철도시스템 바이탈 소프트웨어 테스팅을 위한 Fagan Inspection 지원도구의 개발

Development of Fagan Inspection Tool for Railway System Vital Software

황종규* 조현정 정의진 신경호
Hwang, Jong-Gyu Jo, Hyun-Jeong Jeong, Ui-Jing Shin, Kyeung-ho

ABSTRACT

Recent advances in computer technology have brought more dependence on software to train control systems. Hence, the safety assurance of the vital software running on the railway system is very critical task and yet, not many works have been done. While much efforts have been reported to improve electronic hardware's safety, not so much systematic approaches to evaluate software's safety, especially for the vital software running on board train controllers. In this paper, we have developed the static software testing tool for railway signaling, especially Fagan Inspection supporting tool. This static testing tool for railway signaling can be utilized at the assessment phase, and also usefully at the software development stage also. It is anticipated that it will be greatly helpful for the evaluation on the software for railway signalling system.

1. 서 론

열차제어시스템은 고속으로 동작하는 철도차량의 속도 및 진로를 실시간적으로 제어해야 하는 철도시스템에서의 핵심적인 장치이다. 이러한 열차제어시스템은 최근 기존의 기계 및 전기적인 장치로부터 컴퓨터시스템으로 전환되고 있으며, 특히 안정화된 하드웨어 보다는 소프트웨어에 그 의존성이 급격하게 증가하고 있어, 소프트웨어에의 의존성이 급격하게 증가하고 있다. 최근의 컴퓨터 기술의 발달에 따라 지능화 및 자동화를 위해 열차제어시스템의 소프트웨어가 더욱 복잡해지게 되면서, 시스템에서 소프트웨어가 차지하는 비중이 더욱 증대되고 있다. 이러한 철도시스템 소프트웨어의 주요한 특징은 실시간성과 고신뢰도, 그리고 안전성을 들 수 있다. 즉, 철도시스템 소프트웨어가 고신뢰도와 안전성을 만족하지 못하는 경우 열차의 충추돌 등의 사고를 발생시킬 수 있는 바이탈한 시스템이다. 따라서 이러한 철도시스템에 임베디드화 되는 제어 소프트웨어의 안전성을 검증하는 것이 중요한 문제로 대두되기 시작했다.

이에 따라 철도시스템 소프트웨어 안전성관련 기준들은 IEC 61508-3과 IEC 62279에 의해 국제표준화 되었고, 또한 국내에서도 철도안전법이 제정되는 등 국제표준에서 요구하는 각종 테스팅 및 검증활동을 요구하기 시작했다[1]-[3]. 철도시스템 중 특히 바이탈 제어장치인 열차제어시스템의 소프트웨어 안전성 평가를 위해서는 관련된 국제규격에서 소프트웨어의 분석 및 측정을 통한 정량적인 평가뿐만 아니라 소프트웨어 개발과정의 안전성 활동에 대한 문서 등의 검증을 통한 정성적인 분석도 안전성 평가의 중요한 요소로 요구하고 있다. 특히 대부분 안전무결성 등급(SIL : Safety Integrity Level) 등급이 3 또는 4로 분류되는 ATP나 전자연동장치와 같은 바이탈한 열차제어시스템 소프트웨어의 경우 Fagan Inspection을 정적분석의 방법으로 'HR : Highly Recommend' 조건으로 규정하고 있다. 이러한 Fagan Inspection은 평가 자체가 아니라 평가 과정에 대한 품질확보를 위한 측면이 강한 방법이지만 국제표준에서 정적 테스팅을 위한 방법의 하나로 제시하고 있는 열차제어시스템 소프트웨어 안전성 평가에 있어서 매우 중요한 필수적인 부분이다.

† 책임저자 : 정회원, 한국철도기술연구원, 열차제어통신연구실
E-mail : jghwang@krri.re.kr
TEL : (031)460-5438

하지만 국내에서는 철도시스템 소프트웨어의 Fagan Inspection을 지원하는 한글화된 도구가 개발된 적이 없고 몇몇 외국 제품들에 의해 상용화되고 있다. 본 논문에서는 이러한 철도소프트웨어 안전성 평가 체계 구축의 일환으로 국제규격에서 철도소프트웨어의 안전성 평가를 위해 필수적으로 요구하고 있는 Fagan Inspection을 지원하기 위한 도구를 개발하였다.

2. 열차제어시스템 소프트웨어 안전성 평가

열차제어시스템 임베디드 소프트웨어는 개발초기부터 테스팅 과정을 통해 버그를 확인하여 품질비용을 낮출 수 있으며, 또한 개발 완료 후 검증과정에서도 안전성 평가 과정이 필수적으로 요구되고 있다. 열차제어시스템 소프트웨어는 하드웨어에 의존적이고 높은 안전성이 요구되는 바이탈 소프트웨어여서 다른 어느 제어시스템 소프트웨어보다 엄격한 안전성 평가가 요구되고 있으나 아직 이의 지원을 위한 도구들에 대한 연구가 많이 진행되고 있지 않고 있으며, 단지 산업용 임베디드 시스템의 소프트웨어를 대상으로 일부 항목에 대한 도구들이 사용되고 있다.

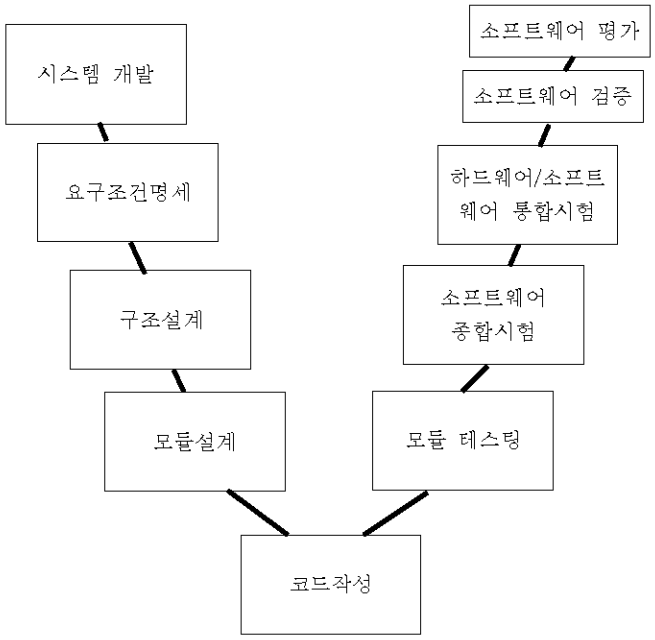

그림 1. 소프트웨어 개발 수명주기

본 논문에서의 열차제어시스템 소프트웨어 안전성 평가 기술 개발 및 체계 구축을 위해 필수적으로 요구되는 정적 테스팅을 위한 Fagan Inspection 지원 도구를 개발하였다. 그림 1은 IEC 62279에서 제시한 소프트웨어 수명주기를 나타낸 것으로 일반적인 제어시스템 수명주기와 유사하다. 소프트웨어의 안전성 평가를 이 수명주기의 마지막 단계인 '소프트웨어 평가' 단계에서 동적 테스팅 위주로 수행하지만, 철도의 열차제어시스템과 같은 바이탈 소프트웨어는 동적 테스팅에 더불어 수명주기 전 단계에서 걸쳐서 수행하는 프로세스 리뷰 등과 같은 정적인 평가방법을 매우 중요한 평가요소로 요구하고 있다. 이러한 바이탈한 소프트웨어의 안전성 평가를 위한 정적 분석 방법 중 본 논문에서는 정형화된 인스펙션 방법의 하나인 Fagan Inspection을 지원하기 위한 도구를 웹기반으로 구현하였다.

Fagan Inspection 방법은 소프트웨어의 안전성 평가를 위한 정적 테스팅 방법의 하나로 철도관련 국제표준에서 'HR'로 요구하고 있는 방법이다. 이는 소프트웨어 개발의 모든 단계에서 정의된 규칙을 기반으로 인스펙션을 수행하고 결과를 등록 및 보고하는 소프트웨어의 품질 및 안전성을 확보하기 위한 방법이다. Fagan Inspection은 소프트웨어의 리뷰 과정을 공식적인 절차에 의해 하도록 하는 페이건에 의해 제안된 방법으로 인스펙션 과정을 통해 코드의 70~80%의 결함이 발견된 것으로 소개하고 있다[4][5]. 이러한 리뷰절차를 체계적으로 수행하기 위한 Fagan Inspection을 소프트웨어 개발 과정에서 적절하게 수행했는지를 확인하는 것이 소프트웨어의 정성적 안전성 평가에서 매우 중요한 요소이며, 이러한 인스펙션 프로

세스가 자동화된 도구에 의해 지원된다면 보다 효율적으로 수행될 수 있을 것이다.

3. Fagan Inspection 지원 모듈의 설계

3.1 Fagan Inspection 일반

Fagan Inspection 방법은 철도 소프트웨어관련 국제 표준에서 소프트웨어의 안전성 평가를 위한 정적 테스팅의 하나로 요구하고 있는 방법이다. 이는 소프트웨어 개발의 모든 단계에서 정의된 규칙을 기반으로 인스펙션을 수행하고 결과를 등록 및 보고하는 소프트웨어의 품질 및 안전성을 확보하기 위한 방법으로, 그림 2와 같은 인스펙션 수행절차를 통해 소프트웨어 개발과정에서의 요구사양이나 설계의 적정성 및 잠재되어 있는 오류를 확인할 수 있다. 안전성 평가를 위해서는 소프트웨어의 각 개발단계에서 이러한 인스펙션이 정의된 규칙 및 절차에 따라 수행되었는지 와 인스펙션 결과의 처리여부 등의 평가를 수행하게 된다.

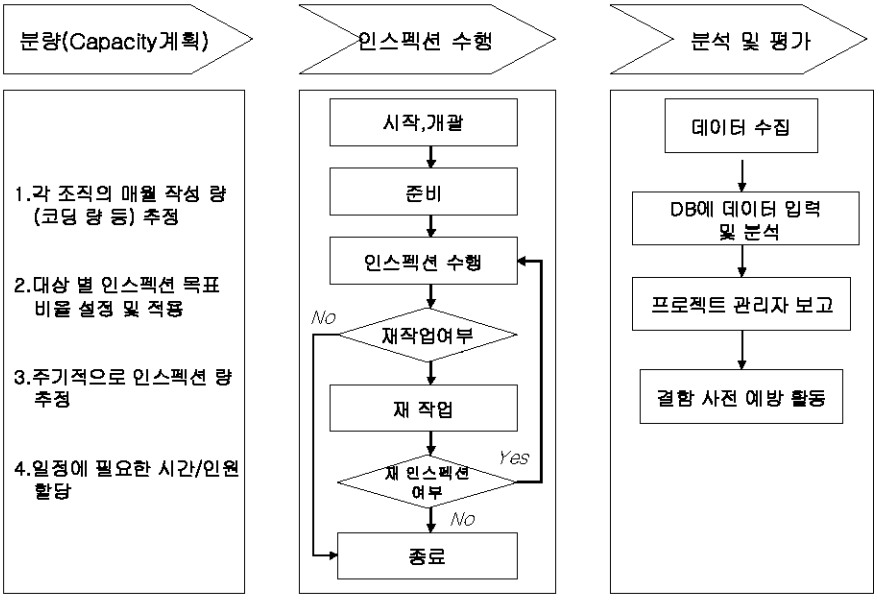

그림 2. 인스펙션 절차

표 1. Fagan Inspection의 단계

단계	설명
Planning	·Inspection할 산출물이 entry criteria를 만족함을 확인 ·Inspector 역할 할당 (Moderator, Author, Reader, Tester)
Overview	·Inspection 팀이 preparation 단계를 잘 실행할 수 있도록 배경 설명, context, rationale에 대한 교육 실시
Preparation	·각자 Inspection할 산출물의 습득 ·각자 맡겨진 역할을 수행할 수 있도록 준비 ·Defect로 단정 짓지 말고 Inspection 회의 시 질문 사항 기록
Inspection	·Find Defect (defect 해결책 또는 개선책을 거론하지 말 것)
Process Improvement	·향후 defect 발견을 향상 시킬 수 있도록 선행 단계 검토 ·Systematic defect 식별과 Recommended fixes
Rework	·모든 발견된 defects 수정 및 조사항목 보완
Verification	·모든 보완 사항과 해결책이 적합한가를 검증

이러한 정적분석 방법의 하나인 인스펙션은 표1과 같은 7가지의 단계를 가진다. 소프트웨어 개발의 각 단계별로 각각 표 1과 같은 인스펙션을 수행하고 또한 그 과정을 기록하여야 안전성 평가 시에 평가될 수 있도록 하여야 한다. 이러한 Fagan Inspection은 지금까지는 대부분 오프라인 방식에 의해 수행되어 왔다. 하지만 열차제어시스템과 같은 높은 안전성이 요구되는 바이탈 소프트웨어의 경우 이러한 인스펙션을 필수적으로 요구하고 있고 또한 그 결과가 안전성 평가에서 검증되도록 규정되어 있어, 지금까지와 같은 오프라인 방법에 의해 수행할 경우 인스펙션 과정이 비효율적일 뿐 아니라 안전성 평가를 위한 문서화 작업이 불가능하여 이를 효과적으로 지원할 수 있는 도구의 개발이 필요하다.

3.2 Fagan Inspection 모듈의 설계

국제표준에 의한 열차제어시스템 소프트웨어 테스팅 자동화 도구의 한 부분으로써 프로젝트 단위로 작업을 구분하여 소프트웨어 개발 절차에 따라 각 단계 별 Fagan Inspection 모듈을 이용하여 개발하는 소프트웨어의 안정성을 평가하는 웹기반 테스팅 도구를 설계하였다. 본 논문에서는 C나 C++에 적합한 Fagan Inspection을 위한 규칙을 정의하였고, 정의된 규칙을 기반으로 인스펙션을 수행하고, 인스펙션 수행 결과 등록 및 관리·보고서 작성 기능, 단계별 산출물 등록, 시험대상 코드 등록, 검사 수행 및 결과 등록, 검사 결과 보고 및 조회 등의 기능을 구현하여 안전성 평가를 지원하기 위한 Fagan Inspection 도구를 설계하였다.

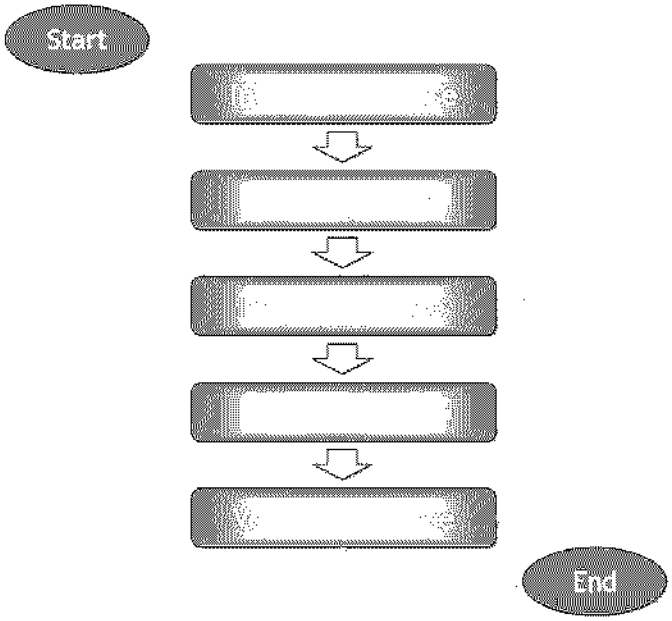

그림 3. 설계한 Fagan Inspection 단계

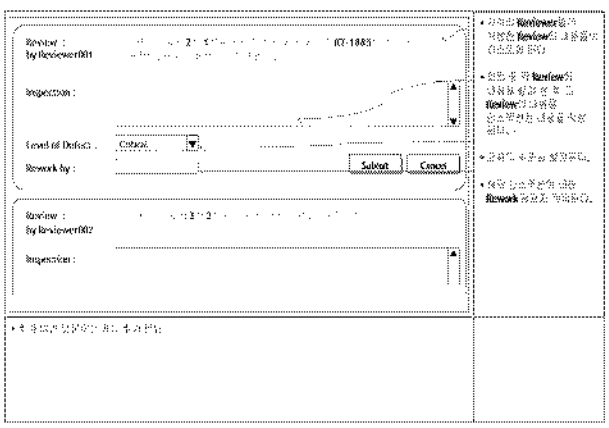

그림 4. 인스펙션 단계의 화면 설계

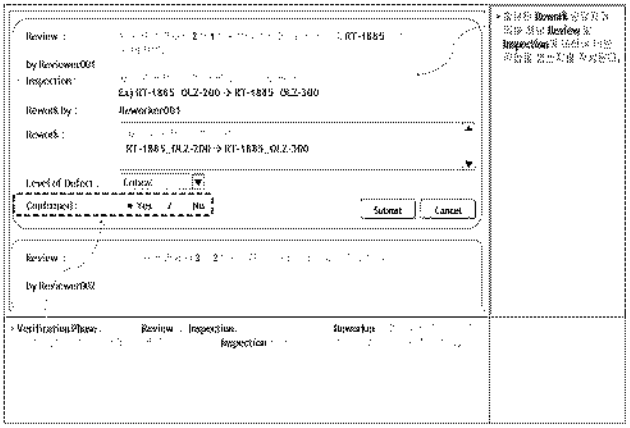

그림 5. Rework & Verification 단계의 화면 설계

본 논문에서는 그림 2와 같은 인스펙션 절차를 일반화하여 그림 3과 같은 기본 단계를 바탕으로 열차제어시스템 소프트웨어 개발 수명주기 전반에 걸쳐 적용 가능하도록 하였다. 또한 열차제어시스템의 각 소프트웨어, 혹은 각 모듈 별로 구성되는 프로젝트 단위로 Fagan Inspection을 적용 가능하도록 하는 기능이 필요하다. 이에 따라 각 프로젝트 별로 관리 및 참여자(User)의 관리 역시 함께 필요하다. 즉, 정의된 규칙을 기반으로 Fagan Inspection 수행하고 그 결과를 등록 및 보고할 수 있도록 하여 단계 별 산출물을 등록하여 Fagan Inspection에서 이용이 가능하도록 지원하고 시험대상 코드 역시 웹 시스템에 등록하여 Fagan Inspection에 참여하는 참여자들이 함께 정보를 공유하고, 열람하며, 그에 따른 검사를 수행할 수 있도록 설계하였다. 각 단계별 인스펙션의 결과를 프로젝트 관리자가 획득하여 활용할 수 있도록 웹 시스템에 등록할 수 있도록 하고, 또한 인스펙션의 결과들을 취합하여 자동으로 보고서가 작성될 수 있도록 설계하였다. 또한 Fagan Inspection 모듈은 크게 관리자와 일반 유저 관점으로 구분하여 제한된 기능이 활용될 수 있도록 하였다. 관리자 관점에서는 이반 유저의 등록이나 수정, 삭제, 검색 등의 일반 유저 관리기능과 프로젝트 등록, 수정, 삭제, 프로젝트 관리자 설정, 검색 등의 프로젝트 관리 기능을 구성하였다. 일반 유저 관점에서는 개인 정보의 관리, 참여 프로젝트 리스트 및 프로젝트 정보 열람 등의 기능으로 구성하였다.

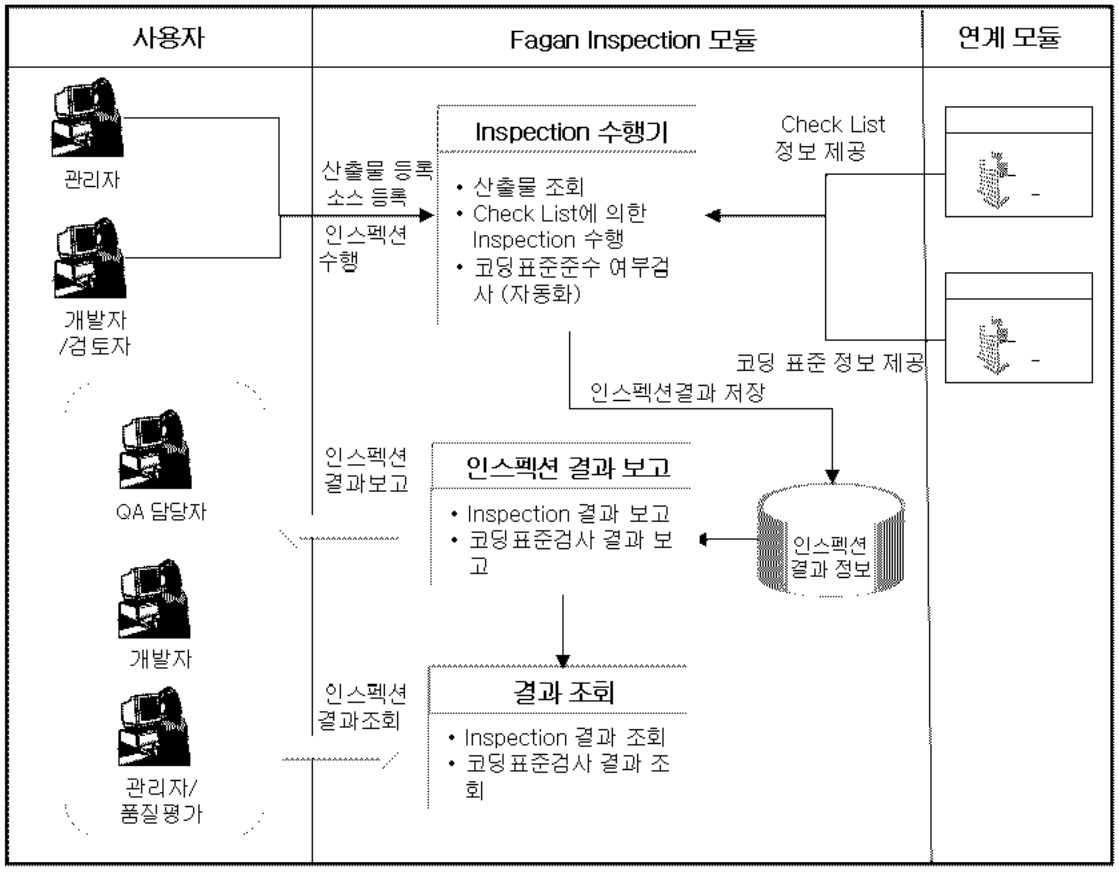

그림 7. Fagan Inspection 모듈의 기능 구성도

그림 4와 5는 앞에서 설명한 바와 같은 Fagan Inspection의 단계들 중 'Inspection Phase'와 'Rework Phase & Verification Phase'을 위해 설계한 윈도우를 나타낸 것이다. 그림 4와 같은 웹기반의 윈도우를 통해 각 리뷰어들이 작성한 내용을 살펴본 후 그 내용을 인스펙션한 내용을 작성하게 되며, 그림 5는 인스펙션한 결과를 해당 리뷰어가 확인하고 그에 응답하고 또한 최종적으로 해당 인스펙션이 반영되었는지를 확인하는 윈도우이다.

4. Fagan Inspection 지원 모듈의 개발

앞 장에서 설명한 국제표준 기반의 열차제어시스템 소프트웨어 안전성 평가를 위한 정적 테스팅을 위한 Fagan Inspection 지원 모듈을 개발하였다. 그림 10은 Fagan Inspection 모듈의 기능 구성도를 나타낸 것으로서, Fagan Inspection 모듈에서는 인스펙션을 위해 체크리스트와 코딩모듈이 입력되어야 한다. 논문을 통해 구현한 정적 테스팅을 위한 지원도구는 MS-SQL Server 2000과 MS-Windows 2003 그리고 MS Internet Explorer 5.0을 기반으로 한 웹 시스템을 구축하였다. 그림 12는 3장에서 설계한 Fagan Inspection 모듈 중 인스펙션 단계의 윈도우 화면을 나타낸 것으로, 이 화면을 통해 Reviewer의 Review의견 별로 인스펙션을 진행하고 그에 대한 의견을 입력할 수 있다. 또한 해당 인스펙션 내용에 대한 Rework 담당자를 결정할 수 있고 Rework 내용을 다시 인스펙션하여 해당 Review 내용의 완료를 결정할 수 있다. 그림 13은 리워크 단계의 윈도우로서 해당 인스펙션의 내용을 볼 수 있으며, 인스펙션의 개요 및 산출물을 검토한 후 등록한 Reviewer의 Review 의견 열람 및 새로운 Review 의견을 입력할 수 있고, 또한 자신이 쓴 Review 의견뿐만 아니라 해당 인스펙션에 참여하는 다른 Reviewer의 Review 의견 역시 함께 검토할 수 있도록 구현하였다. 이러한 일련의 인스펙션 과정들이 구축한 시스템의 데이터베이스에 저장이 되고, 그림 14와 같이 보고서 형태로 출력되도록 하여 모든 인스펙션의 과정과 그 처리 결과를 확인할 수 있도록 하였다.

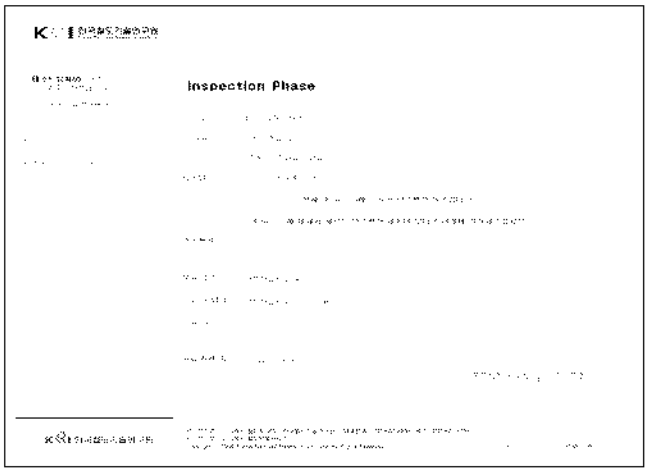

그림 8. 인스펙션 단계 보기

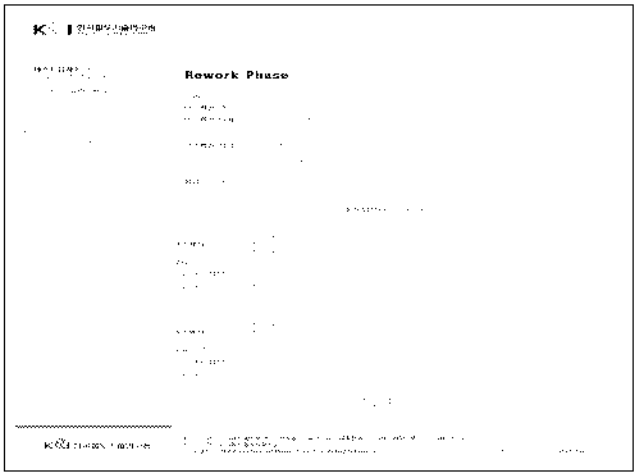

그림 9. 리워크 단계 윈도우

5. 결론

최근 컴퓨터 기술의 발달에 따라 열차제어시스템들이 컴퓨터 소프트웨어에의 의존성이 급격하게 증가하고 있으며, 이러한 기술발전에 따라 열차제어시스템 소프트웨어에 높은 신뢰성과 안전성이 요구되고 있다. 이에 따라 바이탈 제어장치인 열차제어시스템의 소프트웨어 안전성 평가를 위해서는 관련된 국제규격에서 소프트웨어의 분석 및 측정을 통한 정량적인 평가뿐만 아니라 소프트웨어 개발과정의 안전성 활동에 대한 문서 등의 검증을 통한 정적 테스팅도 안전성 평가의 중요한 요소로 요구하고 있다. 본 논문에서는 관련 국제 표준에서 정적 테스팅의 방법으로 요구하고 있는 Fagan Inspection 평가를 지원하기 위한 웹 기반의 도구를 개발하였다. 이러한 정적 테스팅을 위한 웹 기반의 지원 도구는 소프트웨어의 안전성 평가 단계에서 활동될 도구이며, 동시에 소프트웨어 개발과정에서 활용되어질 수도 있도록 화면을 구성하는 등 다양하게 활용될 수 있도록 개발하였다. 즉, 본 도구를 소프트웨어 수명주기 각 단계별 인스펙션에 활용할 수 있고, 이를 통해 보다 높은 안전성과 신뢰성을 갖는 소프트웨어가 확보될 수 있을 것으로 기대된다.

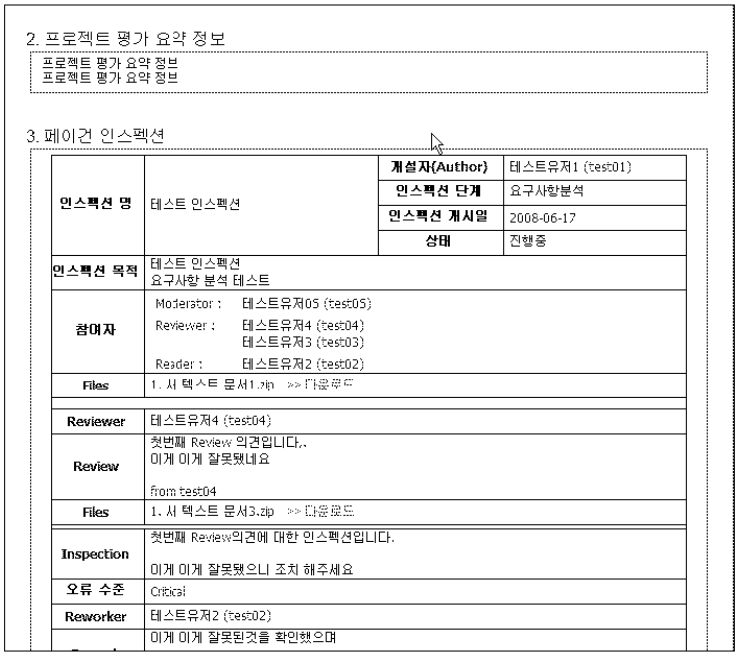

그림 10. 프로젝트 보고서 보기

감사의 글

본 논문은 국토해양부가 출연하고 한국건설교통기술평가원에서 위탁 시행한 철도종합안전기술개발사업의 결과이며, 관계제위께 감사드립니다.

참고문헌

1. IEC 61508-3, "Functional safety of electrical/electronic /programmable electronic safety-related systems - Part 3 Software requirements", 1998.
2. IEC 62279, "Railway Applications - Software for railway control and protection systems", 2002.
3. 철도안전법[법률 8852호], 일부개정 2008.02.
4. N. Eickelmann, F. Ruffolo, etc., "An empirical study of modifying the Fagan Inspection process and the resulting main effects and interaction effects among defects found, effort required, rate of preparation and inspection", Proceedings of the 27th Annual NASA Goddard Software Engineering Workshop, 2002.
5. 소선섭, 차성덕, 권용래, T.J.Shimeall,"페이건 인스펙션의 오류 검출 능력에 관한 실험적 평가", 정보과학회논문지, 제24권 제12호, 1997. 12.