

Templet을 활용한 웹사이트 구축 시스템

윤재광*, 탁진현*, 백영태**, 이세훈***

Web Site Building System Using Templet

JJay Yun*, JinH

yun Tak*, YongTae Baek**, SeHoon Lee***

요약

최근 홈페이지 기반의 다양한 기능이 개발, 적용되고 있어 웹사이트 재구축시 기능 개선 등의 이유보다는 디자인 재구성을 통한 인터페이스 개선을 꼽는 경우가 대다수로 나타나고 있다. 본 논문은 디자인 변경을 위해 기능 전체를 재개발하는 것이 아니라 개발된 템플릿을 활용하여 인터페이스를 변경하는 방법을 제안하고자 한다. 이 방법은 웹사이트 재구축시 개발 기간의 단축, 비용의 절감, 필요 인력의 최소화 등의 이점을 얻을 수 있다.

▶ Keyword : 템플릿(Templet), 웹사이트(Web Site), 홈페이지빌더(Homepage Builder)

1. 서론

온라인 조사기관인 컴스코어(ComScore)가 발표한 보고서[1]에 따르면 당시 전 세계에서 인터넷을 사용한 15세 이상 인구 총계가 7억 4천만여 명이라 발표하고 있다. 또한 이 보고서에 따르면 한국의 인터넷 사용인구는 지난해에 비해 8%증가한 2,630만여 명으로 발표하고 있다. 또한 한국인들의 한달 평균 인터넷 사용시간은 '34시간'이라고 한다. 올해 초 CTR[2]에서 발표한 '2006 중국 도시 인터넷 네티즌 형태 보고'에 따르면 올해 하루 중 가장 이용시간이 긴 매체가

될 것으로 예상된다고 보고했다. 이처럼 현대 정보화 사회에서 홈페이지는 하나의 새로운 매체로 자리잡아 홍보, 온라인 상거래, 정보제공, 온라인 게임 등 다양한 형태의 서비스를 제공하고 있다.

이러한 인터넷 서비스의 경우 초기 제작시 수백만원에서 수억원까지 소요되며 기간 또한 수일에서 수개월까지 소요되기도 한다. 하지만 인터넷 서비스의 대부분은 단순 DB프로그램으로 이루어져 있으며 재구축시 요구되는 항목의 경우 UI 개편, 메뉴 재구성 등으로 초기 제작시에 비해 비교적 단순한 요구사항으로 이루어져 있어 초기 제작에 비해 비용과 기간이 절감되어야 한다.

• 제1저자 : 윤재광

* (주)에듀윌스*, 김포대학**, 인하공업전문대학***

하지만 실제로는 초기제작과 동일한 프로세스에 추가로 기존 웹사이트 분석이라는 프로세스가 추가 되어 오히려 비용과 기간이 초기제작에 비해 증가추세를 보이고 있다 [3,4,5].

본 논문에서는 사용자인터페이스 개편과 메뉴 재구조화의 요구사항의 웹사이트 재구축시의 업무프로세스를 줄일 수 있는 초기 제작 방법 및 웹사이트 구성방안을 제시한다. 기존에 성능과 보안을 위해 구분하였던 3 계층의 구성방안을 활용하여 다수의 사용자 인터페이스와 모듈화의 단일 비즈니스 로직 구성을 통해 재구축시 어플리케이션은 두고 사용자 인터페이스 및 메뉴를 재구성할 수 있는 방안과 기존 비즈니스 로직의 재사용성 및 신규 기능의 개발 용이성 증대를 통해 재구축시 소요되는 비용과 기간을 절감 할 수 있도록 한다.

2. 관련연구

인터넷 서비스의 기본 구성인 웹사이트의 기능들을 살펴보면 그림 1과 같이 UI에는 일부 차이가 있으나 기능은 대동소이한 단순 DB프로그램으로 구성되어 있다 따라서 웹사이트 재구축시 신규 기능의 개발외의 상당부분은 UI 변경을 통한 기존 기능의 재사용으로도 사용자의 요구사항에 부합할 수 있다. 하지만 현재 통용되고 있는 그림 2의 웹사이트 개발 프로세스는 초기 신규 개발과 재구축이 구분되어 있지 않아 해당 프로세스에 따라 웹사이트 재구축 과정을 거치게 되면 기존 기능의 재사용은 불가능하다.

현재도 기존 기능의 재사용을 위해 산출물 등을 통한 기능 명세 등이 이루어지고는 있으나 이는 웹사이트 재구축시 재사용 목적이 아니라 유지 보수 시 사용을 목적으로 진행되고 있어 UI 개편과 메뉴 재구조화가 이루어 져야 하는 재구축에서 활용하기는 어려운 것이 현실이다. 웹사이트의 재구축시에도 활용가능하기 위해서는 일부 기능에 대한 재사용 고려가 아니라 전체 UI와 각 기능의 모듈화가 이루어져 제작되어야 한다.

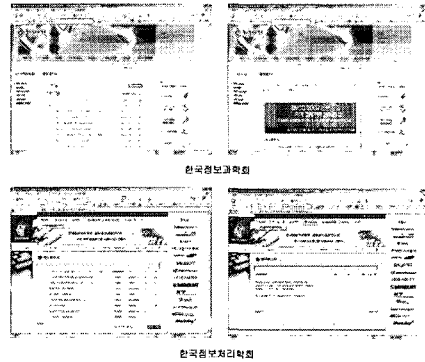


그림 1 정보과학회와 정보처리학회의 UI 비교

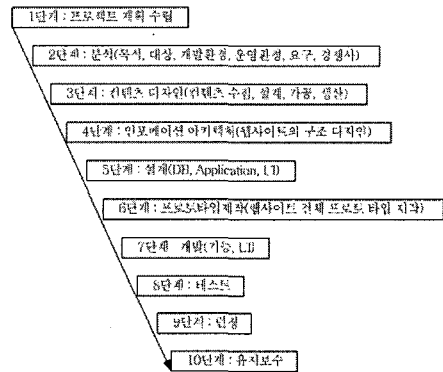


그림 2 웹사이트 개발 프로세스

표 1 웹사이트 개발 방식

구분	개발 방식	재사용 가능
전체 UI	HTML, CSS	없음
개발 가능 UI	HTML, CSS	없음
메뉴 구성	Script, App.	없음
기능	Script, App.	일부 고려

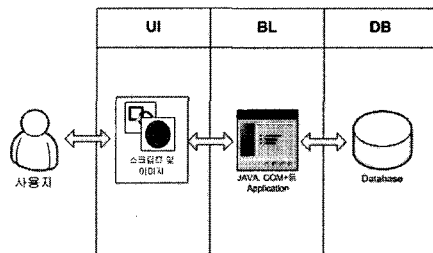


그림 3 기존 웹사이트 구조

현재 사용되는 일반적인 개발방식에서는 표 1 에서와 같이 UI의 측면과 메뉴 구성에서는 재사용에 대한 고려 없이 제작 되고 있으며 그림 3에서와 같이 기능 일부에서만 재사용이 가능하도록 모듈화하여 개발하고 있다. 하지만 재사용 가능성도 유지보수에 국한되면 웹사이트 전체 재구축시에는 프로그램 구성의 제한으로 신규개발이 통용 되고 있다.

3. 템플릿 형태의 웹사이트 구성

이 장에서는 템플릿 형태의 웹사이트 구성방안을 제시 하여 웹사이트의 재구축시 기능의 재사용성 강화와 자유로운 UI변경이 가능하도록 하고자 한다. 그림 4와 같이 초기 설계 및 개발시 템플릿 형태로 웹사이트를 개발함으로써 UI및 기능의 재사용이 용이하도록 할 수 있다.

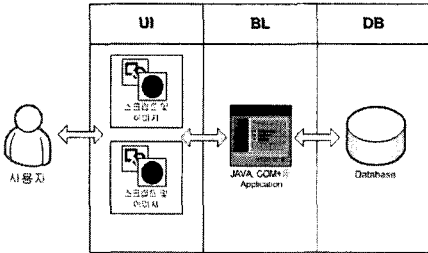


그림 4 템플릿 형태의 웹사이트 구조

또한 템플릿의 정보를 데이터베이스화 하여 템플릿 변경값을 저장할 수 있도록 구성하면 템플릿변경을 위해서 웹사이트의 소스를 수정하지 않아도 되도록 구성할 수도 있다.

이렇게 데이터베이스로 구성하여 관리하는 템플릿 구조의 웹사이트 구축 시스템의 전체적인 프로세스를 살펴보면 그림 5와 같다.

- 템플릿을 폴더별로 분리 구성
- 템플릿의 폴더명등의 값을 데이터베이스에 저장
- 템플릿정보의 사용여부(templet_usetag)의 값을 '1'로 지정
- Templet_info 테이블의 templet_usetag의 값이 '1'인 폴더 노출

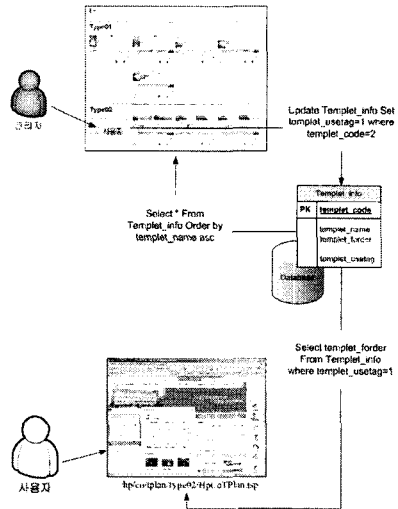


그림 5 템플릿 구조 업무프로세스

템플릿구조의 폴더구성을 살펴보면 보안을 위해 Script 폴더와 HTML, 이미지 폴더를 구분하여 구성하였다. 또한 폴더의 구조는 템플릿별로 구성하여 하나의 템플릿은 같은 폴더명을 가진 Script 폴더와 HTML, 이미지 포더의 SET으로 되어 있다.

그림 6 Script 템플릿폴더

그림 7 Script 문서

그림 6, 7은 웹사이트에 사용되는 Script로 구성된 Script 템플릿 폴더를 보여주고 있다.

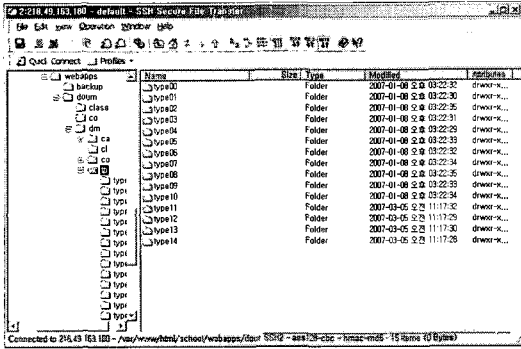


그림 8 이미지 템플릿 폴더

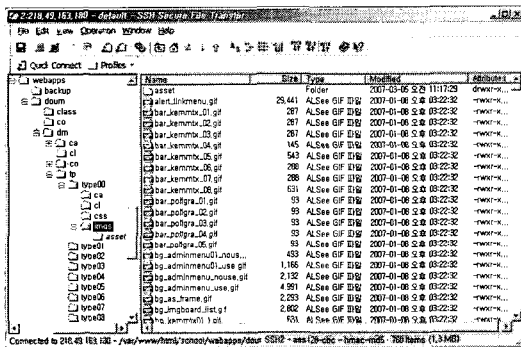


그림 9 이미지 파일

그림 8, 9는 사용자 UI를 구성하는 이미지 템플릿 폴더를 나타내고 있다. 웹사이트의 구성을 그림 6, 7, 8, 9와 같이 구성하고 템플릿에 따라 다른 Script와 이미지를 보여줄 수 있도록 기본 페이지의 소스를 구성함으로써 사용자에게는 다른 형태의 UI를 제공할 수 있다.

프로그램 1 기본페이지 소스

```
<%@ page contentType="text/html; charset =KSC5601" %>
<%@ include file="/include/initial_include.jsp" %>
<html>
<head>
<title>{<eg_ini.html_title%></title>
<meta http-equiv="content type" content="text/html; charset=euc-kr">
<link rel="stylesheet" href="/css/style.css" type="text/css">
</head>
<frameset rows="100%.*" frameborder=0 border=0 marginwidth=0
marginheight=0 topmargin=0 leftmargin=0 framespacing=0>
<frame
```

```
src="/hp/co/tplan/{<eg_ini.tp_folder_name%>}/HpCoTPlan.jsp"
NAME="contents" scrolling="yes" noresize)
<frame name =noshow src="{<eg_ini.vDir%>}/blank.jsp scrolling=no">
</frameset>
</html>
```

또한 그림 10와 같이 기본페이지의 템플릿 지정에 해당하는 부분을 운영자가 직접 변경할 수 있도록 구성하면 전체 UI를 운영자가 직접 선택하여 사용할 수 있도록 구성할 수 있다.

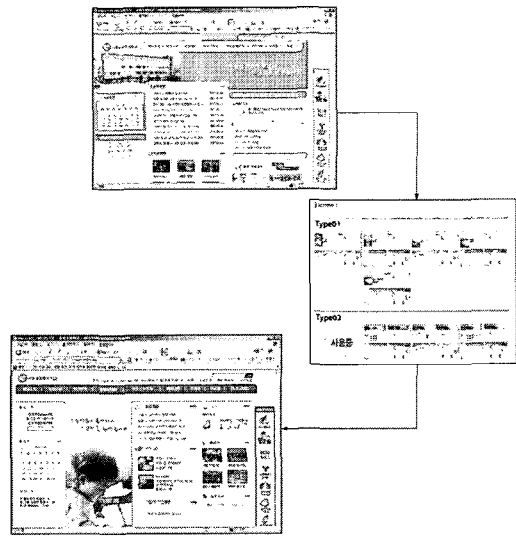


그림 10 템플릿 변경 운영자 UI

웹사이트를 이와 같이 템플릿구조로 개발한 결과 초기 개발시 약 30M/M 이 소요되었으나 이후 신규 템플릿 추가시 1 M/M 정도의 투입으로 개발가능하였다.

4. 기능의 모듈화

이 장에서는 기능의 모듈화 방안을 제시하고자 한다. 웹사이트를 구성하는 기능을 모듈화하고 앞에서 설명한 템플릿형태와 연계하여 개발하면 메뉴의 재구성, 기능 변경, 신규기능 개발시 용이하게 적용할 수 있다.

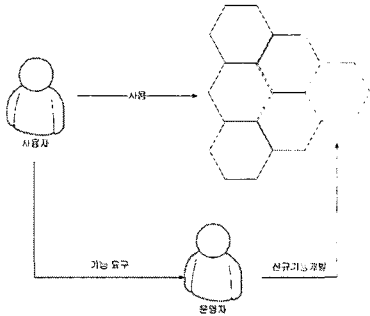


그림 11 기능의 모듈화

또한 기능에 대한 명세를 데이터베이스화하여 관리하고 메뉴역시 데이터베이스화 관리하게 되면 기능을 신규 개발하거나 변경을 용이하게 할수 있었다. 또한 메뉴의 재구조화를 데이터의 변경만으로 가능하였다.

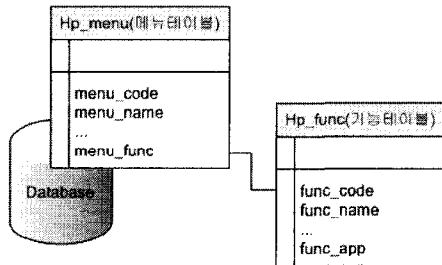


그림 12 DB 구조도

프로그램 2는 기능 선택 Agent 페이지의 소스로 기능의 파라메타에 따라 보여지는 페이지가 다르게 나타나도록 구성한 것이다. 즉, 메뉴 테이블의 기능 값(menu_func)에 따라 기능의 소스 위치(func_app)가 달라지도록 구성하였다.

프로그램 2. 기능 선택 소스

```

<!-- 서버 컨텐츠 영역 시작 -->
<jsp:include page="<%="/" + vr.cDir + "/" + vr.cf + ".jsp" +
(menu_code.equals("") ? "" : "?menu_code=" + menu_code)%>"
flush="true"/>
<!-- 서버 컨텐츠 영역 끝 -->
<!-- 서버 컨텐츠 영역 시작 -->
<jsp:include page="<%="/" + vr.cDir + "/" + vr.cf + ".jsp" +
(menu_code.equals("") ? "" : "?menu_code=" + menu_code)%>"
flush="true"/>
<!-- 서버 컨텐츠 영역 끝 -->
    
```

이렇게 기능 개발시 모듈화하여 구성하고 이를 데이터베이스화 관리하며 메뉴 역시 데이터베이스화 하여 관리하도록 하고 관리자에게 이를 변경할수 있는 인터페이스를 그림 11과 같이 구성하여 메뉴 재구조화에 걸리는 시간을 1 Day 이내로 줄일수 있었다.

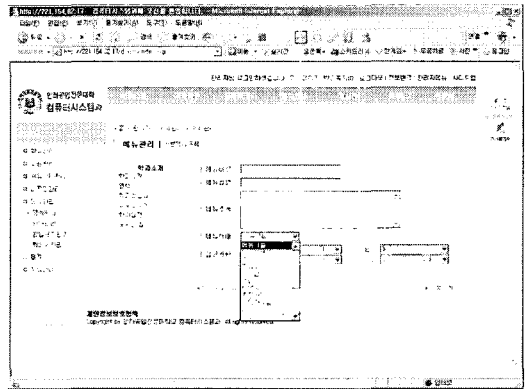


그림 11 기능 모듈화

6. 결론

이 연구에서는 템플릿을 기반으로 하여 UI에 대한 변경 유연성을 강화하고 각 기능 개발시 모듈화를 구현하여 사용자의 요구사항에 대한 대응을 신속히 진행 할 수 있도록 하고 또한 기능의 모듈화를 통해 메뉴의 재구성이 용이하도록 하여 주기적으로 나타나는 웹사이트 재개발에 대한 요구를 최소화 할 수 있는 방안을 제시하였다.

참고문헌

- [1] Worldwide Internet Audience has Grown 10 Percent in Last Year, According to comScore Networks, ComScore, 2007, <http://www.comscore.com>
- [2] 2006 중국 도시 인터넷 네티즌 형태 보고, 중국중앙텔레비전연구공사(CTR), 2007
- [3] 웹 데이터베이스를 포함하는 멀티미디어 홈페이지 제작 시스템, 성승규, 2001
- [4] Bob Boiko, Content Management Bible, Hungry Minds, Inc., 2002
- [5] JoAnn T. Hackos, Content Management for Dynamic Web Delivery, John Wiley&Sons, Inc., 2001