

테스트 단계 감리시스템 설계 및 구현

권 대 곤[†] · 한 판 암^{††}

요 약

소프트웨어 개발과 유지 보수 활동에서 감리는 소프트웨어의 품질을 좌우할 수 있는 중요한 활동이다. 특히 테스트 단계에서 철저한 감리가 이루어져야 한다. 지금까지 감리는 감리자의 소프트웨어 개발 경험이나 감리 경험에 의존하고 있으며, 감리 항목 기준이 예매하여 객관적으로 감리할 수 없었다. 더구나 감리 활동 자체가 시스템화 되어 있지 않아 조직적이고 객관적인 감리에는 한계가 있었다.

본 논문에서는 기존 감리 활동의 제반 문제점을 해결하기 위하여 감리시스템의 모델을 제안하고, 소프트웨어 테스트 단계에서의 감리활동을 시스템화 한 감리시스템을 개발하여 실제 감리 활동에 적용해 보았다. 구현 적용한 결과 소프트웨어 관리자는 효율적인 소프트웨어 권리지침을 수립할 수 있고, 소프트웨어 개발자에게는 신뢰성 있는 고품질의 소프트웨어 개발 도구를 지원하며, 소프트웨어 감리자에게는 객관성 있는 감리지침을 제공할 수 있었다.

The Design and Implementation of Test Phase Audit System

Kweon-Dae Gon[†] · Han-Pan Am^{††}

ABSTRACT

Auditing plays a very important role in the process of developing and managing good quality software. The software developing process should be audited precisely especially in the test phase. Up to the present, because auditing has depended on the auditor's experience of developing and auditing software, it has been impossible to audit objectively. It is limited to audit systematically and objectively because auditing process isn't systematized.

In this paper, the auditing model to solve several problems in present auditing is suggested, a test phase audit system is developed, and the system is applied to the actual auditing process. Consequently, software administrators can establish effective software management, software developers can be supported by a highly reliable and quality software development tool, and auditors can be offered an objective audit standard.

1. 서 론

오늘날 공공 부문과 기업 부문에서 조직의 생산성 증대 및 경쟁력 향상을 위해 소프트웨어의 역할이 계속 증대되고 있다. 소프트웨어가 조직의 목표 달성을 위해 그 역할을 충분히 수행하고 있는지를 진단하고

평가하며, 이러한 소프트웨어를 개발하고 관리하는 제반 활동들이 효율적으로 수행되고 있는지를 점검·확인할 필요가 있다. 즉, 소프트웨어에 대한 진단 및 감리(Audit)의 역할 확대에 따른 대처 방안이 다양하게 요구되어야 한다. 그렇지만 외국의 경우에는 시스템상의 시큐리티, 회계감사, 소프트웨어 결함 측정 등에 자동화된 도구로써 존재할 뿐 소프트웨어 감리를 위한 시스템은 발견하기 어렵다[12-15]. 더구나 국내에서는 소프트웨어에 대한 감리 관련 문헌 또는 연구 보고서는

† 경희대학교 컴퓨터공학과 교수

†† 중신회원 강남대학교 컴퓨터공학과 교수
논문접수: 2000년 1월 19일, 심사완료: 2000년 4월 28일

감리에 대한 원론적 수준을 벗어나지 못하고 있어, 실제 감리를 수행할 실무자 입장에서는 별로 도움이 되지 못하는 실정이다. 따라서 본 연구의 필요성을 다음과 같이 제기할 수 있다.

첫째, 시스템 개발 수명 주기(System Development Life Cycle : SDLC)별 감리 지침은 소단위별 질문 형식이 유사한 경우가 많다. 이에 대한 해결책은 좀더 다양한 질문 형식을 개발하는 것이다.

둘째, 시스템 개발 감리 지침의 질문 범위가 너무 포괄적이다. 이에 대한 해결책은 현실적이고 실무적인 내용이 되도록 좀더 세분화하고 현실을 감안한 내용으로 보강이 필요하다.

셋째, 국내의 소프트웨어 환경과 감리 지침의 이론적인 부분이 상당한 차이가 있다. 실질적인 감리를 위해서는 소프트웨어 운영환경과 사용자 수준등 제반 여건이 감리 기준에 포함되어야 한다.

넷째, 각 시스템 수명 주기별 감리 지침만 있을 뿐 무엇을 어떻게 해야 하는지 구체적인 소프트웨어 감리 방법론이 없으며, 이에 대한 체계적인 이론 정립이 요구된다.

결론적으로 본 연구는 민간 감리의 확대에 대비하여 기존의 감리 기준을 시스템 사용 환경에 따라 현실화·구체화하고, 감리 지침에 대해 산재되어 있는 지식을 체계적인 조사 방법 및 통계적 모델링 기법을 이용 정리 및 분류하여 시스템 사용 및 사용자 환경을 고려한 최적의 소프트웨어 감리 기준을 자동화된 도구 형태로 제공하는데 그 목적이 있다. 특히 본 연구에서는 소프트웨어 개발 단계 중에서 테스트 단계 감리에 국한한다

2. 소프트웨어의 감리 개념

우리의 소프트웨어 감리는 미국, 일본 등에서 사용하는 소프트웨어 감사와는 개념적으로 많은 차이가 있다.

미국의 ISACA[19]에서는 “시스템이나 구성 부분이 신뢰성 있고 정확한 정보를 산출하는지를 평가하고 그 정보가 경영자의 요구사항이나 준수해야 할 규정에 합치하는지를 개발 유지보수·운영 단계에서 검토 및 평가하는 것”이라고 정의한다.

일본의 시스템 감사 기준[18]에서는 “종합적으로 점검, 평가하고, 관계자에게 조언, 권고하는 것이고, 보안 대책의 실효성 담보 및 시스템의 유효성의 이용을 도

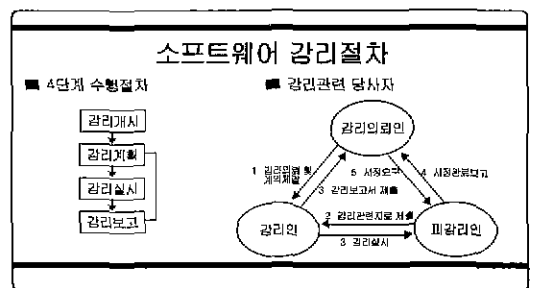
모하는데 있어서 효과적인 수단”으로 언급한다.

국내에서 감리 전문가인 김궁현[22]은 소프트웨어 감사를 “독립성을 가진 감사인이 감사 증거를 수집, 평가하여 관계자에게 보고하는 행위”로 정의하고, 김혁[21]은 “정보 처리의 신뢰성, 소프트웨어의 안전성 및 효율성의 추구하고 소프트웨어의 피해 방지를 위해서 수행하는 감사 활동”으로 정의한다

따라서 본 논문에서는 소프트웨어 감리를 “소프트웨어의 계획 개발 구축·운영의 제반 활동이 효율적이고 효과적으로 수행되며 편리되고 있는지 기술적 및 관리적 전문 지식을 가지고 평가하고 자문하는 활동”으로 정의한다[16].

2.1 소프트웨어의 감리 절차

소프트웨어 감리 절차[18]는 감리 의뢰 기관의 감리 문의와 감리 의뢰의 접수로부터 감리 보고서가 작성되고 통보될 때까지의 절차를 의미한다. 감리 절차 및 감리 관련 당사자간의 관계를 (그림 1)과 같이 나타낼 수 있다.



(그림 1) 소프트웨어 감리 절차

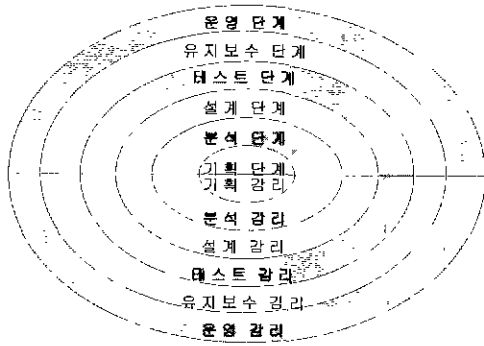
감리개시는 자발적인 감리 요청에 의해 프로젝트 초기에 참여하여 실시한다. 감리 계획은 본 감리의 수행에 소요되는 자원을 파악하고, 감리자별 업무의 체계적 분장으로 감리 업무 수행의 표준화와 각 기관별 역할 규정 및 감리자별 점검 항목을 규정한다. 감리 실시는 감리자가 현장에서 실사와 관계자 면담을 통하여 소프트웨어 준거성 여부를 파악하고, 중점검토 항목별 문제점 등을 확인하여 개선 사항을 작성한다. 감리보고는 감리 보고서 작성[3, 19], 감리 보고서 통보 활동으로 구분할 수 있다. 감리 보고서의 통보는 감리 수행 기관의 명의로 통보되며, 감리자 성명, 발행 일자를 반드시 기입한다. 감리 보고서 배포 기관은 감리 의뢰

기관 및 피감리 기관이 되며, 감리 보고서는 외부로 유출되지 않도록 보안을 유지한다.

2.2 소프트웨어의 감리 분류

소프트웨어의 감리는 크게 운영 감리와 사업 감리로 구분된다[5] 운영 감리는 소프트웨어 운영의 효율성, 효과성, 안전성 및 데이터 무결성 등을 주안점으로 하여 실시하며, 복합년 감리의 형태를 취하며 감리의 주기도 정기적으로 실시되어야 한다.

사업 감리는 (그림 2)와 같이 시스템 개발 수명 주기를 기준으로 구분한다



(그림 2) 소프트웨어 감리 단계

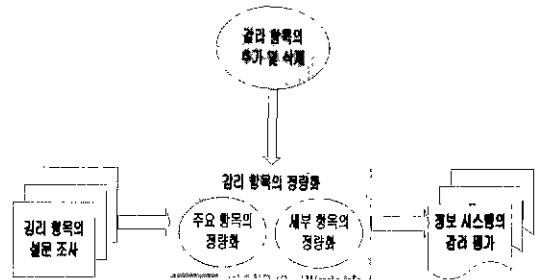
본 연구에서는 소프트웨어 개발 단계 중에서 테스트 단계에서의 테스트 감리 지침에 관한 연구로 국한한다.

2.3 소프트웨어 테스트

국제 표준인 ISO/IEC 12207[2, 6, 21]에서는 개발 활동 중 테스트에 관련된 활동을 소프트웨어 구축 및 단위 테스트 활동, 소프트웨어 통합 테스트 활동, 소프트웨어 자작 요건 테스트, 시스템 통합 테스트 활동, 시스템 자작 요건 테스트 등 5가지로 구분한다.

3. 감리시스템 모델 제안

본 논문에서는 소프트웨어 개발 단계 중에 테스트 단계의 테스트 감리 지침을 구체화하여 정량적인 평가 기준을 수립하고, 소프트웨어 개발에 관련된 관리자, 개발자, 감리자에게 유용한 평가 도구를 제공하는 고품질의 신뢰성 있는 감리시스템의 모델을 (그림 3)과 같이 제안한다.



(그림 3) 감리시스템의 모델

소프트웨어의 테스트 감리 기준 조사서의 테스트 단계는 크게 다섯 부분으로 나누어지며, 각 부분별로 총 35개의 주요 항목을 두고 주요 항목 내에 모두 114개의 세부 항목으로 구성된다 본 연구에서 사용한 설문지는 테스트 감리 기준 조사서를 기본으로 하였다.

3.1 설문 조사의 내용

설문 조사의 대상은 크게 두 그룹으로 구분하였다. 먼저, 세부 항목 설문지의 조사 대상은 전국에 있는 정보시스템 감리사와 주요 항목 설문지는 정보 처리 기술사 협회 소속의 회원을 각각 선정하여 분석하였으며, 그 세부 내용은 <표 1>과 같다

<표 1> 설문 조사의 내용

구분	세부항목 설문지	주요항목 설문지
조사 기간	99년 7월~10월	99년 7월~10월
조사 대상	정보시스템 감리사	정보처리 기술사
조사 방법	우편 설문 조사법	좌담
설문 발송	126부	250부
설문 응답	반 송	18명
	미응답	63명
	응답	45명
		37명
		127명
		86명

3.2 감리 항목의 정량화

3.2.1 주요 항목의 정량화

소프트웨어의 감리 항목에 대해서 35개의 주요 항목 정량화를 위해 주요 항목별로 가중치를 <표 2>와 같이 부여한다.

<표 2> 주요 항목에 대한 중요도별 가중치

응답내용	중요도 낮음			중요도 높음	
	①	②	③	④	⑤
배 점	2	4	6	8	10

<표 3>은 주요 항목 점수 기준으로서 주요 항목 1 개에 대해 86명 모두가 중요도 ⑤를 선택했을 경우를 기본으로 주요 항목 35개의 기준 점수를 계산한 결과이다. <표 3>에서 주요 항목 기준 점수를 좀더 일반화된 수식으로 유도하기 위해 i 번째 주요 항목의 변수와 가중치를 a_i, f_i 라고 했을 때 주요 항목 k 에 대한 점수는 식 (1)과 같이 정의할 수 있다.

<표 3> 주요 항목의 기준 점수

항목 번호	중요도	중요도	중요도	중요도	중요도	기준 점수
	① 2점	② 4점	③ 6점	④ 8점	⑤ 10점	
1	1	9	13	21	42	82
2	3	2	10	30	11	81
3	6	14	29	21	16	66
4	-	9	28	28	21	74
5	5	14	17	35	15	70
⋮	⋮	⋮	⋮	⋮	⋮	⋮
31	-	16	25	25	20	71
32	-	16	20	30	20	73
33	3	3	21	28	31	79
34	13	19	10	21	23	65
35	5	12	29	26	14	67

$$ST_k = \sum a_i \cdot f_i \quad (1)$$

여기서, 주요 항목별 점수를 백분율로 환산하면 수식 (2)와 같이 표현할 수 있으며, 여기서 ST_0 은 주요 항목 기준 값이다.

$$st_k = \frac{ST_k}{ST_0} \times 100 \quad (2)$$

3.2.2 세부 항목의 경량화

세부 항목의 계산 점수를 구하기 위해 위의 수식에 서 결정된 각 주요 항목 점수를 기준으로 세부 항목의 누계 점수에 따라 분산시킨다.

<표 4>는 세부 항목에 대한 점수 계산의 예를 나타 내었다. <표 4>를 일반식으로 유도하기 위해 중요도 변수를 a , 가중치 변수를 f , 중요도 응답변수를 i , 세 부 항목 번호를 j , 주요 항목을 k 라고 하면 주요 항목 에 대한 세부 항목의 누계점수 계산식은 식 (3)과 같 이 나타낼 수 있다

$$ST_{1,j} = \sum a_i \cdot f_i \quad (3)$$

한편, 세부 항목별 계산 점수의 표현식은 식 (4)와 같이 나타낼 수 있다.

$$V_{1j} = \frac{ST_{1,j}}{\sum ST_{1,j}} \times ST_1 \quad (4)$$

따라서, 세부 항목의 정량화된 일반식은 식 (5)와 같 이 나타낼 수 있다.

$$V_k = \frac{ST_{k,j}}{\sum ST_{k,j}} \times ST_k \quad (5)$$

3.3 감리 항목의 추가 및 삭제

본 논문에서 제안하는 감리시스템의 모델은 미리 정해진 감리 평가 항목으로 인해 발생할 수 있는 객관성 결여뿐만 아니라, 특정한 소프트웨어의 개발에 적합하지 않을 수 있는 문제를 보완하였다.

이를 위해 주요 항목 및 세부 항목에 대해 특정한 소프트웨어에 적합한 감리 평가를 할 수 있도록 하는 감리 항목의 추가 및 삭제 기능을 포함시켜 주요 항목의 추가 및 삭제 후 변경될 점수를 계산할 수 있도록 하였다

3.4 소프트웨어의 감리 평가

감리시스템에서 제공되는 세부 항목의 평가 결과는 우수, 양호, 보통, 미흡, 부적정등으로 평가한다 따라서 세부 항목의 평가 결과를 바탕으로 주요 항목별로 개선 권고 사항을 부여하는데, 그 기준은 <표 5>와 같다.

<표 4> 세부 항목에 대한 점수 계산의 예

구 분	세부 항목	중요도①	중요도②	중요도③	중요도④	중요도⑤	누계 점수	계산 점수	주요 항목 기준 점수
		2점	4점	6점	8점	10점			
주요 항목1	1	2		5	11	27	392	(392/1474) 82	82
	2		16		16	13	322	(322/1474)*82	
	3		3	10	11	21	370	(370/1474)*82	
	4			12	6	27	390	(390/1474)*82	
	가중치	2	4	6	8	10		누계 : 1474	

〈표 5〉 개선 권고 사항의 기준

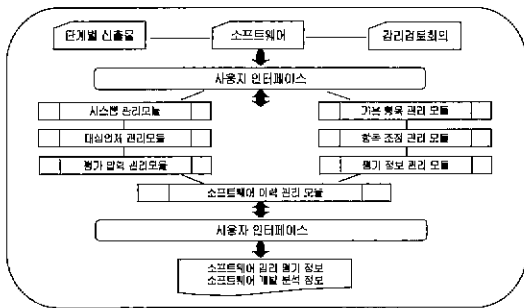
구분	점수
긴급 개선	60점 미만
통상 개선	60~80점 미만
권고 사항	80점 이상

긴급 개선이란 발견된 문제점 중 사전에 정의된 요구사항을 달성하기 위하여 필수적으로 시급히 개선해야 할 사항이며, 통상 개선이란 발견된 문제점 중 사전에 정의된 요구 사항을 달성하기 위하여 필수적으로 개선해야 할 사항이다. 권고 사항은 발견된 문제점 중에서 사전에 정의된 요구 사항을 달성하는데 도움이 된다고 판단되는 사항으로써 감리자와 피감리자의 합의를 거쳐 반영할 수 있는 사항을 의미한다.

4. 시스템 설계 및 구현

4.1 시스템 구조도

감리시스템의 구조는 (그림 4)와 같으며, 주요 내용은 개발된 소프트웨어, 단계별 산출물, 사용자 인터페이스, 각종 관리 모듈로 구성된다.



(그림 4) 감리시스템의 구조

시스템 관리 모듈은 감리시스템의 효율적인 관리를 위하여 사용자 계정 및 비밀번호 등을 관리한다. 기본 항목 관리 모듈은 테스트 단계의 테스트 감리 기준서의 기준 항목을 설정한다. 대상 업체 관리 모듈은 감리 대상 업체, 소프트웨어 개발 업체, 감리 평가 업체 등으로 구분되며, 감리에 대한 일반적인 사항을 관리한다. 평가 정보 관리 모듈 및 소프트웨어 이력 관리 모듈은 감리 평가 보고서 등을 출력하여 시스템 관리자, 개발자, 감리자에게 제공함으로써 소프트웨어 감

리 평가의 효율성과 신뢰성을 제고한다.

4.2 테이블 설계

감리시스템의 테이블 구성은 입력 및 출력 관리로 구분되며, <표 6>과 같다.

〈표 6〉 설계 테이블

테이블 명칭	항목 명칭	형 태	길 이
감리 지침	Index	숫자	8
	감리지침	문자	100
기준_주요항목	Index	숫자	8
	감리기준	숫자	8
	항목명	문자	255
	기준점수	숫자	8
기준_개부항목	Index	숫자	8
	항목번호	숫자	8
	항목명	문자	255
	기준점수	숫자	8
결과_주요항목	Index	숫자	8
	감리기준	숫자	8
	감리번호	문자	4
	항목명	문자	255
	기준점수	숫자	8
	결과점수	숫자	8
	권고사항	문자	1
	사용여부	문자	1
	결과_세부항목	Index	숫자
항목번호		숫자	8
감리번호		문자	4
항목명		문자	255
기준점수		숫자	8
결과점수		숫자	8
검토의견		문자	6
사용여부		문자	1
비고		문자	100
감리 일반 현황		감리번호	문자
	감리명	문자	50
	대상_업체명	문자	50
	대상_대표자명	문자	8
	대상_사업시번호	문자	14
	대상_주소	문자	60
	용역_업체명	문자	50
	용역_대표자명	문자	8
	용역_사업시번호	문자	14
	용역_주소	문자	60
	감리_업체명	문자	50
	감리_대표자명	문자	8
	감리_사업시번호	문자	14
	감리_주소	문자	60
	최수회의시간	문자	12
	조사시간_부터	일자	8
	조사시간_까지	일자	8
	보고서제출일	일자	8
	감리단계	문자	50

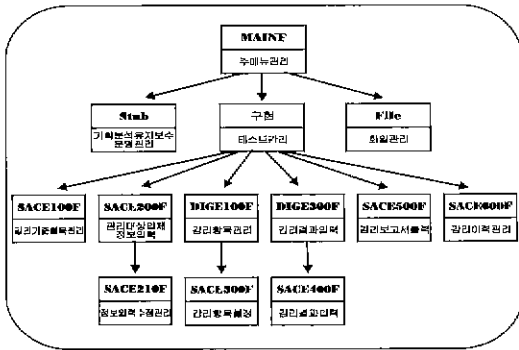
4.3 화면 설계

감리시스템의 주요 화면은 주메뉴, 감리 기준 항목 관리, 감리 대상 업체 정보 입력, 감리 항목 설정, 감리 결과 입력 화면 등으로 구성되어 있다.

감리시스템의 분석 정보에 대한 설계 내용에는 시스템 감리 기준서, 테스트 감리 평가서, 주요 항목별 그래프, 세부 항목별 그래프, 주요 항목별 권고 사항, 세부 항목별 평가 사항 등이 있다. 또한 감리 평가의 이력 관리 측면에서 지원되는 내용에는 업무별 주요 항목 분석 현황 및 권고 사항, 업무별 세부 사항 분석 현황 및 평가 사항 등이 있다.

4.4 설계 구조도

감리시스템의 주메뉴에서 기획 감리, 분석 감리, 설계 감리, 유지보수 감리, 운영 감리 부분은 스텐브로 처리하였고, 테스트 감리에 대한 서브 메뉴는 감리 기준 항목 관리, 감리 대상 업체 정보 입력, 감리 항목 설정, 감리 결과 입력, 감리 보고서 출력, 감리 이력 관리 등의 모듈로 구성되어 있다. 그 개략적인 설계 구조도는 (그림 5)와 같다.



(그림 5) 감리시스템의 구조도

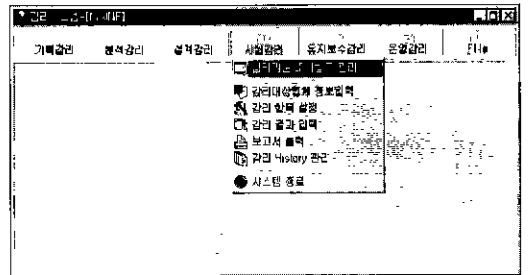
4.5 시스템 구현

설계된 감리시스템의 개발 환경에서 사용된 하드웨어는 IBM PC 호환 기종인 펜티엄 PC를 이용하였고, 운영 체제는 한글 Windows 98이며, 개발에 사용된 언어는 델파이(Delphi) 4.0을 이용하였다.

4.5.1 초기화면 및 주메뉴

감리시스템의 초기화면은 시스템을 사용할 수 있는 사용자의 비밀번호가 정확하게 일치되어야 사용이 가능하며, 주메뉴는 (그림 6)과 같이 기획 감리, 분석 감

리, 설계 감리, 유지보수 감리, 운영 감리 부분은 스텐브로 처리하였고, 테스트 감리에 대한 서브 메뉴는 감리 기준 항목 관리, 감리 대상 업체 정보 입력, 감리 항목 설정, 감리 결과 입력, 감리 보고서 출력, 감리 이력 관리 등으로 구성되어 있다.



(그림 6) 감리시스템의 주메뉴

델피에서 폼을 추가하면 만들어지는 프로그램 소스가 유닛이다 유닛 소스는 .pas 확장자로 구성된다. 감리시스템의 중요한 유닛 소스는 <표 7>과 같다.

<표 7> 감리시스템의 유닛 소스

유닛소스 명칭	주요 프로그램 기능
SACF100DU.pas	감리기준 조사항목 관리 Main 프로시저
SACE100U.pas	주요 항목 설정에서 등록 실행 Sub 프로시저
SACE110U.pas	주요 항목 설정에서 수정 실행 Sub 프로시저
SACE120U.pas	주요 항목 설정에서 등록/수정 확인 Sub 프로시저
SACT200DU.pas	감리대상업체 정보 입력 Main 프로시저
SACE200U.pas	감리대상업체 관리 세부내용 처리 Sub 프로시저
SACE210U.pas	감리대상업체 관리 정보 입력/수정 Sub 프로시저
SACE300DU.pas	감리항목 설정 Main 프로시저
SACE300U.pas	감리항목 설정 세부내용 처리 Sub 프로시저 (등록, 삭제, 가지오기, 재개신 모듈 실행)
SACE400DU.pas	감리 결과 입력 Main 프로시저
SACE400U.pas	감리 결과 입력 세부내용 처리 Sub 프로시저 (Check 해제, 개산 모듈 실행)
SACE500DU.pas	감리 보고서 출력 Main 프로시저
SACE500U.pas	감리 보고서 출력 세부내용 처리 Sub 프로시저 (인체, 전체, 해체, 반건 모듈 실행)
SACE600DU.pas	감리 이력 관리 Main 프로시저
SACE600U.pas	감리 이력 관리 세부내용 처리 Sub 프로시저 (인체, 전체, 해체, 반건 모듈 실행)
SACP100U.pas	테스트 감리 기준서 출력 프로시저
SACP200U.pas	테스트 감리 평가서 출력 프로시저
SACP500U.pas	주요 항목별 그래프 출력 프로시저
SACP600U.pas	세부 항목별 그래프 출력 프로시저
SACP700U.pas	주요 항목별 권고사항 출력 프로시저
SACP800U.pas	세부 항목별 권고사항 출력 프로시저
SACP910U.pas	주요 항목별 분석현황 출력 프로시저
SACP920U.pas	세부 항목별 분석현황 출력 프로시저
SACP930U.pas	주요 항목 권고사항별 분석 그래프 출력 프로시저
SACP940U.pas	세부 항목 평가사항별 분석 그래프 출력 프로시저

4.5.2 감리 기준 항목 관리

감리 기준 항목 관리 프로그램은 주요 항목과 세부 항목을 관리하는 것으로 10개의 프로시저로 구성되어 있다.

4.5.3 감리 대상 업체 정보 입력 관리

감리 평가에 관한 일반적인 사항(사용 업체, 개발 업체, 감리 업체)을 효율적으로 관리하기 위한 것으로 10개의 프로시저로 구성되어 있다.

4.5.4 감리 항목 설정 관리

감리 기준 항목 관리에서 설정한 기본 항목을 감리 업무의 성격에 따라 감리 평가 항목을 재조정하는 것으로 15개의 프로시저로 구성되어 있다.

4.5.5 감리 결과 입력 관리

소프트웨어에 대한 감리를 실행한 결과를 입력하는 것으로 14개의 프로시저로 구성되어 있다.

4.5.6 감리 보고서 출력 관리

소프트웨어에 대한 감리 평가의 결과를 화면이나 프린터로 출력하기 위한 것으로 12개의 프로시저로 구성되어 있다.

4.5.7 감리 이력 관리

개발된 소프트웨어에 대한 테스트 감리의 평가 결과를 누적하여 향후 소프트웨어 감리 또는 소프트웨어 개발에 필요한 정보를 분석하기 위한 것으로 11개의 프로시저로 구성되어 있다.

5. 시스템 적용 및 분석

대상 프로젝트는 '98년 9월부터 '99년 7월까지 약 11개월간 237Man/Month의 요원에 의해 개발된 “초·중등학교 과학 가상 실험·실습 시스템 구축 시스템”을 선정하였다. 이 시스템은 한국교육학술정보원에서 주관하고 (주)미레넷에서 시스템을 개발하였다.

본 절에서는 소프트웨어 개발의 테스트 단계에 적용한 결과를 토대로 평가 및 분석정보를 중심으로 감리 평가에서 어떻게 적용되고, 어떠한 역할을 수행하는지 소개한다

5.1 소프트웨어 평가 정보

5.1.1 테스트 감리 기준서

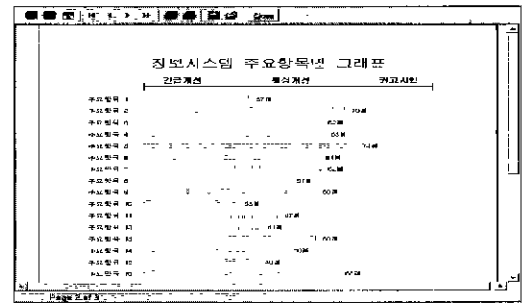
테스트 감리 기준서는 (그림 7)과 같이 소프트웨어에 대한 테스트 감리 평가에서 주요 항목(35개)과 세부 항목(114개)의 내용과 기준 점수를 파악할 수 있는 표준 지침서이다.

구분	내 용	점 수
1	개발 단계별 주요 항목별 평가 기준서	100
2	개발 단계별 세부 항목별 평가 기준서	100
3	개발 단계별 세부 항목별 평가 기준서	100
4	개발 단계별 세부 항목별 평가 기준서	100
5	개발 단계별 세부 항목별 평가 기준서	100
6	개발 단계별 세부 항목별 평가 기준서	100
7	개발 단계별 세부 항목별 평가 기준서	100
8	개발 단계별 세부 항목별 평가 기준서	100
9	개발 단계별 세부 항목별 평가 기준서	100
10	개발 단계별 세부 항목별 평가 기준서	100
11	개발 단계별 세부 항목별 평가 기준서	100
12	개발 단계별 세부 항목별 평가 기준서	100
13	개발 단계별 세부 항목별 평가 기준서	100
14	개발 단계별 세부 항목별 평가 기준서	100
15	개발 단계별 세부 항목별 평가 기준서	100
16	개발 단계별 세부 항목별 평가 기준서	100
17	개발 단계별 세부 항목별 평가 기준서	100
18	개발 단계별 세부 항목별 평가 기준서	100
19	개발 단계별 세부 항목별 평가 기준서	100
20	개발 단계별 세부 항목별 평가 기준서	100
21	개발 단계별 세부 항목별 평가 기준서	100
22	개발 단계별 세부 항목별 평가 기준서	100
23	개발 단계별 세부 항목별 평가 기준서	100
24	개발 단계별 세부 항목별 평가 기준서	100
25	개발 단계별 세부 항목별 평가 기준서	100
26	개발 단계별 세부 항목별 평가 기준서	100
27	개발 단계별 세부 항목별 평가 기준서	100
28	개발 단계별 세부 항목별 평가 기준서	100
29	개발 단계별 세부 항목별 평가 기준서	100
30	개발 단계별 세부 항목별 평가 기준서	100
31	개발 단계별 세부 항목별 평가 기준서	100
32	개발 단계별 세부 항목별 평가 기준서	100
33	개발 단계별 세부 항목별 평가 기준서	100
34	개발 단계별 세부 항목별 평가 기준서	100
35	개발 단계별 세부 항목별 평가 기준서	100

(그림 7) 테스트 감리 기준서의 내용

5.1.2 주요 항목별 그래프

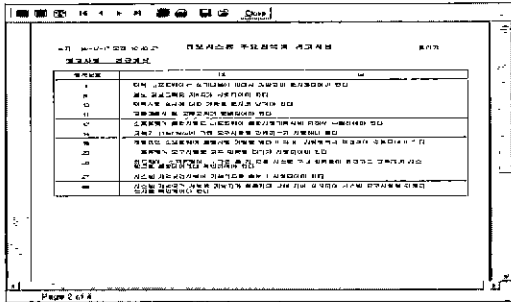
개발된 소프트웨어에 대한 감리 평가를 수행하고 그 결과를 입력하면, (그림 8)과 같이 소프트웨어에 대한 감리 평가 결과는 긴급 개선, 통상 개선, 권고 사항 등으로 구분되어 처리된다.



(그림 8) 주요 항목별 권고 사항

5.1.3 주요 항목별 권고 사항

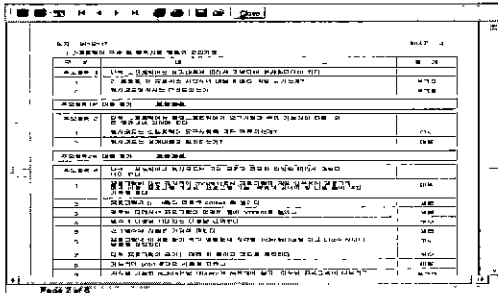
소프트웨어의 감리 평가 결과는 세부 항목별 점수를 계산하여 개선 권고 사항을 결정한다. 개선 권고 사항의 유형이 60점 미만이면 긴급 개선, 60점 이상 80점 미만이면 통상 개선, 80점 이상이면 권고 사항 등으로 분류된다. 개선 권고 사항별로 주요 항목이 처리된 결과는 (그림 9)와 같이 제공된다.



(그림 9) 주요 항목별 권고 사항

5.1.4 테스트 감리 평가서

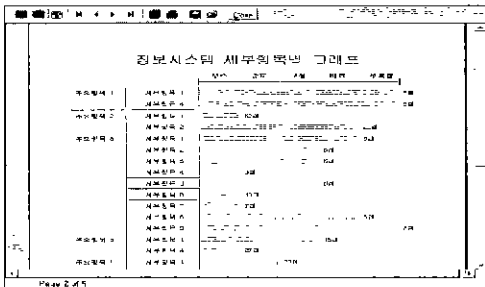
감리시스템의 감리 항목은 주요 항목이 35개, 세부 항목이 114개로 구성되어 있다. 세부 항목별 점수가 결정되면 그에 해당하는 주요 항목의 권고 사항이 (그림 10)과 같이 결정된다.



(그림 10) 테스트 감리 평가서

5.1.5 세부 항목별 그래프

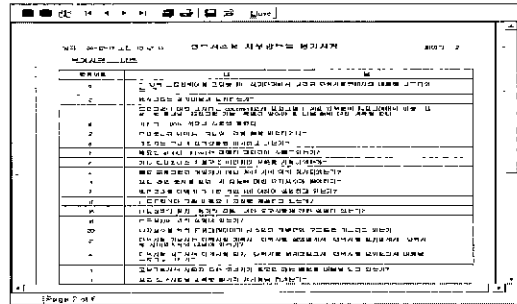
세부 항목별 그래프 정보는 (그림 11)과 같이 세부 항목 평가 결과 그래프로 나타내었으며, 세부 항목 평가 결과에 대해 정량적인 평가가 가능하도록 세부 항목별 점수를 계산하여 나타내었다.



(그림 11) 세부 항목별 그래프

5.1.6 세부 항목별 평가 사항

세부 항목별 평가 사항은 (그림 12)에 나타난 것처럼 감리자가 개발된 소프트웨어를 감리 평가하여 우수, 양호, 보통, 미흡, 부적정 등으로 분류한 것이다. 즉, 세부 항목 114개에 대한 평가 결과를 유형별로 처리한 것이다.



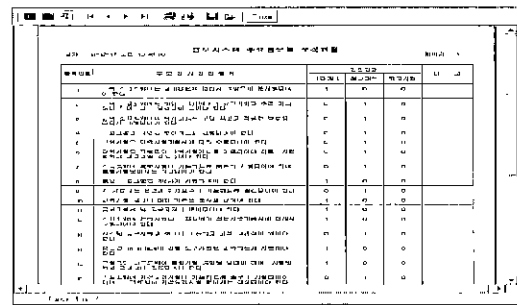
(그림 12) 세부 항목별 평가 사항

세부 항목별 평가 사항을 그룹화 하여 분류함으로써 어떤 분야에 어떻게 소프트웨어를 수정, 보완해야 할 것인지를 판단하는 중요한 정보를 감리자에게 제공할 수 있다.

5.2 소프트웨어 분석 정보

5.2.1 주요 항목 분석

감리 이력 관리에서 이 프로시저는 소프트웨어에 대한 감리 평가 결과를 누적한 값이며, 주요 항목별 분석 현황으로써 그 결과는 (그림 13)과 같다.



(그림 13) 주요 항목 분석 결과

소프트웨어에 대한 감리의 평가 결과를 계속 누적하여 관리함으로써, 주요 항목별 개선 권고 사항별 이력 정보를 지원하여, 향후 소프트웨어의 감리 평가 및 개

발에 필요한 전략을 수립할 수 있다.

5.2.2 세부 항목 분석

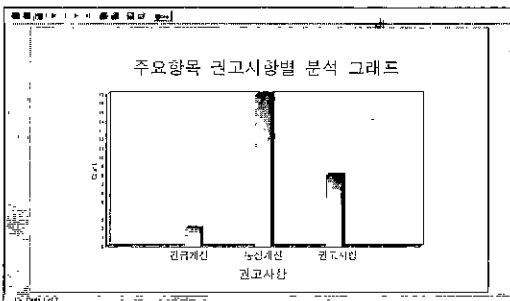
소프트웨어에 대한 감리 평가를 실행한 결과의 누적 값이며, 세부 항목별 분석 현황으로써 그 결과는 (그림 14)와 같이 향후 소프트웨어에 대한 감리 평가 및 개발에 필요한 방향 및 지침 준비에 유용하게 활용될 수 있다.

항목번호	항목명	평가	합계	비고
1	요구사항	1	1	
2	분석	1	1	
3	설계	1	1	
4	개발	1	1	
5	검토	1	1	
6	시행	1	1	
7	유지보수	1	1	
8	총계	8	8	

(그림 14) 세부 항목 분석 결과

5.2.3 주요 항목 권고 사항별 분석 그래프

감리시스템을 이용하여 개발된 소프트웨어의 감리 평가를 계속하면 소프트웨어 감리 평가의 권고 사항별, 즉 긴급 개선, 통상 개선, 권고 사항 등으로 평가 자료를 축적하여 (그림 15)와 같이 향후 감리 평가 또는 소프트웨어를 개발할 때 관리자, 개발자, 감리자에게 분석 정보를 제공할 수 있다.

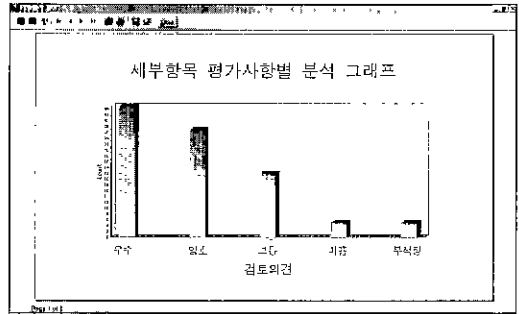


(그림 15) 주요 항목 권고 사항별 분석 그래프

5.2.4 세부 항목 평가 사항별 분석 그래프

감리시스템에서 세부 항목은 부적정, 미흡, 보통, 양호, 우수 등으로 대상의 소프트웨어를 5단계로 평가한다. 평가된 이력 정보를 유지 관리함으로써 각 세부

항목별 평가 사항을 분석하여 (그림 16)과 같이 제공할 수 있다.



(그림 16) 세부 항목 평가 사항별 분석 그래프

6. 결 론

본 연구의 감리시스템 모델 제안 및 구현에서는 소프트웨어 개발 과정 중에 특별히 테스트에 관련이 있는 각 단계에서의 정량적 감리 지침과 평가 시스템을 제공하고 있다. 본 논문에서는 소프트웨어 감리 평가 항목의 객관성·타당성 입증과, 정량적 평가 기준을 마련하였다. 소프트웨어 관리자, 개발자, 감리자가 곧바로 현장에서 적용할 수 있는 내용으로 감리시스템을 설계하고 구현하였다. 본 연구의 결과를 요약하면 다음과 같다.

첫째, 감리 평가의 질문 형식을 보다 현실화, 구체화 하였다.

둘째, 소프트웨어 테스트 단계에서 중요하게 평가해야 할 기준의 대안을 제시하였다.

셋째, 주요 항목(35개 항목) 및 세부 항목(114개 항목) 각각에 대하여 정량적 평가가 가능하도록 기준을 개발하였다.

넷째, 각 항목별 평가 기준을 제시함으로써 각 단계에서 필요한 감리 평가 항목의 추가 및 삭제가 가능하도록 하였다.

다섯째, 평가 기준 및 정량적 평가의 수학적인 표현 절차를 일반화하였다.

여섯째, 소프트웨어의 개발 및 감리 평가에서 관리자, 개발자, 감리자가 사용이 용이하도록 감리시스템을 설계하고 구현하였다.

일곱째, 감리 업무에 활용할 수 있는 권고 사항 및 평가에 관련된 대안을 제시하였다.

향후 연구에서는 감리시스템을 다양한 개별 업무에 적용해 보고 문제점 및 평가 항목의 보완이 요구되며, 각 항목의 평가 기준을 표준화하는 것이다. 또한 소프트웨어의 테스트 단계에서 한정적으로 적용한 감리시스템을 SDLC 전 단계에서 활용되도록 각 단계별 평가 항목 개발 및 시스템을 개발하는 것이다.

참 고 문 헌

[1] Florac, W. A., Park, R. E., and Carleton, A.D., "Practical Software Measurement: Measuring for Process Management and Improvement," CMU/SEI-97-HB-003, April 1997.

[2] Jung, H. W., "Software Quality Metrics Applicable to the ISO 9000-3." Proceedings of the 4th International Conference on Software Quality, Maribo Slovenia, Nov. 1997.

[3] ISACA, "IS Audit & Control Journal," 1997.

[4] Tasso, C., R. Adey, and M. Pighin, "Software Quality Engineering," Computational Mechanics Publications, 1997.

[5] ISACA, "COBIT Audit Guidelines," ISACA, Sept. 1996.

[6] ISO/IEC. "ISO/IEC 12207 : Software Life Cycle Process," 1995

[7] Thomas M Pigoski, "Software Maintenance : A Practical Approach," Technical Software Services, Inc., 1995.

[8] Laudon, K. C. and J. P. Laudon, "Management Information Systems," Macmillan Publishing Company, 1994.

[9] Booch, Grady. "Object-Oriented Analysis and Design with Applications," Addison-Wesley, 1994

[10] Tom Gilb, Dorothy Graham, "Software Inspection," Addison-Wesley, 1993.

[11] Shepperd, Martin. "Software Engineering Metrics I : Measures and Validations," McGraw-Hill, 1993.

[12] <http://www.trillion.demon.co.uk>

[13] <http://www.isaca.org>

[14] <http://www.nic-nz.com>

[15] <http://sassafras.com>

[16] 김현수, "정보시스템 진단과 감리", 법영사, 1999.

[17] 문대원, 장시영, "정보시스템감리", 명경사, 1998.

[18] 한국전산원, "정보시스템 감리교육 교재", 1998.

[19] 한국전산원, "전산감리", 1996.

[20] 서용무, 신동익 외, "시스템 시험 감리지침연구", 한국전산원 연구보고서, 1997.

[21] 김혁, 현대 정보시스템 감사론", 무역적영사. 1996.

[22] 김공현, "정보시스템 감사", 법문사, 1995.



권 대 곤

e-mail : dgkweon@nc.namhae.ac.kr
 1984년 경남대학교 전자계산학과 졸업(공학사)
 1996년 경남대학교 대학원 컴퓨터 공학과 졸업(공학석사)
 1999년 경남대학교 대학원 컴퓨터 공학과 박사 수료

1984년~1989년 육군군수사령부 프로그램장교 근무
 1997년~현재 남해전문대학 사무자동화와 조교수
 관심분야 : 소프트웨어 공학(테스팅, 신뢰성 평가, 품질 관리, 시스템 감리&컨설팅)



한 판 암

e-mail : pahan@zeus.kyungnam.ac.kr
 1969년 동국대학교 졸업
 1975년 동국대학교 경영대학원 졸업(경영학 석사)
 1989년 명지대학교 대학원 졸업(공학 석사)

1992년 인천대학교 대학원 졸업(경영학 박사)
 1980년~현재 경남대학교 공과대학 컴퓨터공학과 교수
 1999년~현재 한국정보처리학회 부회장
 관심분야 : 소프트웨어 품질관리 및 신뢰성, 소프트웨어 개발 환경, 정보 공학