JSP Model 2 Architecture를 적용한 Linux 전자상거래 시스템의 설계 및 구현

○
서순모*, 정혜정**, 양해술*
*호서대학교 벤처전문대학원 컴퓨터응용기술
**평택대학교 정보과학부
bante97@hanmail.net*, jhjung@ptuniv.ac.kr**, hsyang@office.hoseo.ac.kr*

Design & Implementation of Linux EC System applying JSP Model 2 Architecture

Seo Soon-Mo*, Jung Hae-Jung**, Yang Hae-Sool*

* Graduate School of Venture, Hoseo University
** Dept. of Information Science, Pyungtaek University

요 약

전자상거래 시스템에 대한 관심이 고조되고 있다. 언론의 보도에 따르면 전자상거래 시스템을 운영하는 기업의 매출이 지속적으로 늘어나고 있다고 한다. 전자상거래 문화가 성숙해 가고 있는 것이다. 그러나 이러한 전자상거래 시스템들은 시간이 지날수록 고객의 지속적인 요구와 환경의 변화에 따른 유지 및 보수 즉 업데이트의 문제에 직면하게 된다. 현재 이루어지고 있는 비즈니스 로직과 프리젠테이션 계층의 통합에 따르거나 페이지(Page) 개념에 준한 시스템 설계나 구현은 EC시스템의 지속적인 성능 개선을 어렵게 하고 있다. 이에 따라 본 연구에서는 이러한 문제를 해결하기 위한 수단으로서 MVC모델에 기반한 JSP Model 2 구조를 적용하여 EC시스템을 구현하고자 한다.

1. 서 론

최근 전자상거래에 관한 관심이 매우 고조되고 있다. 최근 발표된 통계청의 "전자상거래통계조사결과"를 살펴보면 사업체 수 및 거래규모가 지속적으로 늘어났음을 알 수 있다. 구체적으로도 2001년 1/4분기에 비하여 2/4분기 사이버쇼핑몰 운영업체수는 83개(4.3%)늘었고, 분기별 매출액은 2/4분기 7,901억원으로 1/4분기 7,078억원대비 823억원(11.6%) 늘어났음을 알 수 있다[1]. 이러한 정보 외에도 G2B활성화를 위한 혁신계획이 수립되는 등 정부의 전자상거래에 관한 지속적인 관심 및 투자 그리고 실천으로 산업부문의 성장률도 시너지 효과를 창출하여 꾸준히 늘어나고 있음을 알 수 있다.

그러나 이러한 전자상거래의 산업적 측면의 발전에도 불구하고 병행적으로 따라오는 것은 시스템의 지속적인 유지 보수 및 개선에 관한 문제점이라 할 수 있다. 흔히들 얘기하는 "완벽한 것은 없다"라는 말처럼 전자상거래 시스템 또한 급조되고 장기적인 안목 및 투자를 바탕으로 설계되고 구현된 것이 미비한 현실이다. 이러한 문제는 최초 투자비에 맞먹거나 그 이상에 투자비를 요구하는 유지 보수비용을 생각하면 잘 이해된다. 때문에 전자상거래의 문화 및 산업적 성숙의 단계가 높아질수록 안정적이고 다이나믹한 유지보수를 가능케 하는 시스템의 요구가 시기적으로 요구되어지고 있다. 이에 따라 본 연구에서는 컴포넌트를 이용한 시스템의 설계 및 구축이 컴퓨터 소프트웨어 개발 패러다임의 큰 흐름으로 정착되고 이해됨으로 해서, 본격적인 전자상거래 시스템의 컴포넌트 기반 개발의 전 단계인 연구활동의 하나로, MVC모델 또는 MVC아키텍처라고 잘 알려진 시스템 개발방법모델을 적용한 Sun-microsystems Co.의 Server Side Application 개발도구인 JSP(Java Server Page)의 Model 2 Architecture를 통하여 전자상거래 시스템을 구현하고, 이러한 과정으로서 비즈니스 로직과 시스템의 통합 개발 패턴에 비해 각 레이어의 분리 개발방식이 가지는 장점을 부각 시켜 보고자 한다.

2. 관련연구

2.1 MVC 아키텍처(Architecture)에 기반한 JSP Model 1과 Model 2

MVC(Model-View-Controller) 아키텍처는 1970년대 후반부터 잘 알려져 있으며, 전통적인 OOP에 기반하여 출발하였다. 객체지향언어의 근간이라 할

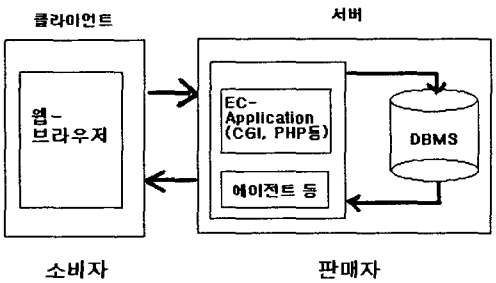
수 있는 Smalltalk에 관한 MVC아키텍처의 적용 방법이 널리 알려져 있다. 근래 들어 Sun-microsystems사에서 내놓은 자바(Java)언어를 이용한 Server-Side Java인 JSP(Java Server Page)는 이러한 MVC아키텍처를 지원하는 가장 대표적인 개발 도구로 주목받고 있으며, Java의 Swing은 MVC 아키텍처를 적용한 가장 유연하고 강력한 최초의 자바 애플리케이션의 UI지원 컴포넌트이다. MVC아키텍처는 전자상거래 시스템 개발모델로 적합한 것으로 받아들여진다.

2.1.1 MVC Architecture의 이해

MVC Architecture를 구성하는 3가지 요소는 다음과 같다.

- **MVC-모델(Model)**
- **MVC-뷰(View)**
- **MVC-컨트롤러(Controller)**

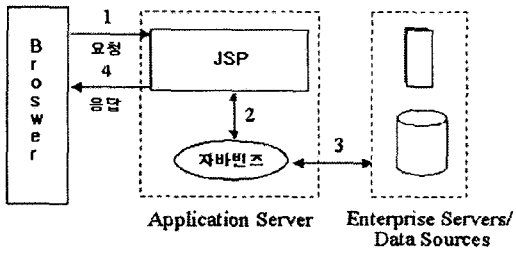
모델(Model)은 컴포넌트가 다루는 자료의 구조를 추상화한다. 즉, 컴포넌트의 모든 정보를 유지하는 기능을 가진다. 뷰(View)는 화면 출력을 담당하는 컴포넌트이다. 시각적인 표현들을 결정함으로써 색깔을 표현한다던가 폰트의 형태를 변경해 주는 등의 기능을 한다. 컨트롤러는 뷰와 모델의 제어(Control)를 담당하는 컴포넌트이다. 요구되고 입력되는 이벤트를 통해서 어떤 액션(Action)이 이루어 져야 하는지를 결정한다[2].



(그림 1) 전통적인 전자상거래 개발 모델

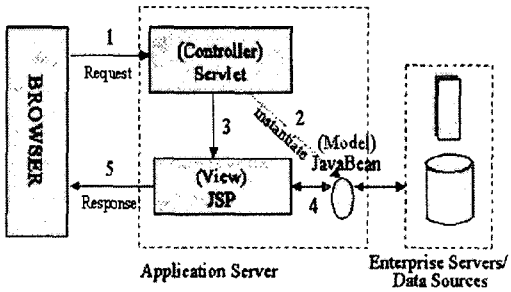
소비자의 요구가 고도화됨으로 인해서 프리젠테이션 및 비즈니스 로직이 매우 복잡하게 설계되는 경향을 보이는데, 이러한 패턴의 설계 및 개발은 팀(Team)내 서로 다른 영역의 전문가 사이의 의사 소통을 어렵게 하고 결과적으로 생산성을 저하시키는 문제점을 불러온다.

JAVA 진영의 JSP는 MVC구조를 반영한 개발 모델인데 이는 JSP 모델 1과 JSP 모델2가 있다. 단순한 애플리케이션 개발에 적합한 모델 1은 [1](그림 2)에 나타난 바와 같은 구조를 지닌다.



(그림 2) JSP Model 1 Architecture[2]

또한 좀 더 복잡한 애플리케이션의 설계 및 구축에 적합한 모델 2는 요청과 응답을 분리 적용 개발하는 구조로서 (그림 3)에 표현된 바와 같다.



(그림 3) JSP Model 2 Architecture[2]

3. 전자상거래 시스템의 설계

전자상거래 시스템을 구성하기 위해서는 다양한 기능의 설계 및 구현이 필요하다. 상품의 등록과 관리, 고객의 관리, 재고 및 판매관리, 지불 및 인증 시스템, 카탈로그 시스템 등 다양한 구성요소의 충분조건이 선행되어야 전자상거래 시스템으로서의 기능을 수행할 수 있다. 본 연구에서는 이러한 전자상거래를 이룰 수 있는 가장 가벼운 경량모델을 설정하고 이를 통해 본 연구가 지향하는 목적을 달성하고자 한다.

3.1 회원관리 및 인증

회원 관리는 지속적인 고객의 참여를 유도하고 전자상거래 시스템을 운영함에 있어 각종 자료 및 정보를 유출해 내는데 필수 불가결한 장치이다. 일반적

으로 회원의 가입을 유도하고 정회원과 비회원으로 구분하여 시스템을 운영하며, 정회원의 경우에 다양한 서비스의 제공이 가능함을 인지시켜 고객의 회원 가입을 유도한다. 기본적인 회원의 DB Table은 <표 1>에서 보는바와 같다.

<표 1> 회원 테이블 구조

id	pwd	name	ju_1	ju_2	e-mail	tel	mobile	addr.	reg.
char	char	char	char	char	char	char	char	char	date

아이디와 비밀번호, 주민등록 번호1, 2, 전자우편, 핸드폰, 주소와 등록일자를 필수로 포함한다.

3.2 상품등록 및 관리

전자상거래 시스템의 구성 요건 중 핵심요소인 상품등록 및 관리를 위한 장치는 다양한 상품을 등록하고 각각의 카테고리별로 관리할 수 있으며, 소비자에게 상품의 정보를 적절하게 보여줄 수 있어야 한다. 이를 위해 상품 테이블을 작성하고, 카테고리 테이블을 작성한다. 상품 테이블에는 상품의 아이디, 카테고리 아이디, 제품명, 일반정보, 상세 정보, 가격, 이미지와 재고현황을 수록하여 상품을 관리 할 수 있도록 한다.

<표 2> 상품관리 테이블 구조

상품 id	카테고리	상품명	기본설명	상세설명	이미지	가격	재고량
int	char	char	text	text	char	int	int

3.3 결제 및 배송

전자상거래 시스템에서 제공하는 일련의 상품정보를 보고 고객이 구매를 원하는 경우 이러한 문제를 해결하기 위한 수단으로서 제품에 대한 지불 가능한 방식을 제공하고 고객으로 하여금 그에 대한 선택을 유도한다. 일반적으로 통용되고 있는 지불 방식은 인터넷 뱅킹을 통한 현금납부, 신용카드결제, 사이버머니를 이용한 결제 등이 있다. 하지만 본 시스템에서는 본 장치가 연구의 목적을 구현하는데 반드시 필요하지 않는 장치이므로 기능의 언급 등으로 갈음한다.

3.4 상품 카탈로그 시스템

제품마다 저마다의 데이터를 포함하고 있다. 이러한 데이터는 일련의 정렬 및 분리 등의 작업과정을 거치지 않으면 제품마다 가지는 특징으로 인해 고객에게 매우 혼란스러움을 안겨줄 수 있다. 이러한 문제를 해결하기 위하여 일정한 기준을 적용한 전자카탈로그를 설정하고 구현하여 고객에게 제공함으로써, 제품을 올바르게 이해하고 실질적인 구매를 유도할 수 있는 수단으로써 많이 이용되고 있는 상품의 카탈로그를 구현하고자 한다. 기본적으로 3.2항에 나타난 상품의 기본정보를 화면에 일정한 형식으로 출력함으로써 상품 카탈로그의 구현을 이루어 보겠다. 즉, 상품의 등록번호, 상품명, 가격, 일반정보, 상세정보, 제품의 이미지, 현 재고량을 화면에 출력하여 고객에게 제품에 대한 정보를 제공한다.

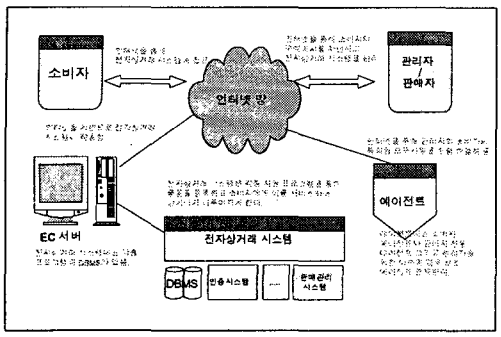
3.5 기타 지원장치

기타 전자상거래 지원장치로는 각종 게시판과, 주문내역, 장바구니 기능 등을 설정할 수 있다. 주문내역은 고객이 선정한 제품의 실질적인 구매 의사 표시를 한 내용을 보여주며, 장바구니 기능은 일반적으로 잘 알려진 고객이 관심을 갖고 있으며 구매결제과정을 거치지 않고 장바구니에 담아둔 상품의 목록을 표현한다. 주문내역에 의해 결제가 이루어진 제품은 그 수량을 데이터베이스를 통해 조회하여 재고량을 조정하여 표시한다. 뿐만 아니라 게시판은 전자상거래 시스템을 운영함에 있어서 필요한 각종 게시물과 고객의 문의 사항들을 기록할 수 있는 장치로 이용할 수 있다.

4. 리눅스용 전자상거래 시스템의 구현

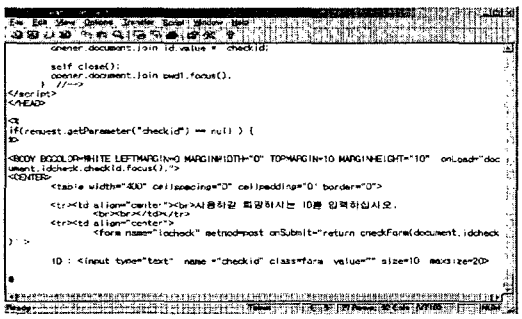
본 연구에 의한 시스템의 설계는 JSP Model 2 Architecture에 의한 방식을 기본 배경으로 한다. 이를 위해 플랫폼은 래드햇 계열의 리눅스 배포판을 바탕으로, 아파치 웹서버, 자카르타-톰캣 서블릿 및 JSP지원 엔진, Mysql의 DBMS, Java 1.3.1, MySql-JDBC 엔진을 사용하여 시스템의 운영환경을 구축하였다. (그림 4) JSP Model 2의 구조는 각종 이벤트 및 시스템의 정보와 요청 등의 문제를 관리하는 컨트롤러 부분과 정보를 저장하고 있는 모델 부분 그리고 정보의 출력을 담당하는 뷰 부분으로 나누어 개발한다. 이러한 구조를 개발하기 위해서는 자바의 JSP 및 서블릿 기술이 효과적이다[2]. (그림 4)전자상거래 시스템의 구성 도메인은 이러한 시스템의 운영환경을 도메인으로 나타내어 일반적으로 이해를 쉽게 하여 준다. 그림에 나타난 바와 같이

소비자와 판매자 또는 전자상거래 시스템 운영자 그리고 전자상거래 시스템 등이 전자상거래를 구성하는 을 일반적인 구성 도메인이 된다.



(그림 4) 전자상거래 시스템의 구성 도메인

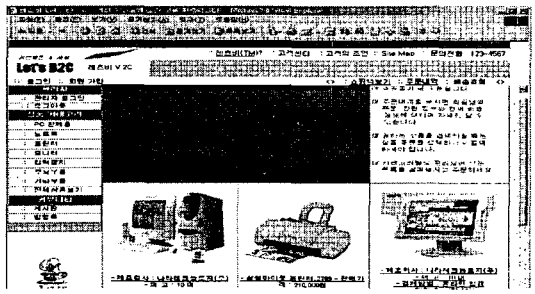
JSP Model 2의 구조에 의한 전자상거래 시스템의 구현은 회원관리 부분과 상품등록 및 관리, 판매관리, 각종 게시판 등으로 나타난다. (그림 5)는 본 연구에 의한 구현 과정의 일부인 회원인증 부분에 관한 아이디 및 비밀번호 등록을 수행하는 JSP프로그램 소스 코드를 나타낸 것이다.



(그림 5) 회원관리 소스 코드(뷰 부분)

(그림 5)의 부분은 MVC Architecture중에서 View부분에 해당한다. 이 부분은 제2장 관련연구에서도 알아보았듯이 시스템의 프리젠테이션의 부문을 관장하는 부문 중 일부이다. 다음 그림인 (그림 6)은 Model 부분 및 Controller부문에 해당하는 그림이다. 본 연구에서는 Model부분 및 Controller부분중 일정한 즉, 상호연관성이 높은 부분을 통합하여 구현하였다. (그림 6은 시스템의 핵심 부분이 된다. 각각의 기능을 정의하고 기술하며 수행하기 때문인데 이 부분을 새롭게 업그레이드 및 유지 보수하면 시스템 전 부분을 재구성하지 않고도 항상 최신의 기능을 유지 할 수 있다.

(그림 6) 모델 부분의 소스 코드



(그림 7) 구현된 전자상거래 시스템의 예

5. 결론 및 향후 계획

본 연구에서는 JSP Model 2구조를 통한 전자상거래 시스템의 구현에 관하여 다루었다. 많은 시스템이 비즈니스 로직과 프리젠테이션 부분을 통합하여 개발하는 데 본 연구와 같은 구현방법은 생산성 및 유지보수성을 강화시켜준다. 제2장에서 관련연구를 제3장에서는 설계를 그리고 제4장에서는 구현과정을 서술하였다. 향후계획으로는 본 방법을 응용하여 보다 나은 시스템의 개발방법 및 기술을 개발 연구하는 것이다.

[참고 문헌]

[1] 통계청, "전자상거래 통계조사 결과", 2001. 8

[2] Understanding JavaServer Pages Model 2 architecture : Exploring the MVC design pattern, http://www.javaworld.com/

[3] 서순모, 양해술,"동적 컴퓨팅 환경에서의 전자상거래 컴포넌트 도입방안",한국정보처리학회 소프트웨어공학연구회지 제3권 제4호, 2000.12