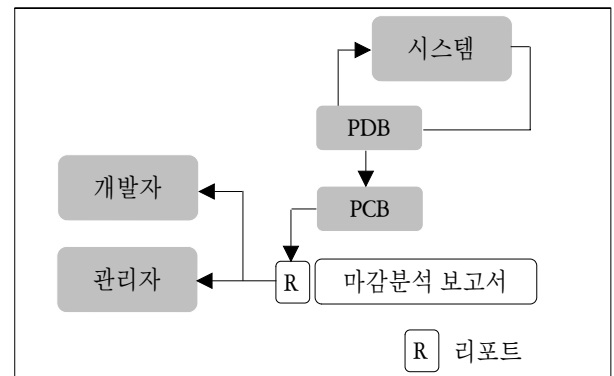


그림 13. 종료단계의 데이터 흐름

개발자와 관리자는 시스템에서 제공한 마감분석 보고서를 바탕으로 수행한 프로젝트 등에 대한 계획 오류나 개발기간 중의 공수 낭비 등을 체계적으로 분석하여 이를 다음 프로젝트의 계획부분에 반영하여 더욱 완벽한 업무수행이 되도록 한다. <그림 14>는 시스템 내의 업무흐름이다.

<그림 15>는 기존 연구에 나타나 있는 마감분석 보고서의 일부이며 본 연구에서는 이를 그대로 차용해서 쓰도록 한다.

3.4 제시된 관리모델의 특성 및 효과

본 논문의 프로젝트 관리 모델의 특징은 크게 조직차원의 특징, 프로세스 측면의 특징으로 나누어 볼 수 있다.

조직차원의 특징은 소규모 소프트웨어 업체의 실정에 맞게 프로젝트 관리자와 개발팀만으로 관리가 가능하다는 것이다. CMM 도입 시 요구되는 관리 조직을 시스템으로 대체함으로

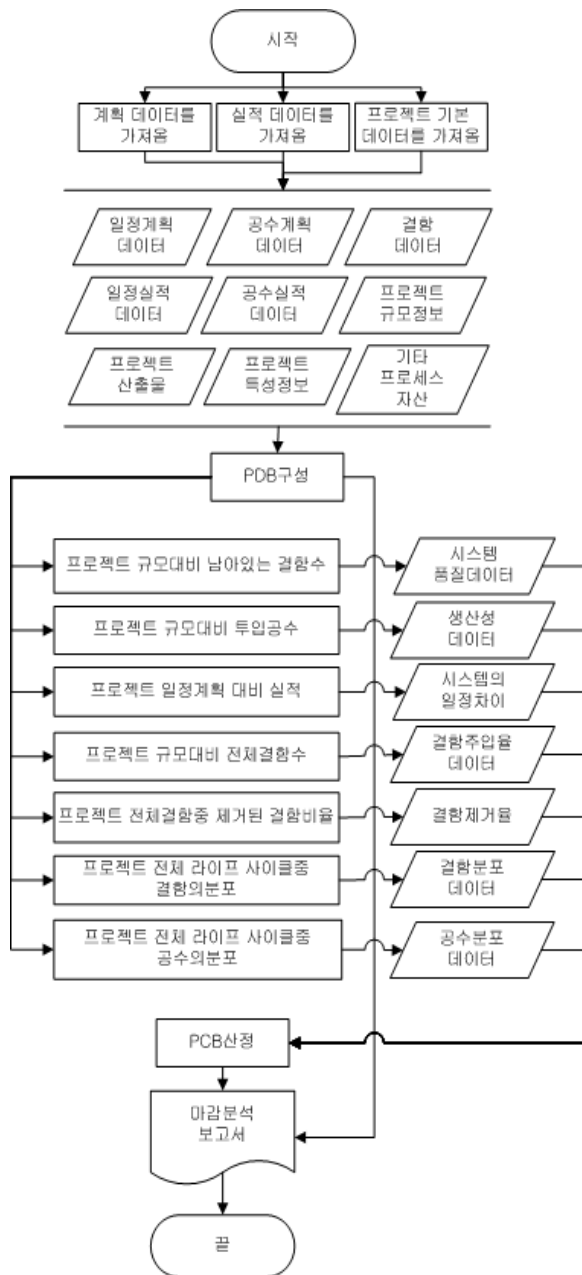


그림 14. 종료단계의 시스템 내 업무처리흐름

써 관리조직을 대폭 축소 할 수 있다. 프로세스 측면의 특징은 일반적인 소프트웨어 프로젝트를 포함하면서, PDB와 PCB를 활용하여 마감분석서와 기존 사례 분석 보고서를 통해 기 수행된 프로젝트의 종료 단계와 다음 프로젝트의 계획 단계를 이어주는 형태의 정량적인 관리가 가능해진다. 그리고 이를 통해 소규모 소프트웨어 업체에 맞춘 프로젝트 관리 부분에 국한된 CMM 레벨 4와 유사한 프로젝트 관리가 가능해진다. 달성 가능한 CMM의 레벨별 KPA는 <표 1>과 같다.

제시된 프로젝트 관리 모델은 문서 작성 공수의 일부분과 관리와 지원공수 대부분을 절약 할 수 있다. <표 2>는 규모와 그에 해당하는 공수의 비율을 나타낸 표이다(Jones, 1998).

마감 분석 보고서(공수분석)					
1.공수					
예측공수	22주*(6일*8시간*3.5명=369(인*시))				
실제공수	8639(인*시)				
1.1 라이프 사이클 단계의 분포					
단 계	테 스 크	검 토	재작업	총 계	
요구사항분석	396	160	12	567	
상위수준설계	449	9.6	70	528.6	
상세설계	477.3	15.4	9	501.4	
코딩	3405	456	614	4475	
단위테스트	228	62	36	326	
통합테스트	88	15	0	103	
시스템테스트	0	0	0	0	
설치, 수락테스트	765	8	24	797	
프로젝트관리	585	0	0	585	
형상관리	20	0	0	20	
프로젝트교육	104	0	0	104	
기타	592	40	0	632	
총계	7108.3	766	765	8693	
1.2 품질비용					
COQ=(검토공수+제작업공수+테스트공수+교육공수)/총공수					
=[(766+765+423+104)/8619]*100 = 23.8%					
1.3 공수분포					
단 계	예 측		실 제		편차(%)
	공수비율	공 수	공수비율	공 수	
요구분석	6.8	31.4	6.6	70.9	125.8
설계	9.0	41.6	11.9	128.8	209.6
빌드	50.0	231	51.8	559.4	142.2
테스트	13.0	60.1	5.0	53.6	-10.8
수락	6.8	31.4	9.2	99.6	217.2
관리	9.0	41.6	7.0	75.6	81.7
교육	2.2	10.2	1.2	13	27.5
기타	2.2	10.2	7.3	79	674.5
총계	100	457.4	100	1079.9	136

그림 15. 마감 분석 보고서(공수 분석)(Chol, 2003)

표 1. 달성된 레벨별 KPA

구 분	달성 KPA	미달성 KPA
레벨 2	요구사항 관리 소프트웨어 프로젝트 계획 소프트웨어 프로젝트 추적과 감독 소프트웨어 품질보증 소프트웨어 형상관리	소프트웨어 협력업체 관리
레벨 3	통합된 소프트웨어 관리 동료검토	조직 프로세스 초점 조직 프로세스 정의 교육 프로그램 소프트웨어 제품 엔지니어링 그룹간 조정
레벨 4	소프트웨어 품질관리 정량적인 프로세스 관리	
레벨 5		프로세스 변경 관리 기술 변경 관리 결함 예방

표 2. 규모별 관리공수

규모 (기능점수: FP)	공수의 비율(%)				코딩 이외의 비율	비중
	코딩	문서작성	결함제거	관리와 지원		
1	70	5	15	10	30	1
10	15	7	17	11	35	↓
100	54	35	20	11	46	↓
1000	30	26	30	14	70	↓
10000	18	31	35	16	82	27

PDB와 PCB를 활용한 정량적인 관리가 가능해 짐으로서 계획의 정확도 향상을 통한 효과와 관리 및 지원을 시스템으로 처리하는 본 프로세스의 특징으로 볼 때 관리와 지원공수와 문서작성 공수의 일부분, 약 10% 정도의 공수 절감 효과가 있을 것으로 예상된다. 또한, 관리가 되지 않고 있던 영세한 소규모 소프트웨어 업체의 관리프로세스 지원으로 산출물의 품질 향상 및 성공적 프로세스 수행능력 향상 등으로 이어질 수 있을 것으로 생각된다.

4. 프로토타입 시스템의 개발

개발된 프로세스의 가능성을 알아보기 위해 개발된 프로세스의 일부분을 포함하고, 일일보고서로 데이터를 축적해서 PDB생성 및 PCB산정을 지원하는 프로토타입 시스템을 개발 하였다.

본 시스템의 핵심은 개발자들의 일일 현황을 미리 정의된 시스템의 현황 폼에 입력함으로써 적은 노력을 들여서 요구하는 관리영역의 관리가 가능하도록 하는데 있다. 본 시스템은 MS SQL 2000과 ASP를 기반으로 WEB환경에서 개발되었다.

4.1 프로젝트 관리 시스템의 기능구조

본 논문에서 제공하는 시스템은 프로젝트 관리의 라이프 사이클인 계획, 개발, 종료로 크게 나누어져 있고, 각 단계별 세부 항목을 관리하도록 하였다. <그림 16>은 시스템의 기능구조도이다.

프로젝트 속성 관리 기능은 프로젝트를 생성하고 기본적인 프로젝트 속성을 입력한다.

프로젝트 계획관리의 액티비티 생성 기능에서 액티비티를 표준 액티비티를 바탕으로 액티비티를 생성하고 순서를 지정한다. 입력된 액티비티를 바탕으로 일정을 일정관리 기능에서 계획 하고, 액티비티별 상세 입력 정보를 입력관리 기능에서 입력한다.

프로젝트 개발관리에서는 실제 프로젝트의 개발부분을 지원하게 된다. 금일의 액티비티 실시상황을 입력해서 프로젝트 관리에서 요구하는 진도관리, 위험관리, 품질관리, 형상관리 등을 진행 현황분석기능에서 지원한다.

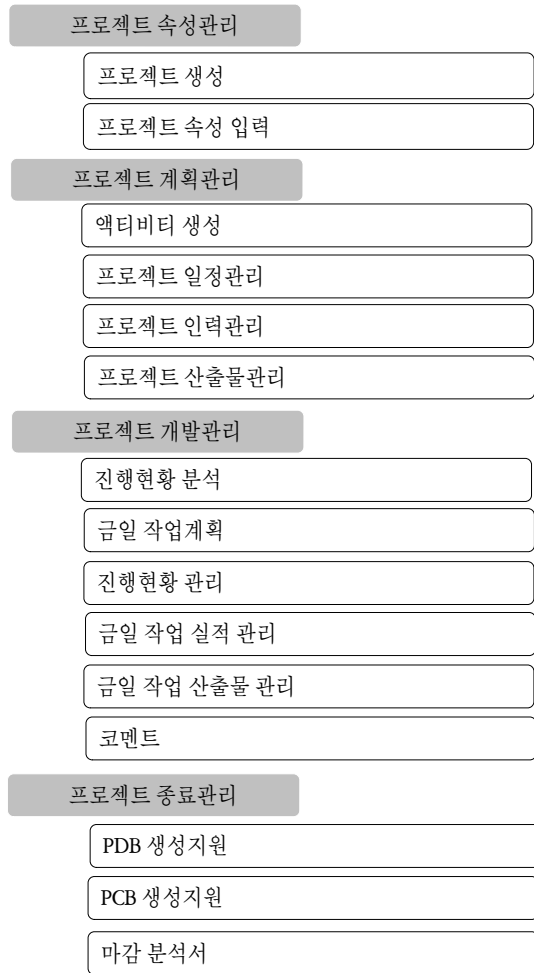


그림 16. 기능구조도

프로젝트 종료관리에서는 개발을 완료한 상태에서 단위 테스트 및 통합 테스트를 계획 및 실행을 지원하고, PDB생성지원 기능에서 PDB를 생성하고 PCB 생성 지원 기능에서 PCB를 산정 하고 마감 분석서 기능에서 마감 분석서를 출력함으로써 프로젝트가 종료된다.

4.2 프로젝트 관리 시스템의 개발결과

프로젝트 관리 시스템은 계획, 개발, 종료의 각 단계를 지원하며, 사용자 편의성 보다는 기능구현에 중점을 두었다. 대표적인 기능은 진행현황 분석, 진행현황 관리, PCB 생성지원 기능 등이 있다. <그림 17>은 로그인 후 처음 실행되는 진행현황 분석 화면으로 일자별 실행해야 할 액티비티와 금일 실행해야 할 액티비티 그리고 현재 실행 중인 액티비티의 현황정보가 나타나게 된다.

<그림 18>은 진행현황 관리 화면으로 일일보고, 산출물, 코멘트의 내용을 보여주는 화면이다. 본 기능의 일일보고서 확인, 산출물 확인을 통해 지시 사항을 작업자별로 남김으로서 진행 현황의 관리가 가능하다.

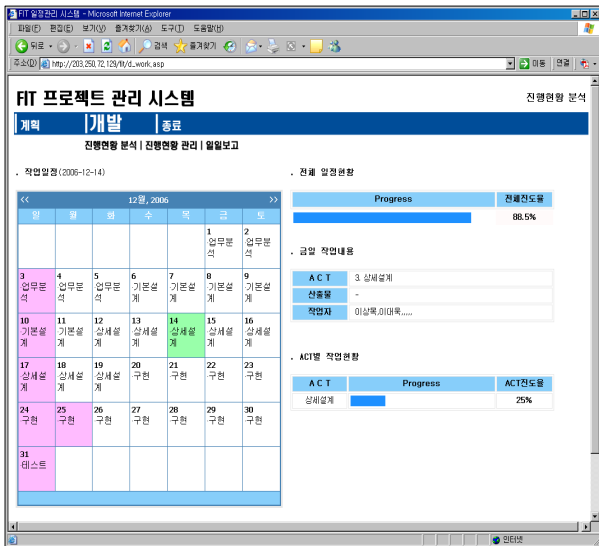


그림 17. 진행현황 분석 화면

<그림 19>는 PCB 생성지원 화면이다. 프로젝트 종료 후 PDB와 PCB를 생성해서 조직의 성숙도 수준을 알아 볼 수 있고, 이것을 바탕으로 다음 프로젝트의 계획에 생성 결과를 반영할 수 있다.

본 논문에서는 일반적인 프로젝트 관리를 포함하고 PDB와 PCB의 도입을 통한 CMM 4단계의 관리수준을 달성할 수 있는 관리 모델을 개발하고, 이를 지원하는 프로토타입 시스템을 개발 하였다. 프로세스와 시스템 측면에서 기본적인 골격은 갖추었지만 추가 되어야 할 부분이 더 있다.

프로세스 측면에서 개발자가 입력해야 할 데이터의 양을 표준 액티비티, 표준 산출물과 같은 표준 데이터 도입을 통해 줄여야 한다. 또, PDB와 PCB를 통한 표준 데이터 개선 프로세스의 추가가 필요하다.

시스템 측면에서 GUI기능의 강화를 통한 계획의 용이성 확보가 필요하고, 통계분석 기능의 강화가 필요하다.

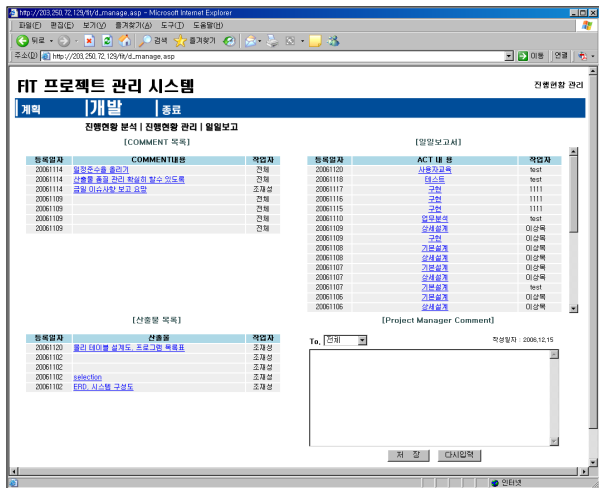


그림 18. 진행현황 관리 화면

5. 결론

소규모 소프트웨어 업체는 작은 조직과 과도한 업무 부담 때문에 프로젝트 관리가 잘 이루어지고 있지 않은 실정이다. 프로젝트 관리의 중요성은 알고 있지만 그에 맞는 적절한 모델이 없었던 것 또한 사실이다. 본 논문은 소규모 소프트웨어 업체의 실정에 맞게 프로젝트 관리의 여러 영역 중 프로젝트 실행 관리에 초점을 맞추어 관리영역을 한정하고, 각각의 추진해야 할 액티비티를 정의하고 시스템을 통한 관리가 가능한 관리 모델을 제시 하였다. 그리고 CMM의 레벨 4의 KPA인 정량적인 관리를 도입하여 프로세스를 모델링 하였다. 계획, 개발, 종료의 프로젝트 관리단계에서 끝나는 것이 아니라 종료 단계에서 다시 다음 프로젝트의 계획으로 이어지는 고리를 만들어 줌으로써 지속적인 개선과, 조직의 성숙도 향상에 기여할 수 있다. 마지막으로, 제시된 프로세스를 일정부분 수용하는 프로토타입 시스템을 개발하여 프로세스의 적용성을 높이고자 하였다. 시스템 도입을 통한 관리공수의 절감과 CMM의 프로젝트 관리 프로세스 도입으로 소규모 소프트웨어 업체의 조직 성숙도 향상과 관리능력 향상에 도움을 줄 수 있을 것으로 판단된다.

표준 데이터 도입과 GUI기능 강화 등의 개선사항들을 추가 연구하고, 이의 실제 적용을 통한 보완이 이루어진다면 더욱 향상된 프로젝트 관리 프로세스 모델이 구성될 수 있을 것이다.

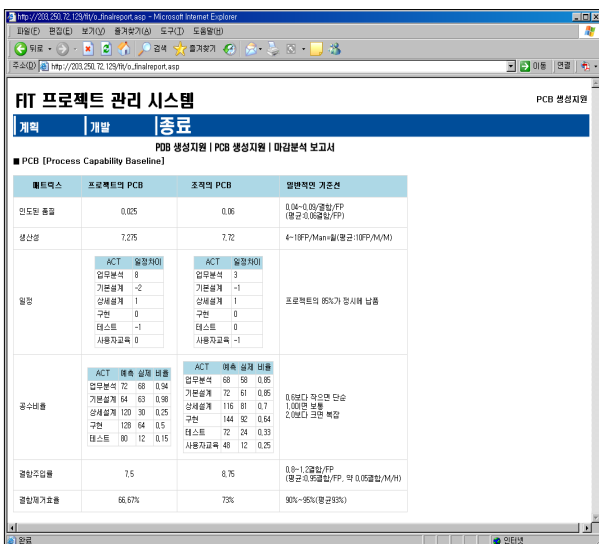


그림 19. PCB 생성지원 화면

참고문헌

Ahn, J. M. (1999), An Integrated Perspective on Software Development Project Management and Information Systems Performance, *The Kon-Kuk Journal of Business and Economic Studies*, 24(1), 220-230.

- Choi, S. H. (1999), Study of Software Development Process Management Model in Small-size Software Development Project, A Master's Thesis, Sogang University, 4-49.
- Choi, E. M. (2001), Software Engineering, Scitech Media, A Revised Edition, South Korea, 2-295.
- Choi, J. T. (2003), A Study on the Project Closure Process Model Based on CMM for a Small Scale Information Technology Enterprise, A Master's Thesis, University of Ulsan, 4-28.
- Kim, Y. M. (1999), Study of Software Development Process Management Model in Small-size Software Project, A Master's Thesis, Dong Kuk University, 12-55.
- Kim, H. S. (1999), A Study on Productivity Management for System Integration Projects, *The Journal of Information Technology*, 5, 81-92.
- Lee, J. B. (2004), Study on Process Improvement by Quantitative Project Management Based on CMM, A Master's Thesis, Yonsei University, 3-44.
- Lee, D. H. (2005), Study on Project Management Methodology Based on CMM and PMBOK, A Master's Thesis, Yonsei University, 4-21.
- Lee, B. L. (2001), A Quality Assessment Model of Software Development Process, *Research of Industrial Management*, 10(1), 203-237.
- Lee, S. W. (2003), The Strategy for Improving Software Process Maturity using CMM, A Master's Thesis, KAIST, 6-30.
- Jalote, Pankaj (2000), CMM in Practice, Soft Engineering Institute, USA, 97-117.
- Jones, T. Capers (1998), Estimating Software Costs, McGraw-Hill, 140.
- Park, S. H. (2001), A Study on CMMI, *Research of Information Science*, 1, 45-55.
- Song, T. K. and Lee, B. H. (2003), The Capability Maturity Model, Pearson Education Korea, South Korea, 3-102.



이정은

울산대학교 산업정보경영공학과 학사
 울산대학교 산업정보경영공학과 석사
 현재: 현대중공업 조선사업기획부
 관심분야: 생산공정최적화, 생산시스템



박주철

서울대학교 산업공학 학사
 KAIST 산업공학 석사 및 박사
 현재: 울산대학교 산업정보경영공학부 교수
 관심분야: 경제성공학, 조선생산정보시스템