

# 의료용 소프트웨어 시험 인증을 위한 지원도구의 설계 및 구현

홍정훈\*, 김혁주\*\*, 양해술\*,  
\*호서대학교 벤처전문대학원  
\*\*식품의약품안전청  
e-mail:jaspers74@naver.com

## Design and Implementation of Supporting Tool for Medical Software Testing Certification

Jung-Hun Hong\*, Hyuk-Joo Kim\*\*, Hae-Sool Yang\*  
\*Dept of Computer Science, Ho-Seo Venture University  
\*\*Dept of Korea Food & Drug Administration

### 요 약

의료기기 산업은 시장규모와 생산규모에서 미국이 전세계의 40% 이상을 차지하고 있어 국내의 모든 의료기기 제조업체에서는 미국시장진출을 목표로 꾸준히 노력하고 있는 실정이다. 그러나 의료기기는 인체에 직접 적용된다는 특성으로 인해 각 국가의 정부는 저마다 다른 엄격한 기준으로 자국내 시장진출을 규제하고 있으며 엄격한 관리제도를 유지하고 있다. 국내 최신의 전자 의료기기는 필수적으로 의료기기를 제어하기 위한 의료용 소프트웨어를 필요로 하므로 의료기기 산업의 발전은 의료용 소프트웨어 산업의 발전에 직결된다고 할 수 있다. 그러나 의료기기의 경우에는 인·허가를 위한 제도 및 절차가 갖추어져 활용되고 있는 반면, 의료용 소프트웨어의 경우에는 적합한 평가기준이 마련되어 있지 못하여 제도 및 체계구축의 필요성이 대두되고 있다. 본 논문에서는 국제표준(ISO/IEC 12119, ISO/IEC 9126)을 바탕으로 평가메트릭을 구축하여 의료용 소프트웨어의 정량적인 품질을 측정, 평가하여 품질측정의 결과를 토대로 품질을 인증하는 의료용용 소프트웨어 시험인증 지원도구를 설계 및 구현하였다.

### 1. 서론

의료기기 산업은 시장규모와 생산규모에서 미국이 전세계의 40% 이상을 차지하고 있어 국내의 모든 의료기기 제조업체에서는 미국시장진출을 목표로 꾸준히 노력하고 있는 실정이다. 그러나 의료기기는 인체에 직접 적용된다는 특성으로 인해 각 국가의 정부는 저마다 다른 엄격한 기준으로 자국내 시장진출을 규제하고 있으며 엄격한 관리제도를 유지하고 있다. 국내 최신의 전자 의료기기는 필수적으로 의료기기를 제어하기 위한 의료용 소프트웨어를 필요로 하므로 의료기기 산업의 발전은 의료용 소프트웨어 산업의 발전에 직결된다고 할 수 있다. 그러나 의료기기의 경우에는 인·허가를 위한 제도 및 절차가 갖추어져 활용되고 있는 반면, 의료용 소프트웨어의 경우에는 적합한 평가기준이 마련되어 있지 못

하여 제도 및 체계구축의 필요성이 대두되고 있다.

이런 문제점에 대한 방안으로 국내의 의료기기에 사용되는 소프트웨어의 안정성, 유효성에 대해서도 평가를 위한 가이드라인을 마련하여 제도화함으로써 의료 서비스의 질적 향상을 추진할 필요가 있으며 따라서 본 연구에서는 의료용 소프트웨어의 평가를 위한 기준을 개발하여 국내 의료용 소프트웨어 산업 전반에 적용할 수 있는 기반을 구축하고자 한다.[2]

### 2. 정형적인 시험절차

의료용 소프트웨어의 품질에 대한 인증절차는 크게 12단계의 작업으로 구성되어 있다. 평가제품에 대한 시험신청서를 작성한 후 이 제품에 대한 적합성 판별회의를 거쳐 시험합의서와 사전조사서, 시험기준안, 시험계획서를 작성한 후 점검표 작성 및 시

험을 거쳐서 테스트케이스 작성 및 시험, 그에따른 문제점 기록, 시험결과서 작성과 시험성적서 작성을 거쳐 최종적으로 제품인증이 이루어지게 된다. 인증 절차를 도표로 나타내면 다음 <표 1>과 같다.

<표 1> 의료용 소프트웨어 시험절차

인증절차	내용
시험신청서 작성	소프트웨어의 시험의뢰
적합성판별회의	시험여부 결정
시험합의서작성	신청자와 시험기관 간의 합의사항
사전조사서작성	시험제품 사전조사
시험기준안작성	시험적용/미적용 시험모듈 결정
시험계획서작성	시험내용, 조직, 환경
점검표작성 및 시험	가부결정에 필요한 측정사항에 대한 표
테스트케이스작성 및 시험	기능의 올바른 수행을 검사하기 위한 사례
점검표시험 및 테스트케이스시험에 따른 문제점기록	시험을 통해 발견된 문제점에 대해 기록
시험결과서작성	시험결과에 따른 점수의 단계적집계
시험성적서 작성	시험결과에 대한 전반적인 내용을 기술
제품인증	인증위원회의 종합적인 심의를 거쳐 최종인증

3. 품질평가 매트릭과 문서화 지침

의료용 소프트웨어 품질인증에 사용되는 매트릭은 크게 두가지로 구분할 수 있다. 소프트웨어 제품 평가에 있어서의 국제규격인 ISO/IEC 12119를 바탕으로 두고있는 소프트웨어의 일반적 품질요구사항과 소프트웨어 프로세스 파악에 있어서의 국제규격인 ISO/IEC 9126을 바탕으로 두고있는 소프트웨어 품질특성평가가 그것이다.[1]

<표 2> 품질특성

구분	품질특성명	내용
일반적 품질요구사항 ISO/IEC 12119	제품설명서	제품설명서에 관한 품질요구사항
	사용자문서	사용자문서에 관한 품질요구사항
	실행소프트웨어	실행소프트웨어에 관한 품질요구사항
소프트웨어 제품품질특성 ISO/IEC 9126	기능성	요구되는 기능을 제공하는 능력
	신뢰성	지정된 수준의 성능을 유지하는 능력
	사용성	쉽게 이해하고 배울 수 있게 하는 능력
	효율성	적은 자원으로 적절한 성능을 제공하는 능력
	유지보수성	개선·수정에 쉽게 대응할 수 있는 능력
	이식성	다른 환경(소프트웨어, 하드웨어)에서 운영될 수 있는 능력

4. 시스템 설계

본 시스템은 웹 환경을 기반으로 하는 웹 애플리케이션의 형태를 띠고 있다. 이는 사용자로 하여금 접근성을 용이하게 하기 위함이다.

4.1 개발환경

시스템의 개발환경은 다음과 같다.

- OS : Windows 2000 Professional
- Language : Java SDK 1.4.0\_03, J2EE 1.3, JSP
- DataBase : Oracle 9i
- Web Server : Apache Tomcat 5.0
- Framework : Jakarta Struts 1.0.2

4.2 데이터베이스 설계

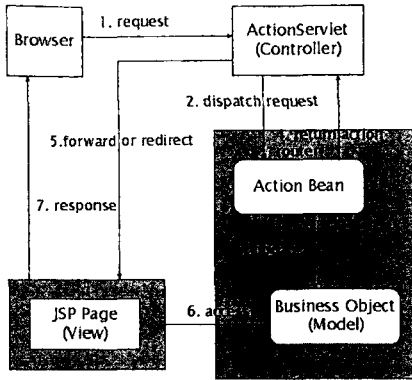
데이터베이스 설계는 <표 3>과 같이 전체 11개의 테이블로 이루어져 있다.

<표 3> 데이터 베이스 Table

테이블 명	설명
Company	제품인증을 의뢰한 업체관련 테이블
Product	인증의뢰된 제품관련 테이블
Users	사용자 관련 테이블
Meeting	회의관련 테이블
MetricValue	추출된 매트릭 측정값 관련 테이블
Main	품질 주특성 관련 테이블
Sub	품질 부특성 관련 테이블
Metric	품질 매트릭 관련 테이블
Input	품질 매트릭 입력치 관련 테이블
QualityGrade	품질 평가기준 관련 테이블

5. 구현

본 시스템의 구현은 (그림 1)의 스트러츠 프레임워크(Struts Framework) 구조와 같이 화면의 프리젠테이션 부분을 담당하는 JSP Page와 이벤트를 처리하는 서블릿(Servlet) 그리고 비즈니스 로직을 담당하는 ActionBean과 Model로써 데이터를 가지고 있는 FormBean들로 구분해 볼 수 있다. 이는 JSP 화면에서 이벤트를 발생시키면 그 이벤트에서 사용되는 데이터들은 FormBean이 담당하고 내부적인 처리는 ActionBean들이 FormBean을 호출함으로써 해당 데이터들을 처리한 후 처리한 내용을 화면에 전달하는 것이다.[6]



(그림 1) Struts Framework

5.1 시스템 설정

스트러츠 프레임워크는 애플리케이션에 필요한 컴포넌트들을 적재하기 위해서 설정 사항들을 정의하여 선언하는 방식의 설정파일을 제공한다. 설정파일은 XML 형식으로 이에 대한 정의는 (그림 2)와 같다.[7]

```
<struts-config>
<!-- ===== Data Source Configuration = -->
<data-sources>
<data-source
autoCommit="false"
description="MSE auth tool DataBase -Oracle 9i"
driverClass="oracle.jdbc.driver.OracleDriver"
maxCount="4"
minCount="2"
password="#####"
url="jdbc:oracle:thin:@211.192.249.###:1521:###"
user="#####"
/>
</data-sources>
<!-- ===== Form Bean Definitions = -->
<form-beans>
<form-bean name="productForm"
type="kr.co.microjava.afts.entity.ProductForm"/>
</form-beans>
<!-- ===== Global Forward Definitions = -->
<global-forwards>
<forward name="logon"
path="/jsp/logon.jsp"/>
</global-forwards>
<!-- ===== Action Mapping Definitions = -->
<action-mappings>
<action path="/product"
type="kr.co.microjava.afts.action.ProductAction"
name="productForm"
scope="request">
</action>
<action path="/company"
type="kr.co.microjava.afts.action.CompanyAction"
name="companyForm"
scope="request">
</action>
</action-mappings>
</struts-config>
```

(그림 2) struts-config.xml 설정파일의 일부

5.2 이벤트 처리

시스템의 동작에 있어서는 웹 애플리케이션 프레임워크인 스트러츠의 기본 동작원리에 따라 이벤트를 주고, 받게 되는데 JSP 페이지에서 이벤트를 요청을 하면 컨트롤러인 ActionServlet이 요청한 이벤트를 감지를 하고 실제 작업을 수행하는 ActionBean을 호출하여 작업이 이루어지게 된다. 이때 데이터를 서로 주고, 받는 수단으로는 ActionForm을 이용하게 되는데 ActionForm은 하나의 객체로 데이터의 전달에 있어 각각의 데이터에 대한 처리를 지양하고 객체 단위로 데이터를 담아서 처리하게 된다. 이렇게 서로 이루어지는 요청에 대한 응답은 struts-config.xml 설정파일에서 ActionServlet과 ActionForm의 Mapping에 의하여 각각이 처리해야할 데이터가 무엇인지를 정의하게 된다.

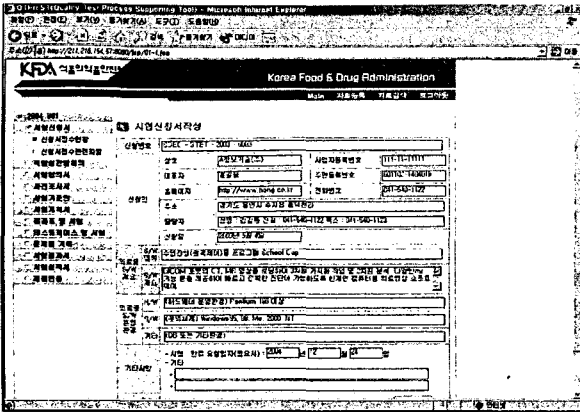
ActionServlet을 호출할 때는 Action 파라미터 (Parameter)를 같이 전송하게 되며 해당 이벤트를 처리하는 ActionServlet은 이 파라미터 값에 따라 해당 작업을 처리하게 된다. 이와 같은 모습은 (그림 3)과 같다.

```
public final class ProductAction extends Action
{
/**
* perform method override
* client의 http request를 수행하는 메소드
*/
.....
public ActionForward perform(ActionMapping mapping,
ActionForm form,
HttpServletRequest request,
HttpServletResponse response)
throws IOException, ServletException
{
if (action.equals("login"))
{
StringBuffer str = new StringBuffer();
try
{
conn = Util.getConnection(servlet);
ProductForm[] product = null;
Integer totalCnt = null;
.....
}
return new ActionForward("/jsp/mainTable.jsp?page="+page);
}
}
```

(그림 3) ActionServlet의 예

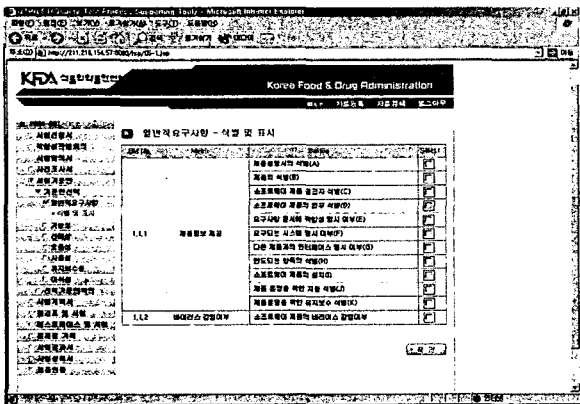
5.3 시스템 실행

본 시스템을 구성하는 12가지의 의료용 소프트웨어 시험절차 프로세스의 핵심기능을 살펴보면 다음과 같다. 다음의 (그림 4) 화면은 평가의뢰된 제품에 대한 상세설명을 기록하는 작업으로 해당제품에 대한 정보를 기록함으로써 해서 프로세스의 첫 번째 단계가 실행되어 진다.

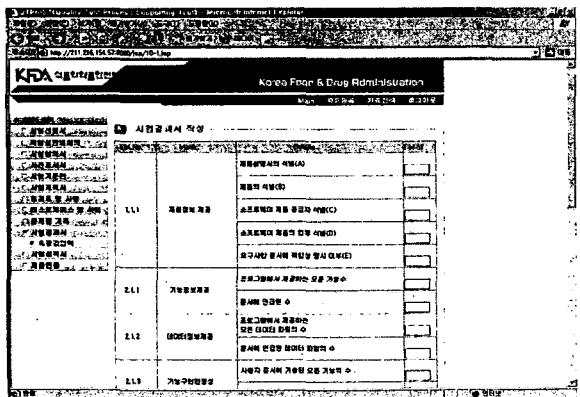


(그림 4) 시험신청서 작성

평가의뢰된 제품에 대한 시험신청서를 작성한 후에 제품에 알맞은 메트릭을 선정할 수 있다. 다음(그림 5)는 품질측정을 위한 각 항목별로 메트릭을 선택할 수 있도록 설계된 화면이다. 각 항목별로 메트릭을 선택한 후에는 선택된 전체 메트릭 항목을 확인할 수 있다.[8][9]



(그림 5) 품질측정 항목 선택



(그림 6) 측정된 메트릭값 입력

위의 (그림 6)은 이와 같이 선택된 항목들에 대한 입력의 기능을 제공하기 위한 것으로 실제 측정된 메트릭 값을 입력할 수 있다. 입력항목은 선택고된 항목들만 선별적으로 보여줌으로써 실제 작업진행을 효율적으로 수행할 수 있도록 하고 있다.[3][4][5]

6. 결론

의료용 소프트웨어 분야는 축적된 기술력을 바탕으로 해외수출선이 확대되고 있는 등 영업환경 또한 지속적으로 개선될 것으로 예상되고 있다. 앞으로 전문화된 의료 소프트웨어 시장에서 높은 인지도와 축적된 기술력을 바탕으로 향후에도 안정적인 외형 신장세를 지속할 것으로 전망된다. 이러한 의료용 소프트웨어 분야의 지속적인 성장을 지원할 수 있도록 의료용 소프트웨어의 품질향상을 통한 의료서비스의 질적향상은 의료분야에서 지향하는 발전방향이므로 의료용 소프트웨어 품질평가 기준개발에 관한 본 연구의 파급효과는 매우 클 것이라 예상되며 향후 지속적인 추가 연구관련 기관에 대한 기술이전을 통해 의료서비스의 질적 향상에 기여할 수 있을 것이다.

참고문헌

[1] Moller, K. H. and Paulish, D. J., "Software Metrics", Chapman & Hall(IEEE Press), 1993.  
 [2] Wallmuller, E., "Software Quality Assurance A practical approach", Prentice Hall, 1994.  
 [3] ISO/IEC 14598, "Information Technology - Software product evaluation - Part 5,6", 1997. 7.  
 [4] ISO/IEC 12119, "Information technology - Software packages-Quality requirements and testing", 1994  
 [5] ISO/IEC 9126, "Information technology - Software Quality Characteristics and Metrics - Part 1,2,3", 1997. 7.  
 [6] David M. Geary, Advanced JSP, PHPTR, Sun Microsystem Press,2001  
 [7] <http://www-106.ibm.com/developerworks/ibm/library/j-struts/>  
 [8] 양해술, 편용범, 이정림 유규하, "의료용 소프트웨어의 평가기준개발에 관한 연구", 한국정보처리학회 논문집, 2003. 8  
 [9] 이하용, 양해술, 이정림, 김혁주, "의료용 소프트웨어의 문서화 체계와 시험사례의 구축", 한국정보처리학회 논문집, 2004. 5