

웹 기반 학습 시스템의 비기능 요구에 대한 구현 검증 기법 연구

서동수[†] · 이혜리^{††}

요 약

웹 기반 학습 시스템의 사용자 요구는 일반적으로 기능 요구와 비기능 요구로 구분된다. 그 중 사용자가 원하는 품질 관련 비기능 요구는 개발자가 반드시 검증해야 할 주요 사항이지만 기능 요구와는 달리 확인 과정에 많은 어려움이 있었다. 본 논문은 웹 기반 학습 시스템의 개발에 적용할 수 있는 검증 기법으로서 품질 관련 비기능 요구사항이 적절히 구현되었는지를 검증하는 방법을 제안한다. 특히 효율성, 사용용이성과 같은 품질 요구가 설계에 영향을 주는 과정을 추적해 얻은 정보를 검증에 이용함으로써 이들 비기능 요구가 적절히 구현되었는지 확인할 수 있게 한다.

키워드 : 비기능 요구, 구현 검증

Verification Methods for the Implementation of Non-functional Requirements in Web-based Learning Systems

Dongsu Seo[†] · Heyli Lee^{††}

ABSTRACT

In general, user requirements in web-based learning systems are divided into functional and non-functional requirements. Developers are responsible for the verification of quality related requirements, which is known to be difficult to perform. This paper suggests a verification method that can be applied in the area where the conformance of implementation for non-functional requirements is required. The paper performs tests for non-functional requirements by using the information extracted from quality-related features that have imposed constraints on design activities.

Keywords: Non-functional requirement, Implementation verification

1. 서 론

Keywords : Curriculum

웹 기반 학습은 원거리 학습자를 교육시키는 유용한 방법으로 전통적인 교실 수업에 비해 사용자 중심의 학습 방법으로 교육 현장에서 인정

되고 있다[1]. 성공적인 웹 기반 학습 시스템을 구축하기 위해서는 다음 세 가지 측면을 고려할 필요가 있다. 먼저, 기획 및 설계 측면에서 웹 기반 학습 시스템은 학습자, 교수자, 그리고 운영자가 참여하는 다양한 사용자 요구를 반영해야 한다. 단순한 HTML 위주의 교육시스템에서 탈피하여 학습자에 대한 다양한 피드백 상황을 만들어주는 것이 필요하다[2].

[†] 정 회 원: 성신여자대학교 컴퓨터정보학부 교수(교신지사)

^{††} 정 회 원: 성신여자대학교 컴퓨터정보학부 연구원

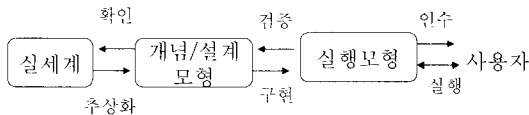
논문접수: 2006년 3월 23일, 심사완료: 2006년 7월 12일

* 본 논문은 2005년도 성신여자대학교 학술연구조성비 지원에 의하여 연구되었음

웹 기반 학습 시스템은 멀티미디어 자료 및 이의 인터랙션에 많이 의존하므로 구현 측면에서 다양한 구현기술을 필요로 한다. 실제 많은 경우 CGI, Perl 등의 인터페이스 언어, ASP, JSP 와 같은 스크립트 언어, 그리고 EJB, .NET과 같은 플랫폼 기술들이 구현에 사용된다.

마지막으로 검증 측면에서 웹 기반 학습 시스템은 당초 목표하였던 학습 목표 및 코스의 목적에 맞도록 구현되었는지 입증되어야 한다. 구현 코드를 대상으로 하는 단순한 테스트만으로는 시스템이 설계목표를 충족시켰는지에 대한 해답을 줄 수 없다. 코드에 관한 테스트 이외에도 적합성 테스트, 비기능 테스트 등 다양한 테스트 및 검증 방법을 동원하여 확인해야 한다.

웹 기반 학습 시스템의 개발은 명세 혹은 모형의 작성에서부터 시작한다. 개발 과정에서 나타나는 명세가 정확한지 판단하는 활동은 그 대상과 목적이 어떤 것인지에 따라 <그림 1>과 같이 구분된다[3].



<그림 1> 명세에 대한 확인 및 검증 구분

여기서의 개념 모형이란 실세계의 문제를 표현하는 모형으로서 문제 해결에 필요한 자료와 기능을 추상적으로 표현한다. 실세계의 문제에 대해 개념 모형이 얼마나 충실한 표현인지를 밝히는 작업을 확인(validation)이라 부른다. 이와는 달리 검증(verification)은 시스템 아키텍처, 설계, 구현 등과 관련된 내부 요구사항들에 대해 적용되는 개념이다. 이 작업은 단계 간의 결과물들이 얼마나 긴밀히 연관되어 있는지를 보여주는데 목적이 있다. 예를 들어 코드가 설계모형을, 설계모형이 개념모형을 얼마나 잘 반영하고 있는지를 보여주는 것은 검증활동의 대표적인 예이다.

비기능 요구는 시스템의 성능, 사용용이성, 보안성, 만족성 등 사용자가 목표로 하는 요구로서 기능 요구와 더불어 시스템의 성공을 가늠하는 중요한 기준이다. 비기능 요구에 대한 확인과 검증은 어려운 작업 중 하나이다. 만일 학습자의 만족성을 높이는 것을 목표로 제작된 시스템이라

면 이를 위해 어떠한 설계요소가 고려되었는지를 객관적으로 확인할 수 있어야 한다. 예를 들어 고급 학습 콘텐츠를 제공하는 웹 시스템이라 할지라도 접근이 용이하지 않거나 학습 방식이 어렵다면 학습자의 만족도는 떨어질 수밖에 없다. 따라서 개발자는 접근성과 사용용이성을 강조하기 위해 설계의 어떤 부분에서 이러한 요구를 반영하고 있는지에 관한 명확한 설명을 해야 한다.

비기능 요구에 대한 명확한 명세나 표준스펙이 존재할 경우 적합성 테스트를 통해 명세와 구현의 일치성을 보일 수 있다. 그러나 비기능 요구에 대한 명확한 명세가 어렵거나 내부 요구사항들 간에 존재하는 일치성을 보여줄 목적이라면 적합성 테스트로는 한계가 있다.

본 논문은 효율성과 사용용이성과 같은 비기능 요구사항을 명세하는 단계에서부터 시작하여 설계가 비기능 요구사항의 정확한 반영이라는 내부 일치성을 보이는 검증 단계에 이르는 과정에서 발생하는 명세와 검증 문제를 다룬다.

2. 관련연구

최근 활발히 연구되는 확인에 관한 기법으로 프로세스 모형[5], 오토마타 모형[6] 등을 이용하는 기법이 있다. 이들 정형 모형을 이용하는 기법은 모형 자체가 갖는 높은 추상성과 엄밀함으로 인해 설계 및 구현환경에 독립적인 확인을 수행할 수 있다. 특히 [6]의 카탈리-올리버의 연구는 적합성 테스트에 대해서도 정형 모형이 훌륭히 사용될 수 있음을 보여준다.

모형의 검증 작업에서 주로 다루어지는 사항은 일관성과 완전성, 그리고 정확성의 입증이다. 특히 코드에 대한 테스트는 가장 많이 활용되는 검증 방법으로 설계와 구현의 일치성을 확인하는 중요한 기법이다. 최근 객체지향 테스트와 관련하여 웹 어플리케이션을 객체, 행위, 구조적 관점에서 분석하는 방법[7], 웹 서버에 대한 상태기반의 테스트 데이터를 찾아내는 방법[8] 등이 연구되고 있다.

기존의 다양한 확인과 검증에 관한 연구에도 불구하고 비기능 요구를 대상으로 한 검증 연구는 상대적으로 초보 단계이다. 이는 비기능 요구,

혹은 품질 요구에 대한 공식적인 개념 정의가 최근 4, 5년 전 ISO 9000:2000, ISO/IEC 9126-1과 같은 표준문서를 통해 이루어졌다는 사실을 통해서도 미루어 알 수 있다.

기존의 연구와 관련하여 본 논문이 웹 기반 학습 시스템의 비기능 요구에 대한 구현 검증 문제에 접근하기 위해 사용한 기법은 다음 두 가지에 기반을 둔다. 첫 째, 명확한 비기능 요구의 표현을 위해 정과 닉슨[10]이 제안한 목표지향 비기능 요구 명세방식의 개념을 이용한다. 둘째, 시스템의 설계와 구현 환경이 UML과 Java 언어임을 고려하여 [12]에서 제안한 객체지향 테스트 기법을 적용한다.

3. 비기능 요구의 표현

3.1 웹 기반 학습 시스템의 비기능 요구 파악

웹 기반 학습 시스템을 제작하는 과정에서 요구분석을 위해 유용하게 사용할 수 있는 지침으로 한국교육학술정보원의 교육용 콘텐츠 표준 개발 방법론[11]이 있다. 이 문서는 4 개의 개발 단계로 기획, 설계, 제작, 그리고 평가 단계를 정의한다. 기획 단계는 요구사항 정립을, 설계 단계는 학습객체에 기반한 주제 설정, 교수/학습 전략 설계, 학습 흐름도 작성, 화면 설계, 메타 데이터 작성, 스토리보드 작성과 같은 작업을, 개발 단계는 SCORM 위주의 학습 객체를 이용한 구현을, 평가단계는 평가 항목을 사용하는 구현 평가를 주요 작업으로 규정한다.

웹 기반 학습 시스템에 대한 기획단계의 품질 요소 정의 항목을 살펴보면 <표 1>과 같다.

<표 1> 품질요소항목

관심 요소	사용자 측면	품질항목
성능	빠른 접근이 가능한가? 보안은 충분한가?	접근성 보안성
설계	정확한 기능이 정의되었는가? 기능의 검증은 가능한가?	정확성 검증성
운용	운용은 수월한가?	운용성

	콘텐츠의 재사용이 가능한가?	재사용성
효용성	학습효과는 얼마나 증대했나? 다양한 콘텐츠를 제공하는가? 사용하기는 쉬운가?	효용성 다양성 사용성

[11]의 문서에서 권고하는 평가 단계의 평가 항목의 구성은 <표 2>와 같다. 이 표에 나타나는 평가 항목은 대부분 정성적이고 주관적인 판단을 해야만 하는 문항임을 알 수 있다.

<표 2> 교수설계 평가 항목의 예

교수설계평가 평가 항목		적합/ 부적합
학습	학습위치 이동이 용이한가?	
	애니메이션 자료나 동영상 자료는 학습진행속도에 맞게 동작하는가?	
개별적 교수	학습자의 학습성취도에 의해 적용된 학습자료를 제공하는가?	
	학습자의 선수지식을 고려하여 상호작용적인 미디어를 적용하는가?	
	학습 평가목적에 타당성이 있는가?	
	학습평가방법에 다양성이 있는가?	
	평가결과 활용에 다양성이 있는가?	

3.2 목표지향 요구표현과 설계

목표지향(goal oriented) 접근이란 시스템의 성능, 신뢰성, 사용용이성 등 비기능적 요구사항을 파악하는 방법으로 정과 닉슨에 의해 처음 명명되었다. 이 방법은 각각의 비기능 요구를 목표로 설정하여 이를 충족시키기 위해 기존의 문제 분할 및 점령 방식의 목표 분해를 하도록 권고한다. 예를 들어 개발자는 웹 시스템을 통해 학습자의 흥미를 유발시키는 목표를 설정할 수 있으며 이를 충족시키기 위해 설계에서는 다양한 색상을 사용할지 혹은 움직이는 폰트를 제공하는 자바 애플릿을 사용하거나 플래시 애니메이션을 사용할지를 세부적으로 결정하는 방식이다.

목표지향 명세 작성하는 과정에서 정과 닉슨이 제시한 다음과 같은 3 가지 사항을 결정한다.

목표정의: 시스템의 목표는 품질 요구사항을 의미하는 것으로서 설계에 고려해야 할 제약사항을 표현한다. 만일 하나의 목표가 너무 추상적이

거나 큰 경우 다수의 세부 목표로 재구성될 수 있다.

목표 간의 관계: 하나의 목표에 대해 예측 가능한 다른 목표들 간의 관계를 서술한다. 목표 간에는 보완, 갈등, 협력관계가 정의된다.

목표의 정제: 하나의 목표를 이루기 위해 긴밀한 관련이 있는 기능들을 파악하여 나열한다. 기능이 구체적으로 표현되는 과정에서 이를 구현하는 설계 모듈 혹은 함수들이 열거된다.

목표지향 서술방법은 요구사항을 파악하는 단계에서 적용되는 개념이다. 만일 객체지향 설계 기법과 같은 방법에 목표지향 요구명세를 접목시키기 위해서는 기존의 설계 표기법과 개념을 확장해야 할 필요가 있다.

목표를 표현하는 가장 적절한 방법으로 품질 요구의 구체적인 내용이 포함되도록 유스케이스를 확장하여 사용하는 것이다. 첫 번째 확장은 시나리오를 대신하는 시퀀스 다이어그램 상에 스

테레오타입 <<quality>>을 도입하여 품질관련 함수를 명시적으로 표현하는 것이다. 이 표기는 <그림 2>에서와 같이 품질관련 이벤트가 어느 모듈에서 어떤 순서에 의해 작동되는지를 보여준다.

두 번째 확장은 품질조건 서술부의 도입이다. 유스케이스에 나타나는 사전조건과 사후조건 의 용도가 특정 기능의 작동 시 반드시 참으로 만족해야 할 제약조건이라면 품질조건 역시 그 기능의 작동 시 반드시 참으로 만족시켜야 할 품질조건이라 정의한다. 따라서 이들 제약 조건들을 병치시키는 것은 자연스러운 생각이라 볼 수 있다. <표 3>은 강의참여에 대한 품질 조건이 어떻게 서술되는지를 보여준다.

유스케이스에 서술된 품질 조건은 개발자가 달성해야 할 지역적 목표를 표현하며 이러한 목표는 적절한 이벤트와 이의 인터랙션을 통해 달성될 수 있다.

<표 3> 강의참여 유스케이스

유스케이스 ID	UC_LECTURE	유스케이스 명	강의참여
기능명	강의참여		
설명	가입한 회원에 대해 효과적으로 강의를 들을 수 있게 한다.		
관련액터	학습자		
사전조건	1. 로그인이 되어있어야 한다. 2. 이전 강의가 완료되었는지를 체크한다. 3. 강의를 수강할 수 없는 과락의 횟수가 3회 이상인 경우 분 강의를 수강 할 수 없다.: 과락의 조건 > 각 단원 수강 후 시험결과가 60점 미만으로 보충수업을 수강한 후 재시험에 응시했으나 다시 60 점 미만을 받은 경우.		
사후조건	1. 강의를 듣고 나면 강의진도상태가 갱신되어야 한다. 2. 특정 단원에 포함된 모든 페이지를 수강해야 한다. 3. 최종 페이지에서 단원을 마친 후 시험을 통해서 60점 이상을 받아야만 해당 단원을 수강 완료한 것으로 한다. 만약 특정 단원의 모든 페이지를 Open 하지 않은 상태에서 로그아웃하면 강의진도는 완료상태가 되어서는 안된다.		
품질조건	강의참여는 학습자가 효과적으로 강의에 참여하도록 유도하고 학습자 위주의 사용성을 강조해야 한다.		
기본흐름	1. 액터가 강의참여 메뉴를 클릭한다. (A 1) 2. 시스템은 강의목차를 표시한다. 3. 액터가 강의목차 중 수강할 강의를 선택한다. (A 2) 4. 시스템은 액터가 선택한 강의를 화면에 표시한다. 5. 액터가 모든 내용을 수강 후 마지막 페이지에서 "다음"버튼을 클릭한다. 6. 시스템은 해당 단원의 수강여부를 체크한 후 테스트 페이지를 표시한다. 7. 액터는 시험이 완료되면 테스트 확인 버튼을 클릭한다. 8. 시스템은 자동으로 시험지를 채점하고 60점 이상이면, 최종 수강이 완료이므로, 강의목		

	차를 화면에 표시한다. (A 3)
대안 흐름	<p>A 1. 수업을 들었으나 시험을 안보고 나왔을 경우 강의참여를 선택, 강의 목차를 선택하면 바로 테스트 페이지를 보여준다.</p> <p>A 2. 만약 이전 단원이 완료되지 않았을 시에는 “이전 강의를 완료해야만 강의를 보실 수 있습니다.”라는 메시지 표시 후, 확인버튼을 클릭하면 강의목차를 화면에 보여준다.</p> <p>A 3. 자동채점 후 60점 미만인 경우는 별도의 보충학습을 통해서 다시 테스트 후 통과하면 해당 단원이 최종 완료된 것으로 한다.</p>

이벤트는 외부의 자극에 반응하는 특정 업무의 단위로서 목표가 구체적인 표현물로 정제되는 과정에서 도입된다. 유스케이스는 이벤트들의 적절한 조합에 의해 실현될 수 있다. 예를 들어 위의 유스케이스에 표현된 “강의참여는 학습자가 효과적으로 강의에 참여하도록 유도하여야 하고 학습자 위주의 사용성을 강조해야 한다.”라는 품질 제약을 만족시키기 위해서는 <표 4>와 같은 구체적인 기능이 추가로 정의된다.

<표 4> UC_LECTURE 품질 제약의 정제

품질 제약	정제된 기능
효과적인 강의 참여 유도	F1: 이전강의 완료여부확인
	F2: 60점 미만일 경우 3번의 재시험 기회 부여
학습자위주의 사용성 강조	F3: 시험평가의 실시간 확인
	F4: 잘못된 선택에 대한 팝업 가이드 표시
	F5: 이전 강의참여 활동에 대한 쿠키 유지

이벤트 간의 상호작용은 UML의 시퀀스 다이어그램으로 표현된다. 시퀀스 다이어그램은 객체들끼리 주고받는 메시지의 순서를 시간의 흐름에 따라 보여주므로, 시간에 따른 제어 흐름이 명확히 나타난다.

<그림 2>는 강의참여 진행에 대한 시퀀스 다이어그램을 보여준다. 이 다이어그램에는 이벤트를 책임지고 있는 모듈이 표현된다. 예를 들어 “강의리스트 중 수강중인 강의 선택” 이벤트는 “강의리스트 GUI” 모듈에서, “사용자정보와

LectureID전송” 이벤트는 사용자강의 페이지모듈에서 담당함을 알 수 있다.

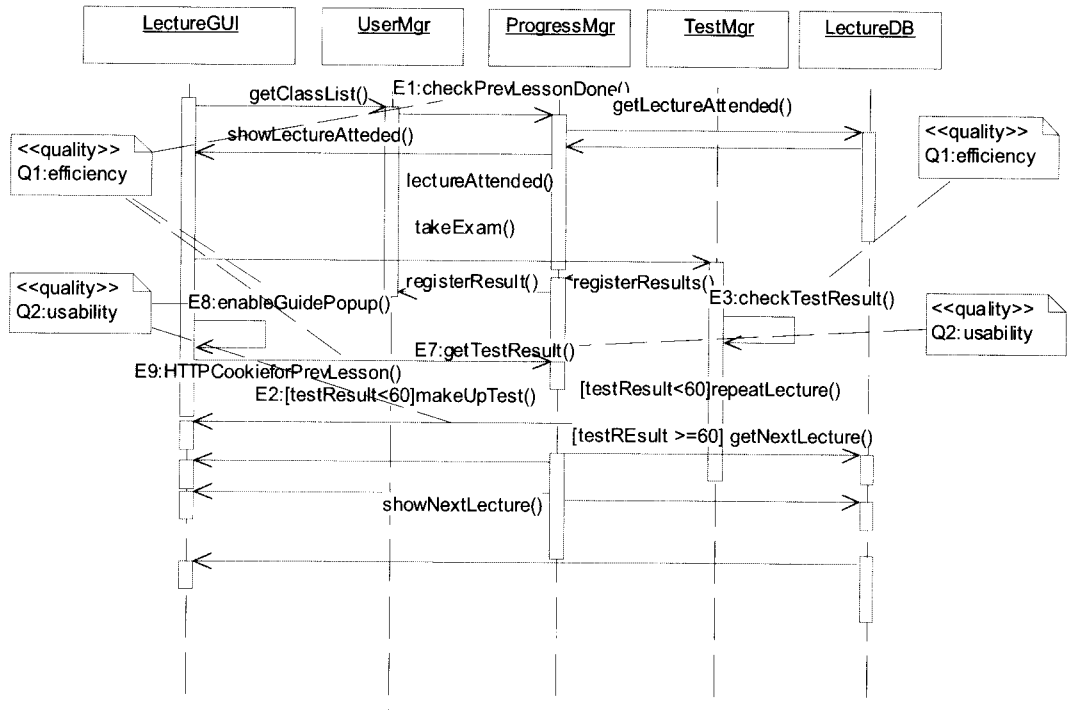
3.3 테스트 설계 절차

목표지향 명세는 품질과 관련한 비기능 요구사항 명세를 담고 있다. 본 논문에서 수행하는 구현 테스트란 설계 모듈이 시스템의 목표로서 품질 요구를 만족시키기 위해 얼마나 적절히 구성되었는가를 확인하는 테스트를 말한다. 따라서 이 작업은 품질 제약 사항을 시스템 서비스에 반영할 수 있는 모듈이 실제 작동하고 있는지를 객관적으로 살펴볼 수 있도록 한다.

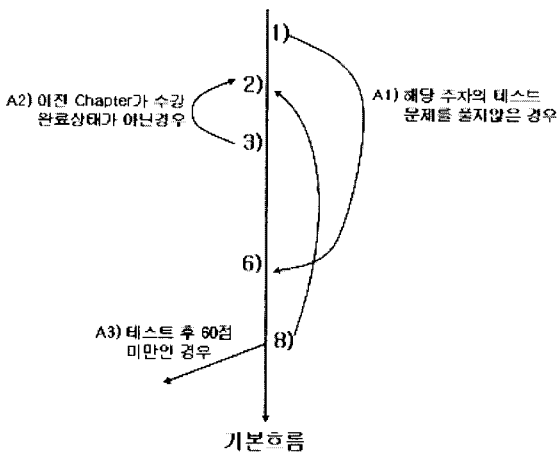
본 논문에서 제안하는 테스트는 크게 다음과 같은 6 개의 단계를 통해 수행된다.

- 단계1) 품질제약에 종속되는 이벤트 목록작성
- 단계2) 시퀀스 다이어그램 분석을 통한 해당 이벤트 흐름도 작성
- 단계3) 이벤트 흐름도 커버리지 분석
- 단계4) 목표-구현 이벤트 목록 작성
- 단계5) 목표-구현 이벤트 목록에 기초한 테스트 케이스 생성
- 단계6) 테스트 실행 및 결과 분석

구현 테스트를 설명하기 위해 앞서 살펴본 강의참여 유스케이스 경우에 적용하면 다음과 같다. 먼저, <표 4>를 통해 파악된 이벤트와 이를 반영한 <그림 2>의 강의참여 시퀀스 다이어그램의 분석을 통해 <그림 3>과 같은 이벤트 흐름도를 작성한다.



<그림 2> 강의참여 시퀀스 다이어그램



<그림 3> 강의참여 유스케이스의 이벤트 흐름

이 단계에서 파악되는 이벤트 흐름 정보는 테스트 시나리오 설계에 직접 이용될 수 있다. 예를 들어 <표 3>의 UC_LECTURE 유스케이스에는 학습자가 정상적으로 해당 단원의 강의 수강을 완료하는 경우(기본흐름)와 그렇지 못한 경우

(대안흐름)가 존재한다. 이를 바탕으로 <그림 3>과 같은 기본흐름과 대안흐름을 도식화하며 이로부터 테스트에 사용할 수 있는 시나리오를 유도한다. 이 예의 경우 설정된 4 개의 시나리오는 다음과 같다.

- S0: 정상적인 수강완료 - 기본흐름
- S1: 해당 주차의 수업을 성공적으로 듣고, 테스트 문제를 풀지 않았을 경우 (A1) 대안흐름
- S2: 이전 단원이 수강완료 상태가 아닌 경우 (A2) 대안흐름
- S3: 테스트 후 60점 미만인 경우 (A3)대안흐름
- S4: 보충학습 실시 (A3)대안흐름

테스트 시나리오를 결정한 후 각 테스트 시나리오 별로 프로그램이 해당 경로를 수행할 수 있도록 입력 값들을 결정한다. <표 5>는 강의참여 유스케이스의 각 테스트 시나리오 별로 주어저야 하는 입력과 예상결과를 기술한 것이다.

올바른 테스트가 되기 위해서는 품질 목표를 충족시키기 위해 정의된 이벤트를 추적할 필요가 있다. 시스템 개발 과정에서 설정되었던 품질조

건은 <표 3>에 나타나며 이를 살펴보면 다음과 같다.

- Q1: 효과적인 강의참여 유도
- Q2: 학습자 위주의 사용성 강조

각각의 품질조건은 이를 구현하기 위한 이벤트와 연결될 수 있다. 이러한 이벤트를 품질이벤트라 하며 품질조건과 품질 이벤트의 관계는 <표 6>에 정의된다.

<표 5> 강의참여 유스케이스의 테스트 케이스

테스트 케이스 ID	시나리오	입력				예상결과	실행결과	특수조건
		"강의참여" 메뉴	강의목차	마지막 페이지의 "다음" 버튼	시험후 "확인" 버튼			
TC 1	S0	클릭	클릭	클릭	클릭	정상적인 수강완료 후시험통과		60점 이상
TC 2	S1	클릭	클릭	클릭	없음	해당 페이지표시		
TC 3	S2	클릭	클릭	없음	없음	수강미종료 메시지 출력, 목차 표시		
TC 4	S3	클릭	클릭	클릭	클릭	재시험 실시		60점미만
TC5	S4	클릭	클릭	클릭	없음	보충학습실시		60점 미만

<표 6> 목표 구현 테스트를 위한 이벤트 목록

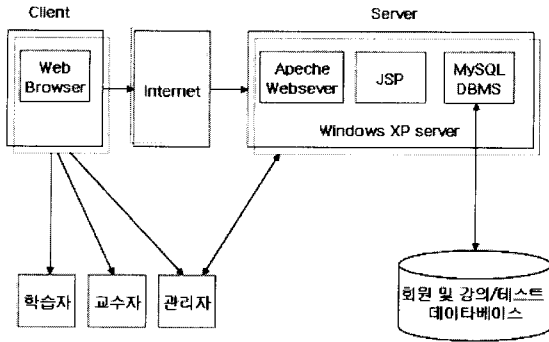
품질조건	품질이벤트 명	구현모듈명	설명
Q1: 효과적인 강의참여 유도	E1: checkPrevLessonDone()	LectureGUI	F1:이전강의 완료여부의 확인
	E2: makeUpTest() E3: checkTestResults()	TestMgr	F2: 60점 미만일 경우 3번의 재시험 부여
	E8:EnableGuidePopUp()	TestMgr	
Q2: 학습자 위주의 사용성 강조	E7:getTestResults()	TestMgr	F3: 시험평가의 실시간 확인
	E8:EnableGuidePopUp()	LectureGUI	F4:잘못된 선택에 대한 팝업 가이드 표시
	E9:HTTPCookieforPrevLesson()	LectureGUI	F5: 이전 강의참여 활동에 대한 쿠키 유지

어떤 함수들이 품질 이벤트로 작동하는지는 알아내기 위해서는 이미 정의된 함수 목록을 이용할 수도 있지만 시퀀스 다이어그램의 분석을 통해서도 가능하다. 경우에 따라서는 하나의 이벤트가 여러 개의 품질목표에 관련될 수 있으며 EnableGuidePopUp() 함수는 그러한 예이다.

어 교육을 위한 웹 기반 학습 시스템이다. 테스트의 대상이 되는 시스템의 구성도는 <그림 4>와 같다. 학습자는 로그인 과정을 거쳐 단원별로 수강할 수 있는 메뉴를 선택하게 되며 각 단원의 끝에서는 객관식 시험을 치른다.

4. 대상 시스템 구현 및 테스트

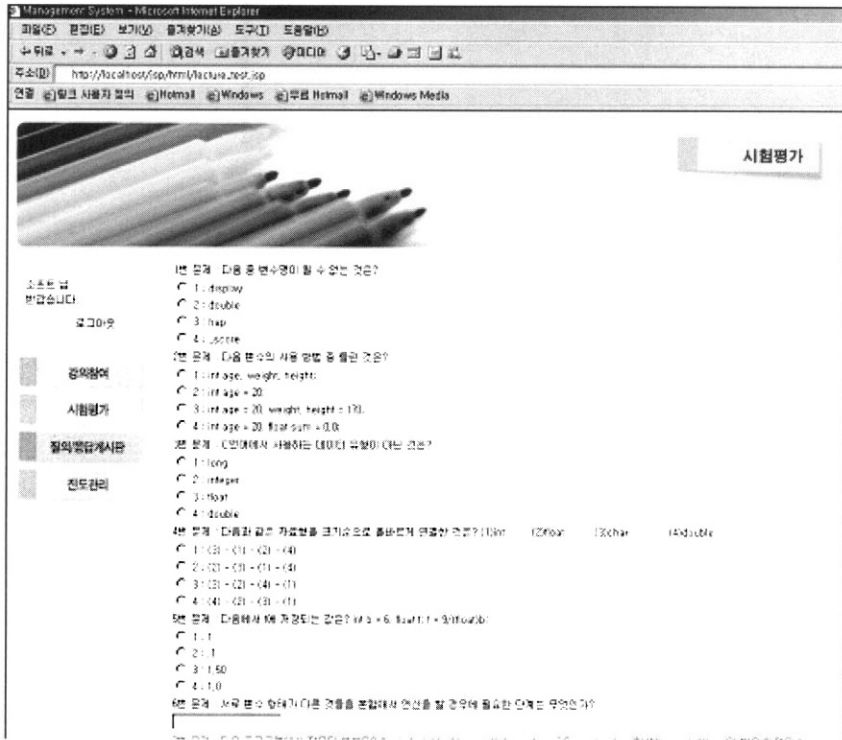
본 논문에서 소개되는 목표지향 명세에 관한 구현 테스트를 위해 구축된 대상 시스템은 C 언



<그림 4> 대상 시스템의 구성

<그림 5>는 정상적으로 일반강의를 참여한 후 시험평가를 수행할 수 있도록 하는 실행화면이다. 대상 시스템을 테스트하기 위해서는 앞서 살펴본 유스케이스와 각 유스케이스에 대응하는 시퀀스 다이어그램을 이용하는 테스트 케이스의 설계가 필요하다.

이를 위한 테스트 케이스의 설계에는 앞서 살펴보았던 <표 6>의 유스케이스 테스트 케이스 목록표와 <표 7>의 목표-구현 이벤트 목록을 필요로 한다.



<그림 5> 시험 치루기 화면

테스트 케이스의 선정은 먼저 각 시나리오에 대해 수행 과정에서 만족시킬 품질 조건을 목표-구현 이벤트 목록으로부터 선정한다. 선정된 이벤트들은 최종 테스트 목록에 개별 표시되며 각 품질 이벤트의 수행여부는 유스케이스에 바탕을 둔

테스트 목록과 통합되어 나열된다.

<표 6>은 강의참여 유스케이스의 각 시나리오 별로 주어져야하는 입력과 예상결과, 그리고 테스트를 통한 실행결과를 보여준다.

<표 6> 강의참여 유스케이스의 테스트 케이스 사례

TC6	Q1,Q2	S0	클릭	클릭	클릭	클릭	E1, E9	수강완료후 시험통과
TC7	Q1	S1	클릭	클릭	클릭	없음	E1, E8	해당 페이지 표시
TC8	Q2	S2	클릭	클릭	없음	없음	E8, E9	수강 미종료 메시지 출력, 강의목차 표시
TC9	Q1,Q2	S3	클릭	클릭	클릭	클릭	E2, E3, E7	수강완료후 점수60점 미만
TC10	Q1	S4	클릭	클릭	클릭	없음	E1, E8	보충학습용 페이지 표시

<표 7> 테스트 수행 목록표

테스트 케이스 ID	테스트 케이스명	예상결과/예상품질 이벤트	수행 결과	통과여부
TC6	학습자의 정상적인 학습참여	정상적인 강의화면/ 테스트 화면출력	정상적인 강의화면 테스트 화면출력	통과
		checkPrevLessonDone() HTTPCookieforPrevLesson()	해당 이벤트 메시지 출력	Q1,Q2 통과
TC7	강의 진행 상태 중 다음페이지로 전환	다음페이지 출력	다음페이지 출력	통과
		checkPrevLessonDone() EnableGuidePopUp()	해당 이벤트 메시지 출력	Q1 통과
TC8	수강 마지막페이지에 대한 테스트	해당 학습자에 따른 테스트 문제를 보여줌	학습자에 따른 테스트문제 표시	통과
		EnableGuidePopUp() HTTPCookieforPrevLesson()	해당 이벤트 메시지 출력	Q2 통과
TC9	테스트 문제풀이	강의 리스트 또는 보충화면 출력	학습자에 따른 테스트문제 표시	통과
		makeUpTest() checkTestResults() getTestResults()	해당 이벤트 메시지 출력	Q1,Q2 통과
TC10	보충학습	보충학습 화면출력	보충학습 화면출력	통과
		checkPrevLessonDone() EnableGuidePopUp()	해당 이벤트 메시지 출력	Q1 통과

<표 7>은 <표 6>의 입력 시나리오와 <그림 2>의 시퀀스 다이어그램에 나타난 이벤트를 고려한 테스트 케이스 선정표이다.

테스트가 성공적으로 수행되었는지를 확인하기 위해서는 각 이벤트가 작동될 때 이를 알려줄 수 있는 탐침(probe)을 이용한다. 테스트 실행을 위해서는 각 이벤트가 발생할 때 마다 생성되는 탐침을 기록하는 로그파일을 웹 서버에 두어서 객체들 사이의 주고받는 메시지에 대한 이벤트 정보를 확인할 수 있게 하였다. 이 로그파일을 분석함으로써 테스트를 검증할 수 있다. <그림 6>은 log.txt 파일에 기록된 정보를 보여주고 있다. 학습자 아이디가 'selab' 인 사용자가 강의참여에

대한 로그파일 내용이다. 1강 수강 후 2강의 수강 흐름의 이벤트들에 대한 함수호출 및 정보들을 보여주고 있다.

다른 예로서 사용자 로그인을 처리하는 유스케이스에 근거한 테스트의 케이스 목록은 <표 8>에 나타난다. 이 역시 블랙박스 테스트에 품질조건 확인을 함께 수행하는 경우를 보여준다. 이 경우 앞서 살펴본 <표 7>과 같은 이벤트의 작용을 전제로 하는 경우이지만 활성화된 특정 이벤트 명을 출력하여 확인하는 경우가 아니라 작동 결과만을 확인함으로써 테스트 과정을 간소화시킨다.

```

574 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getLastLectureNumber 함수 호출.
575 쿼리문 : select idPass from lecture_test_log where lectureID=1 and numberID=1 select
576 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getUserLecturePage 함수 호출
577 쿼리문 : select lecturepage from member_lecture where numberID='sealab' and lectureID=1
578 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getUserLecturePage 함수 호출. 사용자 데이터 없음. 초기 데이터 입력.
579 쿼리문 : insert into member_lecture(memberID, lectureID, gradeLine, sidcount, lecturepage)
580 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getUserLecturePage 함수 호출. 초기데이터 입력후 nextPage값 : 1
581 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getLastLectureNumber 함수 호출.
582 쿼리문 : select nextLecturepage as lecturepage from lecture_study where lectureID=1
583 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getUserLecturePage 함수 호출. 사용자 데이터 없음
584 Sun Nov 27 16:40:33 JST 2005 >>> LectureHest.getUserLecturePage 함수 호출후 lastpage 값 : 4
    
```

<그림 6> log.txt 파일에 기록된 사용자 정보

<표 8> 등록 유스케이스에 대한 테스트 케이스 목록표

유스케이스 ID : UC_LOGIN					
목 적 : 시스템의 화면 테스트					
특수 요구 : 사용자 ID와 패스워드의 검증					
단계	입력	예상출력	결과	품질액션	품질 항목
시스템 메인화면에서 로그인					
1. 정상적인 사용자 ID와 패스워드로 로그인을 시도한다.					
a	사용자 ID필드 클릭	사용자 ID 필드 반전		N/A	
b	정상 사용자 ID 입력	사용자 ID 화면에 보임		입력목록 드롭다운	사용성
c	패스워드 필드 클릭	패스워드 필드 반전		N/A	
d	정상 패스워드 입력	화면에 보이지 않음		SHA1암호화	보안성
e	“로그인” 버튼을 클릭	로그인된 화면		N/A	
1. 사용자 ID와 패스워드 없이 로그인을 시도한다.					
a	“로그인” 버튼을 클릭	로그인 실패창이 뜬		새사용자 등록 버튼	사용성
b	“확인” 버튼 클릭	로그인 화면으로 리턴		N/A	
2. 정상 사용자 ID와 패스워드가 없이 로그인을 시도한다.					
a	사용자 ID필드 클릭	사용자 ID 필드 반전		N/A	
b	정상 사용자 ID 입력	사용자 ID 화면에 보임		입력목록 드롭다운	사용성
c	“로그인” 버튼을 클릭	로그인 실패창이 뜬		패스워드조희 링크	사용성
d	“확인” 버튼 클릭	로그인 화면으로 리턴		N/A	
3. 정상 사용자 ID와 틀린 패스워드로 로그인을 시도한다.					
a	사용자 ID필드 클릭	사용자 ID 필드 반전		N/A	
b	정상 사용자 ID 입력	사용자 ID 화면에 보임		입력목록 드롭다운	사용성
c	패스워드 필드 클릭	패스워드 필드 반전		N/A	
d	틀린 패스워드 입력	화면에 보이지 않음		SHA1암호화	보안성
e	“로그인” 버튼을 클릭	로그인 실패창이 뜬		패스워드조희 링크	사용성
f	“확인” 버튼 클릭	로그인 화면으로 리턴		N/A	
4. 사용자 ID의 입력없이 정상적인 패스워드로만 로그인 시도한다.					
a	패스워드 필드 클릭	패스워드 필드 반전		N/A	
b	정상 패스워드 입력	화면에 보이지 않음		SHA1암호화	보안성
c	“로그인”버튼 클릭	로그인 실패창이 뜬		N/A	
d	“확인” 버튼 클릭	로그인 화면으로 리턴		N/A	

본 논문에서는 웹 기반 학습 시스템을 개발하는데 적용할 수 있는 검증 기법의 하나로서 사용 용이성, 효율성과 같은 비기능 요구 사항들이 정

확히 구현되었는지를 확인할 수 있는 테스트 방법을 제안하였다. 본 논문에서는 품질 요구가 구현되는 과정을 추적하여 얻은 정보를 테스트에 이용한다는 측면에서 명세간의 일관성, 즉 비기능 명세와 설계 문서, 그리고 설계 문서와 구현 코드 간의 일치성을 보여줄 수는 검증 방법이라 하겠다. 이러한 측면에서 본 연구는 기존의 적합성 테스트와 화이트박스 테스트와는 목적과 방법 측면에서 구분되는 특징을 갖는다.

본 연구를 통해 파악된 사항들을 살펴보면 다음과 같다.

첫 째, 요구명세단계에서 파악된 목표지향 명세의 내용을 설계에 반영하기 위해서는 설계단계의 배려가 필요하다. 파악된 품질조건을 분해하여 유스케이스의 사전조건, 사후조건과 함께 명세할 경우 구현 테스트의 기준으로 유용하게 사용됨을 알 수 있다.

둘 째, 설계 단계에서부터 비기능 요구와 직접 관련이 되는 함수를 분리 정의하는 것은 테스트를 위한 추적 리스트 작성에 큰 도움을 준다. 이것은 또한 향후에 시스템을 확장하거나 변경할 때 중요한 변경 정보를 제공할 수 있다.

셋 째, 유스케이스의 시나리오, 그리고 시퀀스 다이어그램에 품질관련 이벤트 항목을 포함시킴으로써 기능 테스트의 수행 시 추가의 노력을 들이지 않더라도 품질관련 요구사항이 구현과정에 어떻게 반영되었는지를 쉽게 파악할 수 있도록 한다.

본 논문에서 제안한 검증 기법이 효과적으로 적용될 수 있는 범위는 품질 관련 함수가 명확히 정의될 수 있는 경우에 한정한다. 따라서 하드웨어 환경에 종속되는 품질 특성들, 예를 들면 강인성, 고장 감내성과 같은 조건들, 신뢰성과 같이 특정한 몇 개의 함수로는 구현시킬 수 없는 부분들에 관한 검증은 다른 접근을 필요로 한다. 또한 유지보수성과 같이 코딩 스타일에 영향을 받는 부분 또한 검증 대상이 될 수 있는지에 관해서는 향후 연구가 필요하다.

습방법과 학습과제 유형이 학업성취도 및 학습태도에 미치는 영향”, 컴퓨터교육학회 논문지 5권 2호, 2002.

- [2] Badrul H. Khan, “Web Based Instruction : What is it and why is it?”, Educational Technology Publications, pp. 5 18, 1997.
- [3] Frantz F. “A Taxonomy of Model Abstraction Techniques” Proc. of 1995 Winter Simulation Conference, pp. 1413-1420, 1995
- [4] 최은만, “웹 기반 소프트웨어의 시험 및 검증 기술”, 정보과학회지 4권 3호, pp. 139 150, 2001.
- [5] Tasiran, S. Falloh, F. 외, “A Functional Validation Technique: Biased-Random Simulation Guided by Observability-based Coverage” Proc. Int’1 Conf. on Computer Design, pp.82-88, 2001
- [6] Cardell-Oliver, R. “Conformance Test Experience for Distributed Real-time Systems” Proc. Symposium on Software Testing and Analysis, pp.156-163, 2002
- [7] D. Kung, 외, “An Object-oriented Web Test Model for Testing Web Applications” Proc. First Asia Pacific Conference on Quality Software, pp.111 120, 2000.
- [8] 강재성, 윤광식, 오승욱, 권용래, “웹의 상태기반 기능시험 기법”, 정보과학회 봄 학술발표논문집 27권 1호 pp. 501 503, 2000.
- [9] 한동수, 정인상, “소프트웨어 테스트 입문”, 한국소프트웨어공학협회, 2004.
- [10] L. Chung, B A. Nixon, E Yu, J. Mylopoulos, Non Functional Requirements in Software Engineering, Kluwer Academic Publishers, 2000
- [11] 한국교육학술정보원, 교육용콘텐츠 표준 개발 방법론: SCORM 표준적용, 연구자료 RM 2002 28, 2002
- [12] Binder, R, Testing Object-oriented Systems, Addison-Wesley, 1999

참 고 문 헌

- [1] 방성준, 이성근, “웹을 활용한 자기 주도적 학



서 등 수

1987 중앙대학교 컴퓨터공학과
(이학사)

1994 영국 맨체스터대학
(전산학 박사)

1998~현재 성신여자대학교 컴퓨터정보학부
부교수

관심분야: 컴퓨터 교육, 소프트웨어 공학

E-Mail: dseo@sungshin.ac.kr



이 혜 리

2003 성신여자대학교
컴퓨터정보학부 (이학사)

2006 성신여자대학교
전자계산교육학과
(교육학석사)

관심분야 : 컴퓨터 교육, e-learning

E-Mail: lh10601@sungshin.ac.kr