
COM 포트를 이용한 웹기반 POD 시스템

함종완* · 김학진* · 정희경**

Web-based POD system Using the COM Port

Jong-Wan Ham* · Hak-Jin Kim* · Hoe-Kyung Jung**

요 약

본 논문에서는 접근성이 뛰어난 웹 서비스라는 환경을 이용하여, 템플릿을 생성·제공하고, 사용자가 쉽게 편집할 수 있도록 제공하며, 편집한 파일과 동질의 인쇄물을 실시간으로 출력할 수 있는 웹 표준을 따르는 COM 포트 (Communication Port)를 이용한 POD(Print on Demand) 시스템의 프레임워크(framework)를 제안한다. 본 논문에서 제안한 프레임워크의 특징은 기본적인 인터넷 활용 능력을 갖춘 사용자들로 하여금, 기존의 전문 편집 프로그램과 비교하여 상대적으로 쉽게 배우고 활용 할 수 있으며, 사용자가 직접 제작한 파일을 화면 그대로의 품질로 실시간으로 인쇄할 수 있는 프로그램을 만들고, 접근성을 높이도록 하였다.

ABSTRACT

This article proposes a POD system framework which creates templates by using the Web service environment with good accessibility, provides them so that users can easily edit them, and uses COM ports following Web standard capable of real-time printing with the same quality as edited files. The characteristic of framework proposed in this article is that it creates programs which users with basic skill in Internet use can learn and use relatively easily when compared with existing specialized editing programs and which allows users to print real-time such files as personally created by them in the same quality as on the screen, and thereby enhances accessibility thereof.

키워드

POD, COM Port, PDF, 전자출판

Key word

POD, COM Port, PDF, Electronic publication

* 정희원 : 배재대학교

** 종신희원 : 배재대학교 (교신저자, hjkung@pcu.ac.kr)

접수일자 : 2010. 12. 31

심사완료일자 : 2011. 04. 25

I. 서 론

인터넷의 보급에 따라 인쇄 분야에도 적용되어, 최근에는 인쇄물을 사용자가 직접 제작하여 실시간으로 인쇄할 수 있는 POD(Print on Demand) 시스템을 갖춘 웹사이트들이 생겨나고 있다.

이에 인쇄 기획자들로 하여금, 포토샵이나 일러스트레이터와 비교하여 상대적으로 쉽고, 접근이 쉬운 인쇄 분야에 활용할 수 있는 명함 제작, 스티커 제작과 같은 활용 가능한 프로그램을 만들고, 접근성을 높이기 위한 프로그램의 보급이 시급하다.

본 논문에서는 웹상에서 작업된 인쇄 작업물에 대한 인쇄 요구가 있을 때, 작업물을 웹상에서 보여지는 모습을 고품질로 COM 포트를 이용한 스크린샷 기술을 적용하여 이미지를 생성하고 썸네일 형태로 관리하며, 실시간으로 인쇄소에서 사용할 수 있는 형태의 POD[1] 시스템으로, 접근성과 사용의 편리성, 이미지의 고품질을 목표로 다음과 같은 기능을 수행할 수 있는 프레임워크를 제안한다.

연구의 주요 내용을 살펴보면 POD 시스템부분에서는 관리자가 다양한 템플릿을 작업하여 등록하고 수정 및 삭제할 수 있도록 하였으며, 모든 웹서비스를 제공하기 위해 필수적인 게시판에 위시웍(WYSIWYG : What You See Is What You Get) 기능을 첨부하여 편리한 인터페이스 기능을 제공할 수 있도록 하였다. 또한, HTML을 PDF로 변환하기 위해 현재 화면의 HTML 정보를 COM 포트를 이용하여 이미지로 변환한 후, 변환된 이미지를 PDF로 변환하는 방법을 사용하여 단순하게 이미지를 PDF화함으로써, 항상 동일한 작업을 수행할 수 있고, 고해상도로 작업된 데이터에 대해서 축소 형태를 가질 수 있도록 효율적으로 제공한다. 또한, 기존에 문제가 되는 PDF파일에서 다양한 폰트지원의 문제와, 인쇄용 폰트와 웹 폰트의 차이에서 오는 미세한 차이를 이미지화함으로써, 화면에서 보이는 형태의 폰트를 그대로 활용가능하다는 장점을 가지며, 무료로 배포하고 있는 다양한 웹 폰트를 라이선스의 제약 없이 활용할 수 있도록 하였다.

II. 관련 연구

본 장에서는 COM포트를 이용한 웹기반 POD 시스템에서 필요로 하는 기술 중에서 화면의 HTML 정보를 이미지로 변환한 후, 변환된 이미지를 PDF로 변환하기 위해 COM 포트를 이용한 스크린샷 기능에 대해 알아본다. POD 시스템을 구현하기 위해서는 크게 두 가지의 접근 방법을 생각해 볼 수 있다.

첫째는 웹에서 작업된 HTML를 PDF파일 형태로 변환하는 방법이다. 이론적으로 가장 효율적인 방법이지만, 단순한 HTML을 PDF로 변환하는 데는 크게 문제가 되지 않지만, 각종 스타일시트가 적용된 HTML이나, 여러 개의 레이어로 구성된 HTML을 PDF로 변환하는 데는 한계를 가질 수 있다[2]. GUI 형태의 편집 작업이 주 작업이 되는 POD의 경우에는 현실적으로 한계를 가질 수밖에 없는 상황이다.

두 번째는 HTML을 PDF하기 위해 현재 화면의 HTML 정보를 이미지로 변환한 후, 변환된 이미지를 PDF로 변환하는 방법이다. 이 방법은 단순하게 이미지를 PDF화함으로써, 항상 동일한 작업을 수행할 수 있고, 고해상도로 작업된 데이터에 대해서 축소 형태를 가지면 효율적이다. 하지만 이 방법의 한계는 작업 모들이 큰 인쇄물 작업에는 효율적이지 못하다는 점이다. 그럼에도 불구하고, 이 방법은 기존에 문제가 되는 PDF파일에서 다양한 폰트지원의 문제와, 인쇄용 폰트와 웹 폰트의 차이에서 오는 미세한 차이를 이미지화함으로써, 화면에 보이는 형태의 폰트를 그대로 활용가능하다는 장점을 갖는다. 이는 현재 네이버, 다음과 같은 포털 사이트에서 무료로 배포하고 있는 다양한 웹 폰트를 라이선스의 제약 없이 활용할 수 있다는 장점을 갖는다. 이러한 이유로 POD 시스템에서는 후자의 방법을 이용하는 것이, 전자의 방법보다 효율적일 것이라 사료된다.

III. POD 시스템 설계

본 장에서는 제안된 COM 포트를 이용한 웹 POD 시스템의 전체적인 흐름도와 본 시스템을 관리자, 시스템, 인쇄 기획자/사용자, 인쇄소 모듈로 나누어 각각의 모듈

별 기능을 기술한다.

3.1 POD 시스템의 구성

웹 기반의 POD 시스템은 그림 1과 같이 관리자, 시스템, 인쇄소, 인쇄 기획자/사용자, CD 관리, 외화 관리의 모듈로 구분하여 설계하였다.

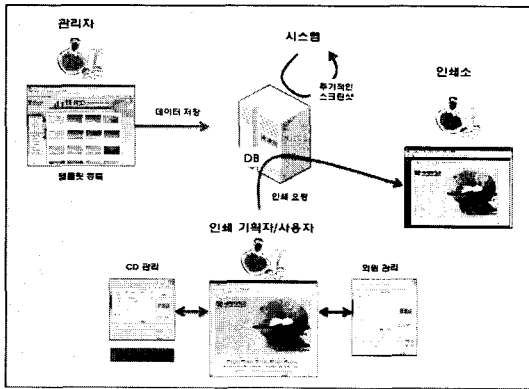


그림 1. 시스템 구성도
Fig. 1 System Block Diagram

3.1.1 관리자 모듈

이는 스타일 시트나 게시판, 카테고리 등을 등록, 수정, 삭제와 같은 관리자 영역과, 생성된 카테고리를 중심으로 템플릿 데이터를 생성, 유지 보수하는 영역으로 구성된다. 따라서, 사용자들과 소통할 수 있는 스타일 시트나 게시판을 구성하는 모듈과 템플릿을 생성하기 위한 위지윅 게시판 모듈과, 카테고리를 생성하는 `Text2Image` 컨트롤 모듈로 구성하였다.

3.1.2 시스템 모듈

시스템의 핵심 기능으로, 관리자가 등록한 데이터에 대해 내부적으로, COM 포트를 이용하여 화면을 스크린샷하고, 데이터베이스를 관리해 주도록 설계하였다. 이를 위해 시스템은 COM 포트를 이용하여 하드웨어의 도움을 받아 수행된다.

PHP 스크린샷 모듈은 웹 서비스를 이용하여 특정 사이트에 대한 스크린샷이 매우 유용하게 사용할 수 있다. 이 방법은 썸네일[3-5] 형태로 사이트 링크에 대한 미리 보기가 가능하여 보다 효과적인 웹 서비스 제공이 가능하다. 이런 스크린샷 모듈을 이용하여, POD에 활용하기

위해서는 최대한 큰 사이즈의 화면 스크린샷 정보가 필요하다. 이를 위해 풀 스크린으로 스크린샷 하는 것이 필요하다. 이는 일반적으로 "COM" 컴포넌트를 이용하여, 새로운 창을 띄워서 `imageGrpScreen()`을 이용하는 경우, 화면 전체가 스크린샷 되어, 해당하는 명함 부분을 자를 수 없는 상황보다 훨씬 효율적인 방법이라고 판단된다. 다음은 화면 캡처를 위한 프로그램 알고리즘을 보인다.

```

if($idx and !$quit)
{
    $src = "/xyxyxy.jpg";

    $browser = new COM("InternetExplorer.Application");
    $handle = $browser->HWND;

    $browser->Visible = true;
    $browser->Fullscreen = true;

    $browser->Navigate("http://localhost/test/man_namecard.php?i
dx=$idx&quit=1");

    while ($browser->Busy) {
        com_message_pump(2000);
    }
    $im = imagegrabwindow($handle, 0);
    $browser->Quit();
    imagejpeg($im, $src);
    imagedestroy($im);
}

// 테스트를 위해 사이즈를 하드코딩으로 처리.
$size = getimagesize($src);
$w = $size[0];
$h = $size[1];
$nw = 130;
$nh = 70;

$fromx = 45;
$fromy = 190;

$tox = 600;
$toy = 280;
$ddest = "/icon/template/namecard/$idx.jpg";

$simg = imagecreatefromjpeg($src);
$dimg = imagecreatetruecolor($nw, $nh);
imagecopyresampled($dimg, $simg, 0, 0, $fromx, $fromy,
$nw, $nh, $tox, $toy);
imagejpeg($dimg, $ddest, 72);
//imagedestroy($ddest);
imagedestroy($simg);
imagedestroy($dimg);
    
```

3.1.3 인쇄 기획자/사용자 모듈

이는 관리자가 등록한 다양한 형태의 템플릿 목록을 이용하여, 쇼핑몰에서 나열된 제품을 고르는 것과 같은 형태로, 가장 유사한 형태의 정보를 선택하며, 기본적인 정보를 입력하여 등록하며, 인쇄 요청 서비스를 가능하도록 하였다. 인쇄 기획자나, 관리자 모두 편집이 가능해야 하기 때문에, 가능한 한 단순화된 인터페이스를 사용하는 것을 원칙으로 설계하였다. 일반 사용자는 스크린 샷된 이미지 정보와, 편집 수정할 수 있는 기능을 가지고 있고, 인쇄 요청이 되는 경우에 진행 절차에서 PDF 파일을 다운로드 받을 수 있도록 하였다.

3.1.4 인쇄소 모듈

인쇄소 역시 동일한 형태로 회원 가입 후, 관리자가 승인한 경우에, 사용자가 인쇄 요청한 데이터에 대한 PDF 파일을 다운받아, 이를 출력하고, 진행 상황을 역방향으로 전달할 수 있는 형태로 회원 서비스 형태로 구성하였다.

사용자와 마찬가지로, 인쇄 절차에 들어간 데이터를 클릭함으로써 PDF 파일을 다운로드 받을 수 있다. 따라서, PDF파일로 생성되기전, 한글 지원 모듈과, HTML 지원 모듈, Image2PDF 변환 모듈과 인쇄용 데이터 처리를 위한 RGB2CMYK 변환 모듈을 거쳐야 한다 [6-8]. 또한, 인쇄소는 일반 사용자들이 인쇄 주문 요청을 할 때 해당 인쇄소를 선택할 수 있어야 하기 때문에, 관리자의 승인이 있는 경우에만 목록에 표시할 수 있도록 설계하였다.

3.2 POD 시스템의 프로그램 구조

본 논문에서 제안한 웹기반 POD 시스템은 요청이 있을 때 출력할 수 있는 실시간 서비스를 제공하기 위한 목적을 가지므로 PHP기반의 웹 서비스를 통해 이루어졌다. 이를 위해 기본적인 웹의 설계가 필수적으로 이를 위한 기본 골격은 다음과 같다.

3.2.1 웹 페이지 골격

모든 웹 페이지는 팝업창을 제외하고, main화면(main.php)을 통해 이루어진다. 화면의 전체적인 구성은 그림 2와 같다. 이 페이지를 통해 팝업이나, 블랙리스트 검사, IP검사 등을 효율적으로 수행할 수 있다. 변수 \$main_h1은 셀의 픽셀(pixel)을 의미하며 해당하는 값이

“0”인 경우에는 해당하는 곳을 표시하지 않는다. 많은 페이지가 웹페이지에 존재하는데, 각각의 구분은 \$request값을 통해 구분할 수 있고, \$request가 없는 경우에는 “홈”을 눌렀을 때를 의미한다. 이때 \$main_total_w = \$main_w1 + \$main_w2 + \$main_w3로 이 값은 페이지가 오른쪽으로 늘어나는 것을 방지하는데 사용되어야 한다. 내용이 포함될 ⑦에 긴 텍스트가 존재하는 경우 전체적으로 화면 늘어나는 상황이 발생할 수 있다.

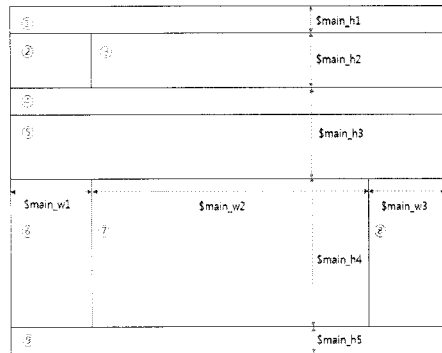


그림 2. 웹 페이지의 구조
Fig. 2 Structure of Web Page

3.2.2 main.php 파일의 기본 구성

main.php 파일의 기본 구조는 다음과 같다.

```

01: include ("./inc/con그림inc.php");
02: include ("./inc/db.inc.php");
03: include ("./inc/function.inc.php");
04:
05: // session processing
06: include ("./head.php");
07:
08: // table start
09: // print ① : include ("./top_menu.php");
10: // print ②
11: // print ③ : include ("./up_menu.php");
12: // print ④ : // RESERVED
13: // print ⑤ : include ("./flash.php");
14: // print ⑥ : include ("./menu.php");
15: // print ⑦ : include ("./display.php");
16: // print ⑧ : include ("./banner.php");
17: // print ⑨ : include ("./siteinfo.php");
18:
19: include ("./tail.php");

```

3.2.3 con그림inc.php 파일의 기본 구성

con그림inc.php 파일은 모든 페이지에서 반드시 필요한 정보를 설정하기 위한 파일로, 세션 정보가 대표적이다. 이때 세션 값은 고정된 값을 사용하게 되는 경우 프로그램의 유연성이 떨어지기 때문에, 세션값을 변수화하여 사용하였다.

이 경우 \$_SESSION['id']와 같은 형태로 사용하는 것이 아니라 \$_SESSION[\$session_id]처럼 사용하고, 이 파일에서 \$session = "pod_session_id"와 같이 함으로써 설정파일만 수정하면, 모든 코드에 반영된다.

IV. POD 시스템 구현

본 장에서는 앞에서 설계한 모듈들을 사용하여 POD 시뮬레이션 시스템을 관리자측면과 사용자측면으로 나누어 구현한 모습을 보여준다.

4.1 시뮬레이션 환경

시뮬레이션 환경으로 플랫폼은 Windows XP에서 개발 언어로는 PHP 4.x, 데이터베이스는 MySql 5.0.77 community-nt, 웹서버는 APM_APACHE2, 언어셋은 euc-kr에서 하였다.

4.2 관리자 화면

화면 스킨은 전체적인 화면 구성과 각종 버튼의 색상 등에 대해 관리자 메뉴를 통해 자유자재로 관리하기 위한 용도의 메뉴이다. 이 메뉴는 하나의 테이블만 있으면 되기 때문에 등록 기능은 없고, 수정기능만 제공한다. 관리자로그인 후 "스킨관리"를 클릭하면 그림 3과 같은 화면을 볼 수 있다.

4.3 인쇄 기획자/사용자 화면

화면 스킨은 관리자 화면에서 설명한 내용과 유사한 절차를 따른다. 원하는 카테고리에서, 해당하는 로고나, 이름, 주소 정보 등을 찾아 원하는 데이터를 변경한다. 본 시스템은 아주 정밀한 작업보다 다양한 템플릿이 제공되는 시스템에서, 작업의 최소화를 목적으로 한다.

4.3.1 템플릿의 결정

사용자는 관리자가 등록된 데이터 중 원하는 형태와

가장 흡사한 디자인 템플릿을 선택한다. 선택한 데이터는 관리자 화면의 편집화면과 완전히 동일한 화면과 데이터이다. 이는 그림 4와 같다.

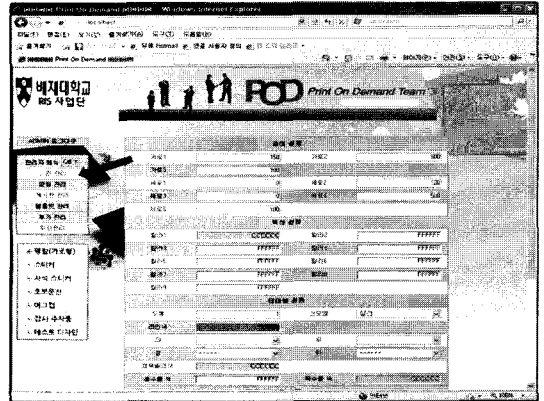


그림 3. 스킨 적용전 화면
Fig. 3 Screen of Skin Before Application

4.3.2 편집 또는 새로운 작성

선택한 템플릿을 이용하여, 편집하는 화면 및 이를 등록한 후의 화면은 그림 5에 나타나 있다. 화면의 하단에서 보는 바와 같이, 서버가 주기적으로 스크린샷을 하기 때문에 "이미지 준비중"과 같은 임시 이미지로 화면에 나타난다.

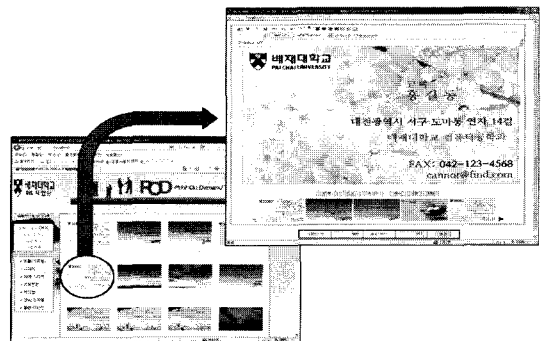


그림 4. 템플릿의 결정
Fig. 4 Decision of Template

이는 서버가 10초 간격으로 스크린샷을 수행하기 때문에 평균적으로 5초, 최대 10초의 시간 후면 캡처가 종료된다.



그림 5. 웹 페이지의 화면 구조
Fig. 5 Structure of Web Page

4.3.3 작업 목록 확인

사용자가 새롭게 등록해 놓은 작업 데이터는 편집 화면에서 관리자가 등록한 데이터와 동일한 화면에서 표시되지만, 실제 사용자의 등록 데이터는 인쇄 요청을 하기 위한 데이터이다. 그렇기 때문에 사용자가 등록한 데이터는 사용자 메뉴의 “작업 목록”에서 확인 할 수 있다.

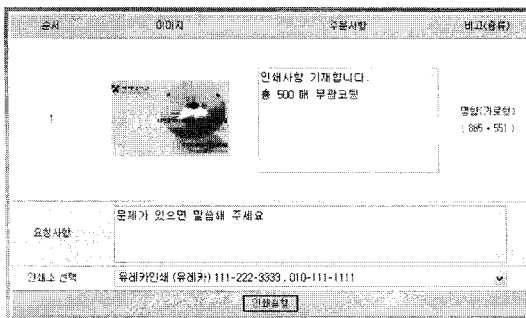


그림 6. 주문 실행 메시지 등록
Fig. 6 Message Registration of Order Execution

4.4 인쇄소 화면

그림 6의 선택메뉴를 클릭하면, 인쇄 요청을 실행할 수 있다. 인쇄 요청을 클릭하면 선택된 목록에 대해 각각 주문 사항을 기재할 수 있고, 이미지를 포함한 간단한 작업 규격 등이 표기되고, 주문에 대한 각각의 상세 주문 사항을 기재할 수 있다. 화면 하단의 요청 사항은, 복수

개의 데이터 처리를 위해서 추가적으로 작성해야 하는 경우를 위한 공간으로 할당하였다. 뿐만 아니라, 인쇄소를 선택함으로써, 출력소를 선택할 수 있고, 목록에 표시되는 인쇄소 목록은 관리자에 의해 인쇄 기업으로 등록된 사용자에 대한 사업장명, 이름, 전화번호, 휴대폰번호를 표시하여, 선택 시 혼란을 배제하였다.

V. 결 론

본 논문에서는 웹상에서 작업된 인쇄 작업물에 대한 인쇄 요구가 있을 때, 작업물을 웹상에서 보여지는 모습의 고품질로, COM 포트를 이용하여 스크린샷하여 이미지를 생성하고, 썸네일 형태로 관리되며, 실시간으로 인쇄소에서 사용할 수 있는 형태의 POD 시스템으로, 접근성과 사용의 편리성, 이미지의 고품질을 목표로 연구되었다.

본 논문에서 제안한 시스템은 다양한 템플릿을 제공하여, 최소한의 데이터 작업만으로, 원하는 결과물을 얻는 것이 가능한 형태의 시스템에 대해 웹을 통해 구현함으로써, 타 시스템이나 프로그램에 비해 향후의 확장성이나, 빠른 추가기능의 개발이 가능한 환경을 구축할 수 있을 것으로 사료된다.

또한, HTML을 PDF로 변환하기 위해 서버측에 스크린샷 기능을 부여하여, 현재 화면의 HTML 정보를 이미지로 변환한 후, 변환된 이미지를 PDF로 변환하는 방법을 사용하여 단순하게 이미지를 PDF화함으로써, 항상 동일한 작업을 수행할 수 있고, 고해상도로 작업된 데이터에 대해서 축소 형태를 가질 수 있도록 효율적으로 제공토록 하였다. 기존에 문제가 되는 PDF파일에서 다양한 폰트지원의 문제와, 인쇄용 폰트와 웹 폰트의 차이에서 오는 미세한 차이를 이미지화함으로써, 화면에서 보이는 형태의 폰트를 그대로 활용가능하다는 장점을 가지며, 무료로 배포하고 있는 다양한 웹 폰트를 라이선스의 제약 없이 활용 가능하도록 하였다.

향후 연구로는 보다 소형 인쇄 결과물에 국한하지 않고, HTML2PDF와 같은 기술을 이용하여 A4용지 이상의 작업물에 대한 웹 서비스가 가능한 POD 개발이 필요하다.

참고문헌

- [1] 김정훈, 김정미, 새로운 도서 혁명 : Print-on-Demand, 유비쿼터스 트렌드 제3호, 2006. 5.
- [2] 김도현, “GPL 폰트를 내장한 PDF 전자문서의 저작권,” 동국대학교 비교법문화연구소, 비교법연구 제6권 제1호, 2005.
- [3] 김경원, 임태범, 이석필, “이미지 특성 추출 정보를 이용한 콘텐츠 검색 엔진 설계,” 대한전자공학회, 제33권, 제1호, 2010.
- [4] 홍지영, 채행성, 조운정, 정대현, 김종완, 김성은, 이혜정, 한광희, “Internet Protocol TV 환경에서 효율적인 웹 탐색기법의 사용성에 대한 비교 연구,” Journal of the Ergonomics Society of Korea, 제 27권, 제4호, 2008.
- [5] Heman. A, “Image Browsing on a Large Display,” Information Technology Interfaces. 2007, ITI 2007. 29th International Conference on. pp245-250, 25-28, 2007.
- [6] 정우성, 이은주, “CSS 가독성 향상을 위한 최적화기법,” 한국컴퓨터정보학회, 제15권. 제7호, 2010(7).
- [7] M. Keller and M. Nussabaumer, “Cascading style sheets: a novel approach towards productive styling with today’s standards,” in Proc. of
- [8] 김선희, 이용환, “Adobe Photoshop에서 인쇄목적으로 한 색상모드 변환에 관한 연구- RGB에서 CMYK로 변환된 데이터를 중심으로-,” 한국사진학회지, 제18호.p6-19, 2008.

저자소개



함종완(Jong-Wan Ham)

2009년 배재대학교
컴퓨터공학과(공학사)
2011년 배재대학교
컴퓨터공학과(공학석사)

2011년 2월~현재 배재대학교 컴퓨터공학과(박사과정)
※관심분야: 멀티미디어 문서정보처리, XML, SVG,
Web Services

김학진(Hak-Jin Kim)



1978년 한남대학교 화학과
(이학사)
1980년 숭실대학교
화학과(이학석사)

1988년 한남대학교 화학과(이학박사)
2010년~현재 배재대학교 산학협력단 교수
※관심분야: 기술창업 및 사업화, U-Health, IT
융합기술



정희경(Hoe-Kyung Jung)

1985년 광운대학교
컴퓨터공학과(공학사)
1987년 광운대학교
컴퓨터공학과(공학석사)

1993년 광운대학교 컴퓨터공학과(공학박사)
1994년~현재 배재대학교 컴퓨터공학과 교수
※관심분야: 멀티미디어 문서정보처리, XML, SVG,
Web Services, Semantic Web, MPEG-21, Ubiquitous
Computing, USN