

HTML5를 활용한 웹 기반 파일 전송 시스템

Web based File Transmission System using HTML5

김유두*, 김모한**, 문일영*

Yu-Doo Kim*, Mo-Han Kim**, and Il-Young Moon*

요 약

스마트 폰, 태블릿 PC 등 다양한 스마트 단말의 등장으로 이를 활용한 다양한 서비스를 받고 있다. 특히 언제 어디서나 단말에 상관없이 서비스가 연결되어 사용될 수 있도록 콘텐츠를 공유하는 기술이 많이 활용되고 있다. 하지만 현재의 콘텐츠 공유 시스템은 다양한 단말 지원을 위해 각 단말의 OS에 맞는 별도의 애플리케이션을 제작하여 제공하고 있다. 이러한 방법은 서비스 제공을 위한 개발 비용 증가로 이어진다. 따라서 어느 단말에서나 하나의 애플리케이션으로 서비스가 가능한 웹 기술이 콘텐츠 공유 기술로 활용된다면 개발 비용의 감소로 이어질 수 있다. 이에 본 논문에서는 최신 웹 기술을 활용한 파일 전송 시스템을 통해 그 성능을 분석하고 현재의 애플리케이션 기반 시스템과의 비교를 수행 하였다.

Abstract

After launched various smart devices such as smart phone and tablet PC, people who using smart devices are provided various services. Especially, contents sharing technology is utilized in any devices. But current contents sharing systems are made on various versions for various smart devices. These methods will be raising cost for development of application. Therefore it must using web technology for reducing cost. In this paper, we had analyzed performance of file transmission using web technology and compared with application based system.

Key words : Web, HTML5, Web Socket, File API, File Transmission

I. 서 론

다양한 스마트 기기의 발전으로 1인 1 PC 시대에 이어 1인 N 단말의 시대가 도래 하고 있다. 특히 스마트폰 및 태블릿 PC 등의 휴대용 단말 뿐 아니라, 스마트 TV와 같이 고정되어 사용되는 가전제품 에서도 스마트 기능이 활용되기 시작하였다. 이로 인해 다양한 스마트 단말간의 정보를 공유하여 사용자가

여러 단말에서 동일 서비스를 연속적으로 제공 받고자 하는 수요가 급격히 증가하고 있다. 따라서 구글과 애플 등의 기업에서 다양한 클라우드 서비스가 제공되고 있다[1]. 하지만 이러한 서비스는 동일 플랫폼을 제공하는 단말간의 서비스만 제공되거나 표준으로 정해진 동영상 재생 등의 부분에만 한정되어 제공되고 있다. 이를 보완하기 위하여 다양한 플랫폼을 지원할 수 있는 서비스도 있지만, 이러한 서비스를

* 한국기술교육대학교 컴퓨터공학부(Dept. of Computer Science Engineering, Korea University of Technology and Education)

** 캘리포니아주립대학(산호세) 항공 및 산업기술학과(Dept. of Aviation and Technology, San Jose State University)

· 제1저자 (First Author) : 김유두 (Yu-Doo Kim, Tel : +82-10-2910-0992, email : kydman@koreatech.ac.kr)

· 투고일자 : 2012년 10월 17일

· 심사(수정)일자 : 2012년 10월 18일 (수정일자 : 2012년 12월 20일)

· 게재일자 : 2012년 12월 30일

위해서는 각 플랫폼에 맞는 애플리케이션을 모두 구현하여 제공하여야만 하기 때문에 개발 비용의 문제가 있다. 따라서 모든 스마트 단말에 기본 탑재되는 브라우저를 활용하는 웹 기반의 콘텐츠 공유 시스템이 제공된다면 플랫폼에 상관없이 서비스를 제공할 수 있다.

웹 기술은 어느 단말에서나 접근이 가능하기 때문에 다양한 스마트 기기에 공통된 서비스를 제공하기에 적합하다. 하지만 기존의 웹 기술은 파일을 접근하거나 전송할 수 있는 부분이 제공되지 않아 콘텐츠 공유를 위한 서비스를 구현 할 수 없었다. 하지만 현재 HTML5 표준에서는 File API와 Web Socket 등의 기술을 기본으로 제공하기 때문에 웹 기반 파일 전송 시스템을 구현할 수 있는 기반 기술이 갖추어져 있다. 또한 스마트 단말은 모두 HTML5 표준을 모두 준수하고 있기 때문에 웹 기반의 기술을 동작 하는 데에 어려움이 없다.

이에 본 논문에서는 공개된 웹 라이브러리를 활용하여 웹 기반 파일 전송 시스템을 구현하여, 하나의 웹 애플리케이션으로 다양한 스마트 단말에서의 파일 전송이 원활히 이루어 질 수 있게 하였다. 또한 기존의 플랫폼 각자의 파일 전송 시스템을 구현한 프로그램과의 성능 분석을 통해 웹 기술의 보완되어야 할 부분에 대한 연구를 수행 하였다.

본 논문은 HTML5 기술의 소개와 현재 진행 현황

을 살펴보는 것을 시작으로, HTML5를 활용한 웹 기반 파일 전송 시스템에 대한 구성에 대해 소개 한다. 그 다음으로, 구성된 시스템과 기존 방식의 파일 전송을 수행하여 그 성능을 분석하고 결론을 맺는다.

II. HTML5 기술

웹 기술이 급격히 확산된 가장 큰 계기는 HTML이라는 마크업 언어를 통해 정보를 표현하고, 이를 다양한 단말에서 브라우저를 통해 쉽게 접속할 수 있게 하여 언제 어디서나 표준 브라우저를 통해 웹을 사용할 수 있게 하였기 때문이다. 그림 1 에서와 같이 1993년 HTML 1.0 표준이 만들어 진 후, 1997년 HTML 4.0과 1999년의 HTML 4.01 표준이 만들어지는 순간에 웹 기술이 폭발적으로 성장 하였다[2]. 하지만 HTML은 단순 정보 표현과 연결을 위주로 제공되어 확장성이 부족하였다. 이에 W3C에서는 1999년부터 확장성에 초점을 두고 XHTML(Extensible Hypertext Markup Language)개발을 하였으나 복잡하다는 단점이 있었다. 이에, 2004년 WHATWG(Web Hypertext Application Technology Working Group)를 구성하고 다양한 웹 애플리케이션에 효과적으로 사용할 수 있는 보다 단순하면서도 다양한 확장성을 갖

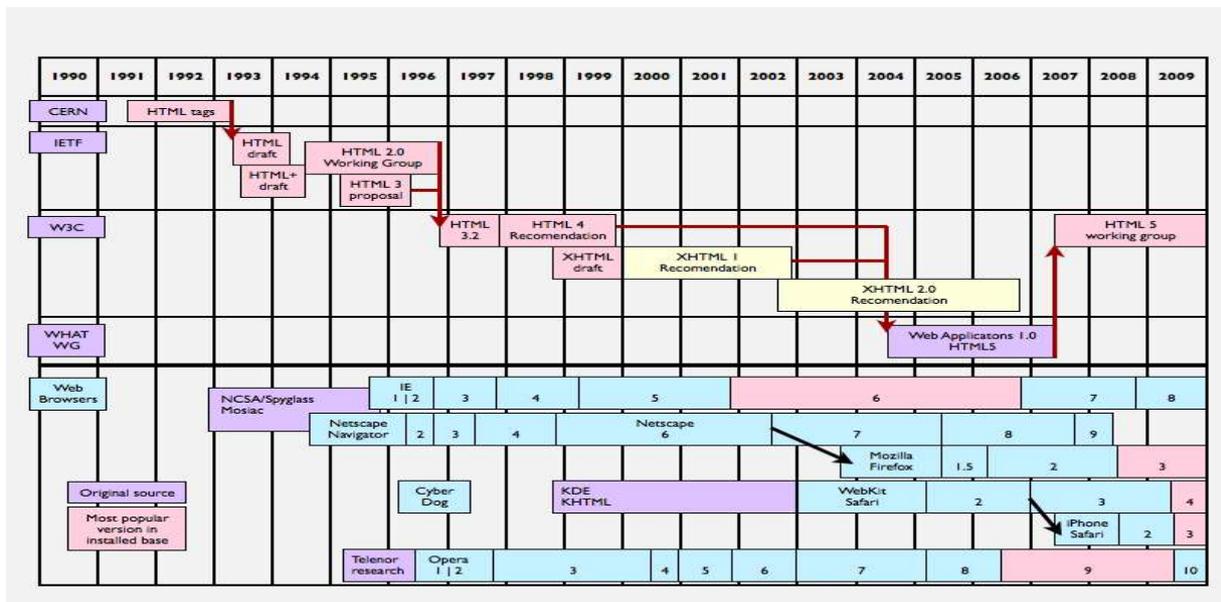


그림 1. HTML 마크업 기술의 발전 과정.
Fig. 1. History of HTML markup technology.

는 HTML 5.0 규격을 만들기 시작하였다[3]. 이에 W3C는 2008년 새로운 HTML 규격을 만들기 위한 HTML WG를 구성하였고, WHATWG의 규격을 기초로 한 새로운 HTML5 표준안을 만들기 시작하였다.

HTML5의 가장 큰 특징은 기존의 HTML에서 확장을 위해 사용하였던 별도의 플러그인 추가와 같은 번거로움을 모두 없애고 언제 어디서나 모든 확장기능이 수행 되도록 하였다[4]. HTML5 표준은 아직 완성되지 않았지만, 현재의 다양한 브라우저가 모두 지원을 하고 있고, 이것이 모든 스마트 단말에 제공되면서 다양한 스마트 단말에 사용될 수 있는 통합 플랫폼으로 발전하고 있다.

특히 HTML 5에서 새롭게 탄생한 File API와 Web Socket을 활용하면 웹 기반의 파일 전송 시스템도 쉽게 구현이 가능하다.

2-1 File API

스마트 단말간의 파일 전송을 위해서는 해당 단말에 저장된 파일을 읽고, 전송된 데이터를 저장하는 기능이 필요하다. 기존의 웹 기술에서는 이러한 부분이 불가능하여 각 플랫폼의 애플리케이션을 개발하여야 했지만 HTML5에서는 File API를 제공하여 이 부분이 해결 되었다. File API는 단말에서 내부파일 접근을 위한 API로 현재 FileList와 Blob 처리, 디렉토리 및 시스템, Writer 등의 세 가지 API 규격을 각각 만들고 있다[5].

2-2 Web Socket

Web Socket API 규격에서는 원격 서버와의 양방향 통신을 가능하도록 하는 웹 소켓을 이용하는 웹 페이지를 가능하도록 API를 정의한다. 웹 소켓에 대한 규격은 IETF에서 표준화 작업을 진행 중에 있다[6].

Web Socket 은 HTML 5에서 간단한 스크립팅으로 서버와의 데이터 통신이 가능하게 구현된 것으로, 이를 활용하여 플랫폼이 다른 각 스마트 단말에서 브라우저를 통해 접속하면 데이터가 전송 될 수 있도록 구현 할 수 있다.

III. 웹 기반 파일 전송 시스템

다양한 스마트 단말 플랫폼을 모두 지원하기 위한 웹 기반의 파일 전송 시스템은 매우 간단하게 구성되어 있다.

그림 2에서와 같이 파일을 전송하는 Sender와 파일을 받는 Receiver로 구성되어 있으며, 각 단말은 File API를 통해 단말의 파일 데이터를 읽거나 쓰게 된다. 또한 각 단말간의 데이터 전송은 Web Socket을 이용한다.

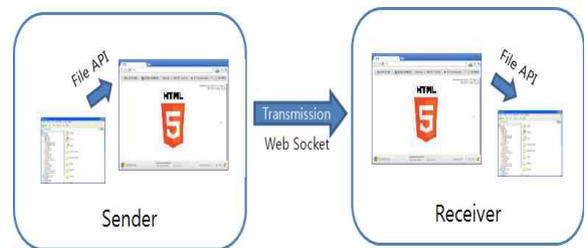


그림 2. 시스템 구성.

Fig. 2. System architecture.

본 시스템은 파일 전송의 성능 분석을 목표로 한 것으로 다양한 기능보다는 각 단말간의 파일이 웹 기반에서 정상적으로 전송 되는 것에 초점을 두어 구현 하였다. 따라서 환경이 다른 스마트 단말에서의 정상적인 동작을 확인하기 위하여 같은 웹 애플리케이션을 데스크탑 PC와 스마트 폰에서 실행하여 테스트를 수행 하였다.

3-1 데스크탑 에서 실행

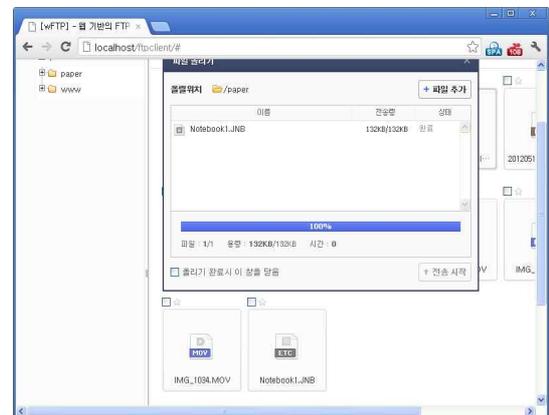


그림 3. 데스크탑 에서 실행.

Fig. 3. Running on the desktop.

그림 3은 데스크탑 PC에서 파일 전송 시스템에 접속하여 파일을 전송하는 모습을 보여주고 있다. 구현한 시스템은 Node.js 기반의 FTP 프로토콜 전송 라이브러리인 wFTP를 이용 하였다[7].

3-2 스마트폰 에서 실행

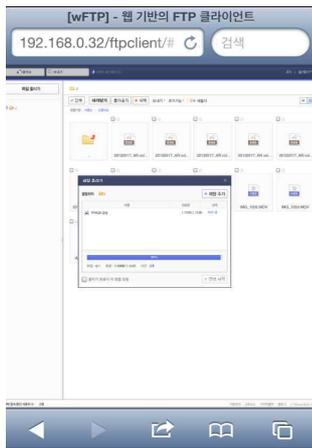


그림 4. 스마트폰 에서 실행.

Fig. 4. Running on the smart phone.

그림 4는 스마트 폰에서 동일한 웹 애플리케이션을 실행한 화면이다. 스마트폰 에서도 데스크탑 PC와 마찬가지로 파일 전송이 정상적으로 이루어지고 있음을 볼 수 있다. 이와 같이 웹 기반의 시스템은 한번의 개발로 스마트폰과 데스크탑 PC 등 브라우저가 실행되는 모든 환경에서 문제없이 동작이 된다.

스마트폰은 데스크탑 PC와 달리 화면이 작고 사용자가 컨트롤을 정교하게 할 수 없는 단점이 있다. 따라서 이 부분은 반응형 웹 기술인 미디어 쿼리를 활용하여 UI(User Interface)만 간단하게 변경하여 주변 더욱 완성도 높은 시스템이 될 수 있다[8].

IV. 성능 분석

웹 기반 파일 전송 시스템과 각 플랫폼에 의해 개발된 파일 전송 시스템의 성능 분석을 위해 표 1에서와 같이 iPhone 4 스마트폰과 Quad 2.5Ghz 프로세서가 탑재된 데스크탑 PC 환경에서 테스트를 진행 하였다.

표 1. 테스트 환경.

Table 1. Testing environment.

System	Hardware and OS	Network
Smart phone	iPhone 4 iOS 6	WiFi
Desktop	Quad 2.5Ghz Windows XP	WiFi

동일한 환경에서의 성능 측정을 위해 네트워크는 WiFi를 사용하여 같은 AP(Access Point)에 연결된 상태에서 진행 하였다.

성능 분석의 방법은 10MB, 20MB, 30MB, 40MB의 용량을 가지는 4가지의 파일을 준비하여 웹 시스템과 각 플랫폼의 시스템을 활용하여 동일 목적지 단말에 데이터를 업로드 하거나 다운로드 하는 속도를 측정 하였다.

4-1 웹 기반 시스템

그림 5와 6은 HTML5를 활용한 웹 기반 파일 전송 시스템의 파일 업로드 및 다운로드 속도를 스마트폰과 데스크탑에서 동작하였을 때 측정된 값을 나타낸다. 웹 기반 파일 시스템에서는 단말의 종류에 관계없이 파일 전송 속도가 거의 비슷하게 나타나고 있다.

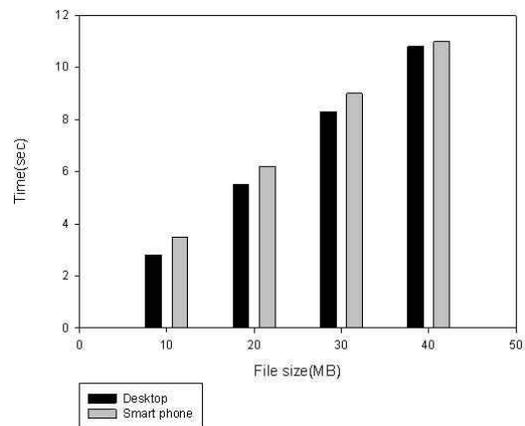


그림 5. 웹 시스템 다운로드 속도.

Fig. 5. Download speed on the web system.

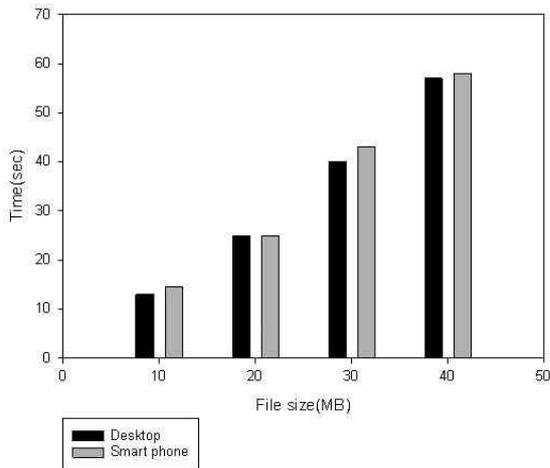


그림 6. 웹 시스템 업로드 속도.

Fig. 6. Upload speed on the web system.

하지만 같은 용량의 데이터를 전송하는데 있어, 웹 기반 시스템은 업로드 속도가 현저히 느리게 나타난다. 이는 업로드의 경우 Web Socket을 통해 구현하였고, 다운로드의 경우 웹 브라우저의 기본 TCP 방식의 다운로드를 활용하였기 때문인데, 이를 통해 Web Socket의 속도가 아직은 느리다는 것을 알 수 있다.

4-2 각 플랫폼의 시스템

그림 7과 8은 위와 동일한 단말에서 웹 기반 시스템을 사용하지 않고 동작한 결과를 나타낸다. 이 결과는 각 플랫폼의 환경에 맞는 별도의 Native 형태로 구현한 애플리케이션으로, 데스크탑과 스마트폰에서의 프로그램은 각각 따로 플랫폼에 맞추어 구현된 것이다.

성능 분석을 위해 별도의 Native 프로그램을 구현하지 않고, 현재 가장 많이 사용되고 있는 파일 전송 프로토콜인 FTP(File Transfer Protocol)를 지원하는 상용 프로그램을 각각 플랫폼에 맞게 사용 하였다.

Native 프로그램을 사용하였을 때의 속도는 다운로드와 업로드 모두 데스크탑 환경에서 빠른 속도를 보이고 있으며, 스마트폰에서는 다소 느린 속도로 파일이 전송됨을 볼 수 있다. 이는 단말의 성능이 파일 전송 속도에 미치는 영향이 있음을 알 수 있다.

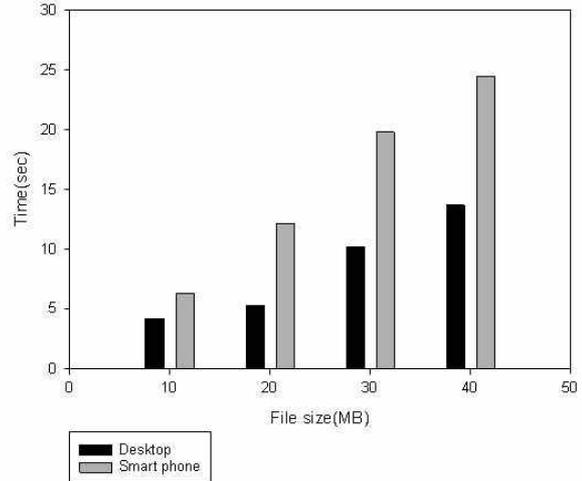


그림 7. Native 시스템 다운로드 속도.

Fig. 7. Download speed on the native system.

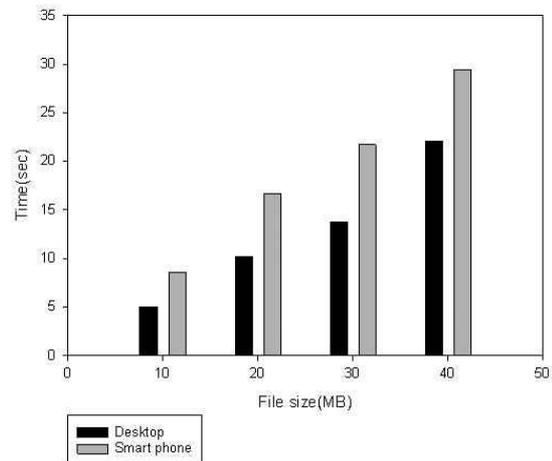


그림 8. Native 시스템 업로드 속도.

Fig. 8. Upload speed on the native system.

4-3 성능 비교

그림 9와 10에서는 업로드와 다운로드 속도를 각각 다른 단말에서 웹 기반 시스템과 Native 시스템에서 동작 하였을 때를 비교한 결과를 보여주고 있다. 본 논문에서는 단말간의 속도 영향 보다는 웹 시스템이 기존 Native 시스템과의 속도 차이가 얼마나 나타나는지가 가장 중요하기 때문에 그 부분에 초점을 맞추어 분석해 보았다.

다운로드 속도를 먼저 살펴보면 웹 기반 시스템이 더 빠른 성능을 보여주고 있다. 하지만 다운로드 시스템은 Web Socket을 사용한 것이 아닌 브라우저에서 기본 제공되는 TCP 방식의 직접 다운로드 방식의 성능이므로, 이는 TCP와 FTP방식의 성능 차이로 분

석 될 수 있다. 따라서 본 논문에서 중점적으로 확인하고자 하는 웹 기반의 시스템의 속도는 Web Socket을 사용한 업로드 속도에서 확인을 할 수 있다.

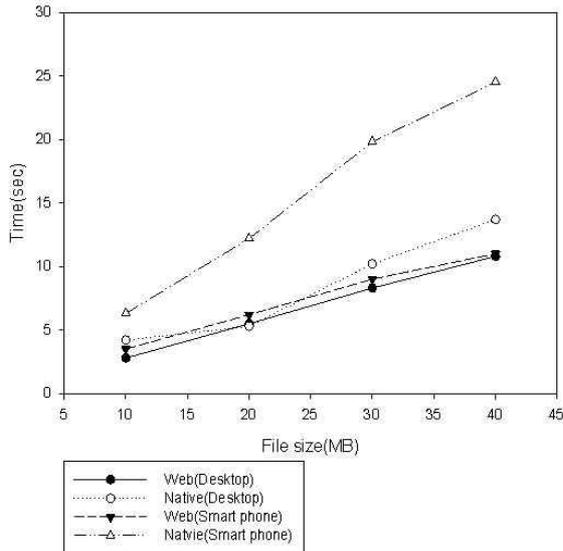


그림 9. 다운로드 속도 비교.

Fig. 9. Comparing download speed.

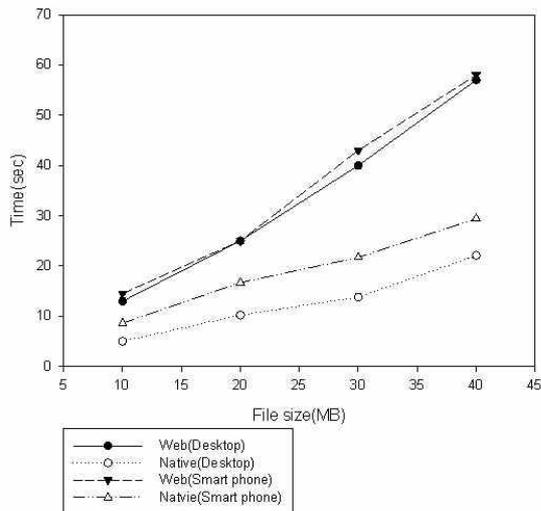


그림 10. 업로드 속도 비교.

Fig. 10. Comparing upload speed.

그림 10에서의 결과를 분석하면 웹 방식의 업로드 속도는 기존 Native방식의 업로드 속도보다 용량이 늘어날수록 더더욱 느려지고 있음을 볼 수 있다. 이는 아직까지 Web Socket 방식이 파일 전송을 하는 데 있어서는 미흡한 성능을 보여주고 있음을 알아 낼 수 있다.

V. 결 론

본 논문에서는 HTML5를 사용하여 웹 기반의 파일 전송 시스템을 구축하고, 이를 기존의 플랫폼 별로 구현된 파일 전송 시스템과의 성능을 분석하였다. 결과에서 보는바와 같이 성능 면에서는 웹 기반의 시스템이 아직 미흡하지만, 다양한 스마트 단말에 모두 다르게 시스템을 구현해야 하는 것에 비하여 웹 시스템은 한 번의 개발로 모두 적용이 가능하다는 장점이 있다. 따라서 현재의 HTML5 기술을 통해 구현된 파일 전송 시스템은 모든 스마트 단말을 지원하는 호환성과 한 번의 개발로 적용되어 개발 비용을 감소하는 효과를 통해 1인 N 단말의 시대에 적합하다고 할 수 있다. 하지만 웹 기술에서의 파일 전송 속도 향상 연구를 통해 기존의 시스템과의 성능 격차를 줄이는 연구 또한 필요하다.

감사의 글

이 논문은 2012년도 한국기술교육대학교 교수 해외 파견 연구비 지원에 의하여 연구되었음.

Reference

- [1] G. Motta, N. Sfondrini, D. Sacco, "Cloud Computing: An Architectural and Technological Overview," *2012 International Joint Conference on Service Sciences*, pp. 23-27, 2012.
- [2] Apple Insider, Why Apple is betting on HTML 5: a web history, <http://is.gd/w9S9a9>.
- [3] Web Hypertext Application Technology Working Group, <http://www.whatwg.org/>.
- [4] HTML5 Differences from HTML4, Working Draft, World Wide Web Consortium, 5 April 2011.
- [5] File API W3C, <http://www.w3.org/TR/FileAPI/>.
- [6] The WebSocket API Editor's Draft, <http://dev.w3.org/html5/websockets/>.
- [7] wFTP Project, <https://github.com/seogi1004/wftp>.
- [8] Park. Joon Woo, Yu. Geum, "Considering a Responsive Web Contents which bases on N-screen

in the Smart Media Environment,” *Journal of Digital Design*, vol. 35, pp. 257-266, 2012.07.

김 유 두 (金裕斗)



2007년 2월 : 한국기술교육대학교
인터넷공학 졸업 (공학사)
2009년 2월 : 한국기술교육대학교 대학원
정보미디어공학과 졸업 (공학석사)
2009년 3월~현재 : 한국기술교육대학교
대학원 컴퓨터공학과 (박사과정)
관심분야 : 모바일 P2P, 웹 플랫폼

김 모 한 (金模漢)



1967년 2월 : 서울대학교 공과대학
공업교육학과 (자동차공학사)
1975년 2월 : 미국 마이아미 대학교
대학원 공업교육학과(공업교육학석사)
1973-1978: 코닐락 (Kornylak) Material
Handling & Automation Company,

R&D engineer.

1978년 2월 : 미국 신시나티대학교 대학원 공업교육학과
졸업 (공업교육학박사)

1979년 -1986년 : 위스컨신 주립대학 산업기술과 조교수,
부교수

1979-1980 : 홀부라이트 교수

1986 년 8월-~현재 : 캘리포니아주립대학 (산호세) 항공
및 산업기술학과 교수

1986-1989 : 미국 자동인식협회 교육 고문 역임

관심분야 : 무선 자동화 응용 및 마이크로 로직 컨트롤러

문 일 영 (文日永)



2000년 2월 : 한국항공대학교
항공통신정보공학과 (공학사)

2002년 2월 : 한국항공대학교 대학원
항공통신정보공학과 (공학석사)

2005년 2월 : 한국항공대학교 대학원
정보통신공학과 졸업(공학박사)

2004년 ~2005년 : 한국정보문화진흥원

선임연구원

2005년 3월~현재 : 한국기술교육대학교 컴퓨터공학부
부교수

관심분야 : 무선 인터넷 응용, 무선 인터넷, 모바일 IP