

# PC 원격 제어 기법을 사용한 모바일 이메일 전송 시스템 설계 및 구현

송혜주\*, 김현주\*\*, 유현정\*\*\*, 이종우\*\*\*\*

## 요약

장소와 시간에 관계없이 언제든 네트워크에 접속해 컴퓨팅할 수 있는 유비쿼터스 환경은 디지털 기술의 발전으로 이제 전문가 영역에서 생활 현장으로까지 퍼져나가고 있다. 본 논문에서는 언제 어디서나 멀리 떨어진 곳에서도 휴대폰을 통해 내 PC를 제어할 수 있는 모바일 PC 원격 제어 시스템 구현을 목표로 한다. 특히, 대표적인 PC 원격 제어 기능 중 하나인 메일 전송 기능을 설계하고 구현함으로써 휴대폰을 통해 내 PC에 접속하여 원하는 파일이 첨부된 메일을 전송할 수 있도록 하였다. 클라이언트 모듈은 WIPI Jlet으로 구현하였으며 KTF 휴대폰에서 동작하도록 개발하였고, 서버 모듈은 J2SE 환경에서 개발함으로써 서버 운영체제와 무관하게 실행될 수 있도록 하였다. 동작 실험을 통해 본 논문에서 개발한 PC 원격 제어 소프트웨어가 올바르게 동작함을 확인하였으며, 본 논문에서 개발한 소스 코드를 공개함으로써 향후 이메일 전송 이외의 PC 기능들이 모바일 에이전트 형태로 구현될 수 있는 발판을 마련한다는 것이 본 논문이 기여하는 바라 할 것이다.

## A Design and Implementation of Mobile E-mail Transmission System Using PC Remote Control Technique

Hye-Ju Song\*, Hyun\_Ju Kim\*\*, Hyun\_Jung Kim\*\*\*, Jongwoo Lee\*\*\*\*

## Abstract

Recently, the ubiquitous computing environment being able to connect all kinds of computing elements at anytime anywhere becomes widespread in human life. In this paper, we focus on a PC remote control system using cellular phones. Especially, we design and implement an mobile e-mail transmission system using PC remote control technique. By using our remote controller, cellular phone users can login into their own PC, and then send any file to others by e-mail attachment. To show the correct running of our system, real demonstration results are presented. We are sure that by opening our source code to the public our results can play an important role to encourage development of the various mobile remote control functionalities.

Keyword PC Remote Control, Mobile Software, WIPI Jlet, Email transmission, James Java Mail

## 1. 서론

‘언제 어디서나 존재한다.’ 라는 뜻의 유비쿼터스(ubiquitous)는 장소와 시간에 관계없이 언제든 네트워크에 접속해 컴퓨팅할 수 있는 정보통신 환경을 뜻한다. 컴퓨터와 휴대전화 등 디지털 기술의 발전으로 전문가 영역에서만 존재하던 유비쿼터스는 이제 생활현장으로 계속 퍼져나가고 있다. 본 논문에서는 언제 어디서나 멀리 떨어진 곳에서도 내 PC를 제어할 수 있는 휴대폰 기반의 PC 원격 제어 시스템 개발 초점을 맞추고 있다. 그동안 유선 네트워크를 통한 제어는 떨어져 있는 여러 PC들 사이에서 적용되어 왔다 [1]. PC간 원격 제어에서는 단말기와 서버의 디

※ 제일저자(First Author) : 송혜주  
접수일자:2007년02월14일, 심사완료:2007년03월20일  
\* 숙명여자대학교 멀티미디어과학 전공  
bigrain@sookmyung.ac.kr  
\*\* 숙명여자대학교 멀티미디어과학 전공  
\*\*\* 숙명여자대학교 멀티미디어과학 전공  
\*\*\*\* 숙명여자대학교 멀티미디어과학 전공 조교수  
▣ 본 연구는 숙명여자대학교 2006년도 교내연구비 지원에 의해 수행되었음

바이스 구조가 유사하기 때문에 원격 제어가 쉽게 이루어지긴 하지만, 컴퓨터 장치 사용 가능 장소에 제약이 따른다는 단점이 있다. 본 논문에서는 사용자가 휴대폰을 이용하여 무선 네트워크를 통해 자신의 PC를 원격으로 제어하는 기법을 제안하고 있으며, 그 구체적인 응용 분야로 휴대폰을 통해 개인의 컴퓨터에 접속하여 원하는 파일을 이메일에 첨부해 전송할 수 있는 원격 이메일 전송 시스템을 설계하고 구현하였다. 이것은 가정에 있는 장비들을 연결하여 서로 통신할 수 있도록 함으로써 어디서나 가정 내 장비를 제어하거나 통신할 수 있게 하는 홈 네트워크의 일부분이라고도 할 수 있다[2]. 본 논문에서 클라이언트인 모바일 단말기 종류로 휴대폰을 채택한 이유는 휴대폰이 PDA나 네트워크 PMP 등의 단말기보다 범용적이고 휴대하기가 더 편하기 때문이다. 이러한 휴대폰을 통해 서버(내 컴퓨터) 상의 디렉터리와 파일 목록을 보면서, 원하는 파일을 첨부하여 이메일로 전송할 수 있게 함으로써 인터넷이 가능한 어느 곳이라던 어디서나 원하는 파일을 받아 볼 수 있게 한다는 것이 본 논문의 주된 목표라고 할 수 있다. 본 논문에서는 기존 모바일 PC 원격 제어 시스템의 단점인 사용자 인터페이스를 보완하였고, PC 상에서 아웃룩 익스프레스 클라이언트를 이용해 메일을 보내던 것을 James Java Mail 서버 [3]를 이용한 SMTP(Simple Mail Transfer Protocol) 방식의 직접 전송으로 개선하였다. 이는 PC에 아웃룩 익스프레스가 설치되어 있지 않은 경우에도 대비하기 위해서이다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 연구에 대해 살펴보고, 3장에서는 서버와 클라이언트 간 통신 순서의 전체적인 구조와 PC 원격 제어 시스템의 구체적인 설계 방향과 특징들을 소개한다. 4장에서는 구현된 시스템의 실행 테스트 결과를 보이고, 끝으로 6장에서는 결론과 향후 연구 과제를 제시한다.

## 2. 기존 연구

본 연구와 유사한 모바일 서비스로는 인투모바일사의 My PC[4]가 있는데, My PC는 파일관리, 이메일관리, PIMs 등의 서비스를 제공한다.

My PC 서비스는 휴대폰에서 자기 PC의 디렉터리를 조회하고 파일을 읽을 수 있으며, 파일을 첨부해 이메일을 보낼 수 있다. 이메일 관리 부분에서는 받은 편지함과 보낸 편지함의 조회와 삭제, 메일쓰기와 회신, 전체 회신과 전달을 할 수 있다. PIMs 관리 부분에서는 연락처와 일정, 메모, 작업의 조회, 편집, 삭제, 추가를 할 수 있고 연락처 번호로 전화를 걸 수 있다. 이처럼 많은 기능을 가지고 있는 My PC이지만 정액제로 가입해야 하는 번거로움과 부담 때문에 사용자가 많지 않다. 거의 모든 휴대폰이 무선 인터넷을 지원하는 요즘 인터넷의 흔한 기능인 이메일 관리를 비용을 지불하고 쓴다는 것은 사용자의 입장에서는 낭비일 것이다. 또한 휴대폰 자체적으로 흔히 구현되어 사용하고 일정관리 프로그램 역시 굳이 무선 인터넷에 접속해서 사용해야 할 이유는 없을 것이다. 본 연구와 기존 My PC 서비스의 차이점을 요약하면 표 1과 같은데, 본 논문에서 구현한 시스템은 소스를 공개함으로써 향후 관련 연구의 활성화에 기여하고자 했다는 점과 서버 상에서 메일 전송 시 아웃룩 같은 기존 상용 제품이 아닌 역시 공개 소스인 자바 메일 패키지를 사용했다는 점이 가장 큰 차이라고 할 수 있다.

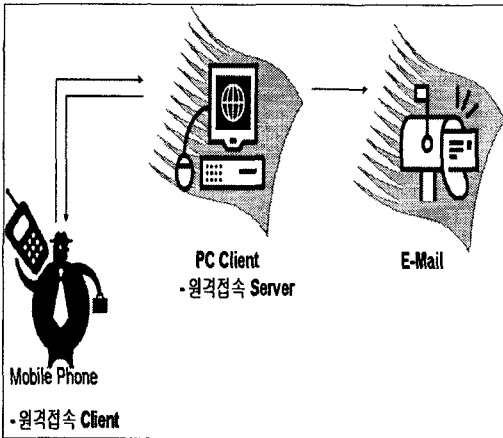
<표 1> My PC[4]와 본연구와의 장단점 비교

서비스 구분	My PC	본 연구
사용비용	유료	무료
서버유형 제한 여부	MS-Windows 로 제한	제한 없음 (자바 런타임)
소스 공개 여부	비공개	공개
메일 전송 방식	아웃룩 활용	James Java Mail API
기능	상용 서비스인 관계로 다양함	메일 에이전트만 지원

## 3. 모바일 원격 PC 제어기 설계 및 구현

### 3.1 시스템 전체 구조

(그림 1)은 본 논문에서 개발한 모바일 원격 PC 제어기 소프트웨어의 구조를 보이고 있다. 클라이언트는 사용자의 휴대폰을 가정하고 있으며, 휴대폰용 클라이언트 프로그램은 WIPI Jlet으로 구현하였고 서버와의 원격 접속 API로는 KTF용 테스트 소켓인 BillComm 소켓을 이용하였다[5]. BillComm 소켓은 WIPI Jlet에서 제공하는 기본 소켓 API에 모바일 인터넷 사용료 계산 기능까지 추가된 버전인데, WIPI Jlet 프로그램을 에뮬레이터에서 실행시킬 때에는 필요 없지만 실제 휴대폰 상에서 실행시키고자 할 때에는 반드시 사용하도록 KTF 측에서 권고하고 있다. 사용자 PC에서 실행될 서버 프로그램은 J2SE 프로그래밍 환경에서 구현되었고, 서버에서 이메일 전송 시 사용할 API로는 James Java Mail 이메일 서버 패키지를 활용하였다. 아울러 로그인 인증을 위한 데이터베이스로는 MySQL[6]을 사용하였다.



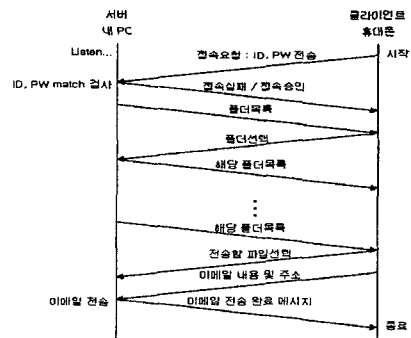
(그림 1) 시스템 전체 구조

### 3.2 클라이언트/서버 간 통신 흐름

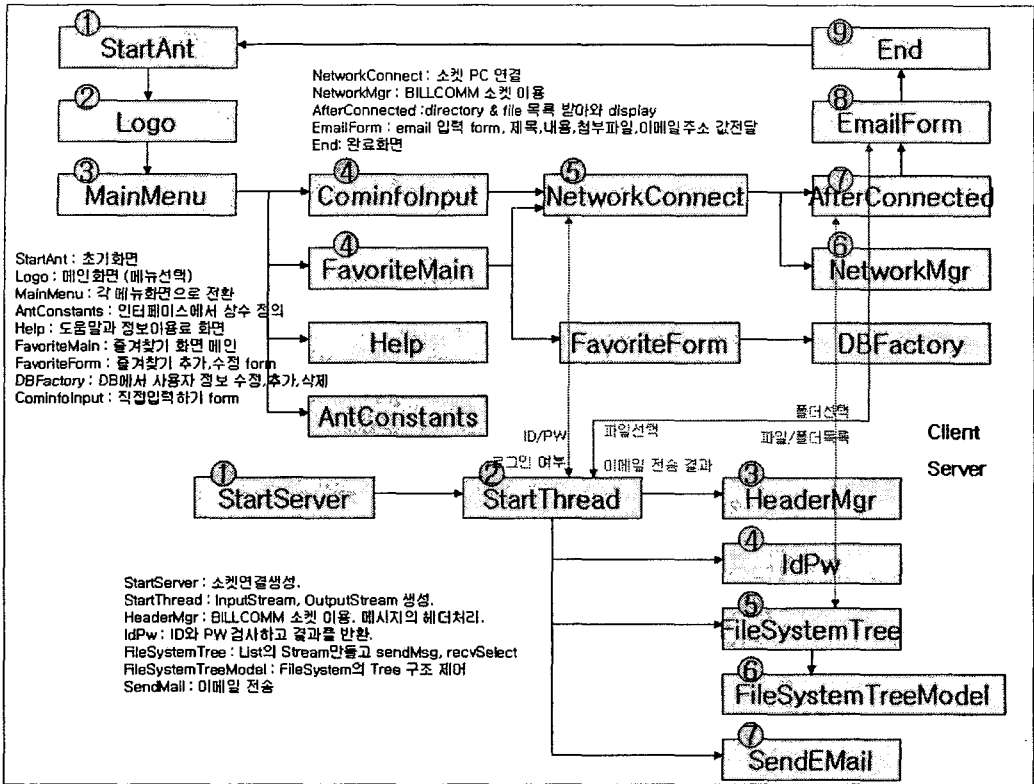
내 PC에서 실행되는 서버는 우선 휴대폰으로부터의 접속을 기다린다. 이 때, 내 PC에는 서버 프로그램과 아울러 James 자바메일 서버와 MySql 데이터베이스가 실행되고 있다. 서버 프로그램은 대기하고 있다가 휴대폰의 접속을 받으면 전담 스레드를 실행시켜 서비스를 시작한다. 휴대폰에서는 접속 후 로그인을 위한 ID와 PW를 서버로 보낸다[7, 8]. 로그인에 실패할 경우 로그인 실패 메시지를 휴대폰에 전송하고 해당 접속을 끊는다. 로그인에 성공할 경우 로그인

성공 메시지를 휴대폰에 전송하고 서버 측 파일 시스템의 기본 경로인 C:\의 파일 목록을 만들어 휴대폰에 전송한다. 휴대폰은 로그인 실패 메시지를 받으면 접속을 끊고 실패했다는 창을 띄운 뒤 다시 처음 페이지로 돌아간다. 로그인 성공 메시지를 받으면 파일 목록을 받고, 이를 파싱한 뒤 화면에 출력한다. 그림 2는 이 같은 클라이언트와 서버 간의 통신 흐름을 보이고 있다.

휴대폰 사용자가 화면에 출력된 목록을 브라우징하면서 진입할 서브디렉터리를 선택하면 해당 폴더의 인덱스 값이 서버로 전송된다. 서버는 그 값에 해당하는 폴더의 하위 파일 목록을 휴대폰에 다시 전송한다. 이와 같은 방법으로 휴대폰 사용자는 내 PC 상의 파일 목록을 자유롭게 브라우징 할 수 있다. 파일 목록을 이동하다가 이메일에 첨부할 파일을 발견하면 해당 파일을 선택한다. 휴대폰에서 이 항목이 파일인지 폴더인지 알고 있기 때문에 파일을 선택했을 경우에 바로 이메일 작성화면으로 넘어간다. 선택한 파일의 인덱스 값이 서버로 전송되면 서버 역시 선택된 항목이 파일인지 폴더인지 알고 있으므로 더 이상의 파일 목록 보내기를 멈추고 이메일 주소와 이메일 제목, 내용을 휴대폰으로부터 받을 준비를 한다. 이때 서버는 현재 경로를 저장해 둔다. 그 이유는 휴대폰 사용자가 방금 전에 보던 목록으로 다시 돌아가 다른 파일을 선택할 수도 있기 때문이기도 하고, 서버에서 이메일 전송 시 이 경로를 첨부 파일을 찾기 위한 경로로 활용하기 때문이다.



(그림 2) 클라이언트/서버 간 통신 흐름도



(그림 3) 클라이언트/서버 모듈의 실행흐름

휴대폰 사용자는 이제 이메일 주소를 확인하고 이메일의 제목과 내용을 입력한다. 이때 만약 사용자가 뒤로 버튼을 누르면 방금 전의 파일 목록 선택 화면으로 돌아간다. 이메일 작성을 마치고 확인 버튼을 누름과 동시에 이메일 주소와 제목, 내용이 서버로 전송된다. 서버는 클라이언트로부터 받은 이메일 주소와 제목, 내용, 그리고 첨부 파일을 한데 묶어 이메일 메시지를 구성하고 메일을 보낸다. 이메일 전송에 성공하거나 서버는 휴대폰에 이메일 전송 성공 메시지를 보내고 해당 휴대폰과의 접속을 끊는다. 휴대폰도 성공 메시지를 받으면 소켓을 끊고 성공화면을 출력한다.

### 3.3 모듈 별 설계 및 구현

#### 3.3.1 서버 모듈

(그림 3)의 하단부는 서버 모듈의 실행 흐름을 보이고 있는데, 서버 부분의 클래스 별 실행 흐름을 요약하면 다음과 같다. 서버는 StartServer 에서 처음 시작한다. StartServer에

서는 서버소켓을 만들고 포트를 열어둔 상태로 StartThread를 호출하고 클라이언트의 접속을 기다린다. 클라이언트가 접속하면 StartThread는 새 스레드를 생성한다. 그리고 해당 소켓을 감싸는 InputStream과 OutputStream을 만들고 HeaderMgr 객체를 생성한다. HeaderMgr 객체 생성 작업은 BillComm 소켓을 통해 이동할 스트림에 헤더를 붙이고 떼어내기 위함인데, 보통의 경우에는 필요 없는 단계이나 모바일 데이터 통신의 패킷 크기 별 과금을 위해 KTF가 권고하고 있는 단계이므로 필수적이다.

서버와 클라이언트 간의 통신 설정이 완료된 후 클라이언트가 ID와 PW를 서버로 전송하면, 서버는 IdPw 클래스를 통해 데이터베이스와 연동하여 ID와 PW를 검사한다. IdPw 클래스로부터의 결과를 받아 StartServer는 로그인 시도 결과를 클라이언트에 전송한다. 실패인 경우 소켓을 닫고 대기 상태로 돌아가고, 성공인 경우 바로 FileSystemTree를 호출해 파일시스템 상의 파일 목록을 클라이언트에 전송한다.

FileSystemTree에서는 FileSystemTreeModel을 호출하여 파일 목록을 작성하는데, FileSystemTreeModel에서는 파일시스템의 구조를 트리형태로 표현한 모델을 만든다[7]. 또, FileSystemTreeModel은 FileFolder를 호출하여 파일과 폴더 구조의 정보를 가져온다. 클라이언트는 받은 폴더 목록을 확인하고 원하는 폴더의 인덱스를 서버로 전송한다. 서버에서는 그것을 확인하고 마찬가지로 해당 하위 폴더의 목록을 클라이언트로 전송한다.

사용자가 이메일에 첨부할 파일을 선택하면 서버는 이를 확인한 후 클라이언트로부터 이메일 정보를 받고 이 정보대로 이메일 메시지를 구축한다. 이메일 메시지가 완성되면 James 자바메일 서버를 이용해 보내고 그 전송결과를 클라이언트에 보낸 뒤 종료한다.

### 3.3.2 클라이언트 모듈

(그림 3)의 상단부에서는 클라이언트 측의 실행 흐름 순서를 보이고 있는데, 이를 요약하면 다음과 같다. StartAnt 클래스에서 초기화면을 구동시키고, 버튼 클릭 시 각 메뉴를 선택할 수 있는 메인화면인 Logo 클래스를 실행시킨다. 메인화면에서는 ‘즐거찾기’, ‘직접입력’, ‘도움말’, ‘정보이용료’의 총 네 가지 메뉴를 선택할 수 있는데, 도움말과 정보이용료를 화면은 Help 클래스가 처리한다. 각 페이지에 관한 상수 정의와 연결 상태는 AntConstants 인터페이스에 정의되어 있다.

즐거찾기를 통한 접속은 FavoriteMain 클래스에서 처리한다. 즐겨찾기 목록 나열 및 신규 등록, 삭제 등을 지원하며 선택된 즐겨찾기를 통해 서버로 바로 접속할 수 있게 해준다[9]. 하나의 즐겨찾기 항목은 서버 IP 주소와 ID, PW, 그리고 이메일 주소의 4개 속성으로 구성되며, 이 중 이메일 주소를 인덱스로 하여 저장되고 관리된다. 즐겨찾기 항목 추가 및 수정, 삭제는 FavoriteForm 클래스와 DBFactory 클래스를 통해 이루어진다.

직접입력 접속 방식을 선택하면 CominfoInput 클래스가 ID와 PW, 이메일 주소, 그리고 서버의 IP 주소를 입력받아 NetworkConnect 클래스로 넘겨준다. NetworkConnect 클래스는 서버와의 소켓 통신을 위한 모든 처리를 전담하는 클래스

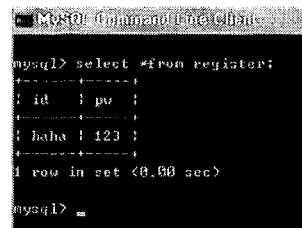
로써 사용자 인증을 위한 서버와의 협력도 담당한다. 즐겨찾기로 접속했을 때에도 접속버튼을 누르면 결국 NetworkConnect 클래스를 거치게 된다.

NetworkMgr 클래스는 서버 측의 HeaderMgr 클래스와 유사한 것으로써 과금 처리를 위해 클라이언트 측 소켓 통신을 감싸고 있는 클래스이다. KTF 측에서는 이를 BillComm 소켓이라 한다. 서버로부터 접속이 승인되면 AfterConnected 클래스가 서버 측으로부터 폴더와 파일 목록을 받아와 표시해주는데, 이 때 각 폴더경로는 자유롭게 이동가능하며 파일을 선택하면 EmailForm 클래스를 실행한다. EmailForm 클래스에서는 보낼 이메일 주소, 제목, 내용을 입력받고, 앞서 선택했던 첨부 파일의 이름과 첨부 파일의 서버상 경로명을 덧붙여 서버로 전송한다.

End 클래스는 이메일이 성공적으로 전송되었을 때 표시되는 화면이며, 다시 보내기를 선택하면 MainMenu로, 종료를 선택하면 StartAnt로 이동한다.

### 3.3.3 데이터베이스 구조

본 논문에서 사용한 데이터베이스는 사용자 인증을 위해서만 사용되므로 매우 간단하다. 테이블 이름은 register이며, 클라이언트가 직접 접속 방식을 통해 서버에 넘겨준 정보 중 사용자 이름(ID)와 암호(PW) 두 애트리뷰트로만 구성되어 있다[10]. 이 두 애트리뷰트는 모두 Varchar 타입으로 정의하였으며, 클라이언트에서 재등록(수정) 및 삭제가 가능하다. 그림 4는 register 테이블의 구조를 보이고 있다.

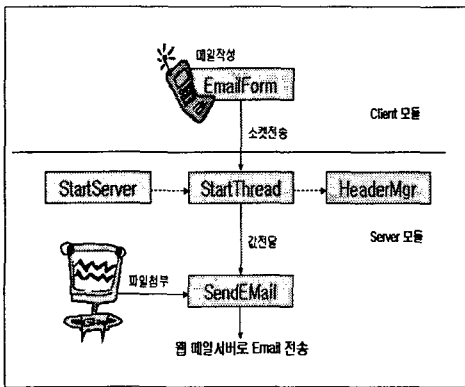


<그림 4> register 테이블

### 3.3.4 이메일 전송 시스템

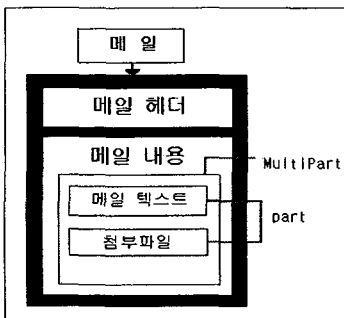
클라이언트가 첨부 파일을 선택한 후 이메일 전송 폼에서 보낼 이메일 주소, 제목, 내용을 입

력하고 전송 버튼을 누르면 이 값들이 BillComm 소켓을 통해 서버로 전달된다. 서버는 이 값들을 이용해 첨부 파일까지 포함된 완전한 이메일 메시지를 구성하고 목표주소로 전송한다. (그림 5)는 클라이언트와 서버 간의 이메일 전송 의뢰/접수 제어 흐름을 보이고 있다.



(그림 5) 이메일 전송 의뢰 및 접수 과정

이메일 메시지는 James 자바 메일 패키지에서 제공하는 마임 메시지(mime message) 클래스를 이용하면 간편하게 구축할 수 있는데, (그림 6)은 이메일 메시지의 구조를 보이고 있다.



(그림 6) 이메일 메시지 구조

(그림 6)에서 알 수 있듯이, 이메일 메시지를 구성하는 각 부분을 part라 하는데 메일 내용이나 제목, 첨부 파일 경로 등이 part의 일종인 셈이다. 이 part들을 모아 놓은 것을 Multipart라 하는데 Multipart 구조체에 part들을 넣으면 이메일 메시지가 구축되는 것이다. 구축된 이메일 메시지는 역시 James 자바 메일 패키지에서 제공하는 Transport 클래스를 통해 목표 이메일

주소로 전송된다. James 자바 메일 패키지는 이메일 전송 시 SMTP(Simple Mail Transfer Protocol)를 직접 사용한다.

James 자바 메일 서버[3]는 SMTP와 POP3, NNTP 등의 프로토콜을 지원하는 자바 메일 엔진으로써 메일 서버가 없는 개인 사용자들이 메일 클라이언트 프로그램을 작성할 수 있도록 자바 언어 기반 오픈소스로 개발되고 있는 서버이다. 하지만 대부분의 상용 웹 메일 서버들은 메일 수신 시 메일을 보낸 서버가 DNS에 등록되어있는 경우에만 메일이 전달되도록 하는 경우가 많기 때문에, James 자바 메일을 사용하여 이메일을 전송한 경우 서버(내 PC)가 DNS에 등록되어 있지 않으면 수신 거절되어 돌아오는 경우도 발생한다.

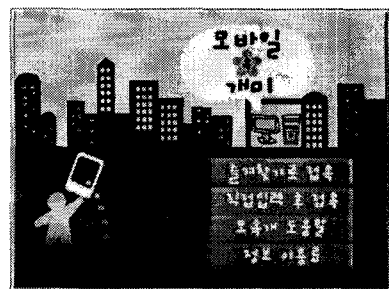
## 4. 실행 및 평가

### 4.1 클라이언트 모듈 평가

클라이언트용 WIPI Jlet 프로그램을 휴대폰에서 실행시키면 (그림 7)과 같은 초기 화면을 보이고, 여기서 사용자가 아무 버튼을 클릭하면 (그림 8)의 화면으로 이동한다.



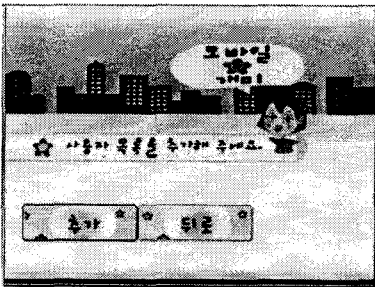
(그림 7) 초기 실행화면



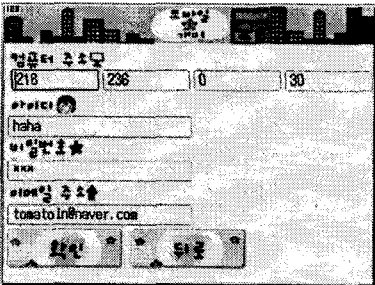
(그림 8) 메인 메뉴 화면

(그림 8)은 주 메뉴 화면을 보이고 있는데, 여기서 즐겨찾기로 접속 또는 직접입력 후 접속 같은 주 메뉴를 선택할 수 있다. 또한 도움말이나 정보이용료에 대한 정보도 볼 수 있다.

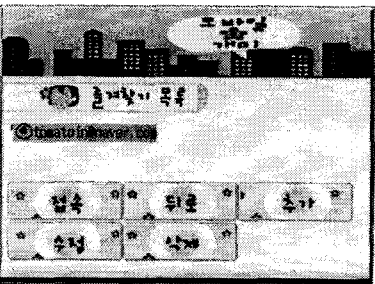
(그림 9)는 즐겨찾기로 접속을 선택했는데 아직 즐겨찾기 항목이 하나도 등록되어 있지 않아 즐겨찾기 목록을 추가하라는 의미의 화면임을 알 수 있다. 여기서 추가를 선택하면 (그림 10)과 같이 서버 정보 및 사용자 정보를 저장할 수 있고, 저장에 성공하고 나면 (그림 11)과 같이 등록된 즐겨찾기 목록이 화면에 보인다. 즐겨찾기는 서버로의 접속을 사용자들이 편하게 할 수 있도록 돕기 위함이며 등록 정보는 수정, 삭제가 가능하다.



(그림 9) 즐겨찾기 초기 접속화면

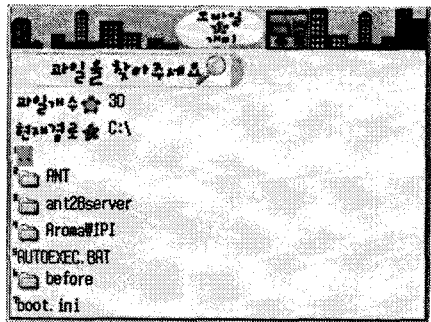


(그림 10) 사용자 정보 추가하기

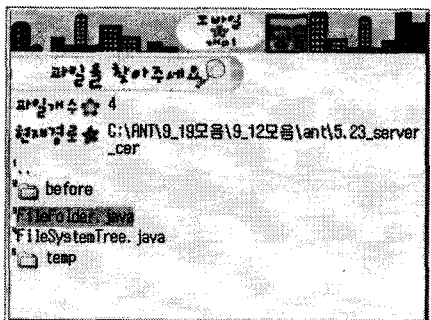


(그림 11) 즐겨찾기 목록 화면

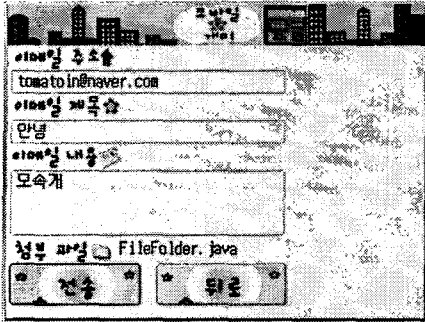
(그림 12)는 ‘즐거찾기로 접속’이나 또는 ‘직접 입력 후 접속’을 통해 서버에 접속한 화면이다. 여기에서는 파일 목록을 이동하면서 바뀌는 현재 폴더 경로와 현 폴더 내의 파일 개수, 그리고 폴더 목록을 볼 수 있는데, 아래위로 파일 목록을 이동하면서 이메일에 첨부할 파일을 선택한다. (그림 13)은 원하는 파일을 선택하는데 파일을 선택하면 (그림 14)와 같은 이메일 작성 화면으로 넘어가 이메일의 주소, 제목, 내용을 입력할 수 있다. (그림 14)의 하단에 첨부파일의 이름이 보이며, 여기서 ‘뒤로’ 버튼을 누르면 그림 13의 화면으로 돌아가고 ‘전송’ 버튼을 누르면 이메일 전송 의뢰를 시도한다. 서버에서 첨부 파일이 포함된 이메일 전송에 성공하면 (그림 15)와 같은 화면이 나온다. (그림 15)에서는 다시 폴더 목록으로 돌아가 파일을 선택해 이메일을 보낼 것인지 아니면 프로그램을 종료할 것인지를 선택할 수 있다.



(그림 12) 서버에 접속한 화면



(그림 13) 폴더목록 이동 화면

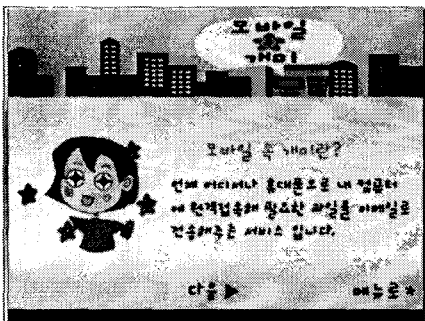


(그림 14) 이메일 작성 화면



(그림 15) 이메일 전송 성공 화면

(그림 16)은 클라이언트 프로그램의 도움말 화면으로 (그림 8)의 주 메뉴 선택 화면에서 '모속개 도움말' 버튼을 누르면 나오는데, 클라이언트 프로그램의 간단한 사용 방법과 만든 사람들의 정보를 볼 수 있다.



(그림 16) 모바일 속 개미 도움말

#### 4.2 서버 모듈 평가

(그림 17)은 사용자가 자신의 서버 상에서 사용자 정보를 등록하는 화면을 보이고 있는데, 사용자가 클라이언트인 휴대폰에서 자신의 서버에 접속할 때 여기에 등록한 아이디 및 비밀번호와

일치할 경우에만 접속이 허용되게 함으로써 다른 사람이 함부로 내 서버에 접근할 수 없도록 하였다. 클라이언트에서는 서버 상에 새로운 계정을 추가할 수 없고 반드시 서버 상에서 제공되는 사용자 등록 프로그램을 통해서만 가능하도록 하였다.



<그림 17> 서버에서의 사용자 등록 폼

### 5. 결론 및 향후 연구 과제

본 연구에서는 사용자가 휴대폰을 이용해 서버인 내 PC를 원격으로 제어하는 시스템을 제안하였다. 즉, 휴대폰을 통해 서버에 접속한 후 원하는 파일이 첨부된 이메일을 보낼 수 있는 모바일 서비스를 설계하고 구현하였다. 비록 현재 유료 서비스로 제공되고 있는 기존 상용 서비스보다는 그 기능 면에서 부족하긴 하지만 본 논문을 통해 개발된 소스 코드를 공개함으로써 향후 휴대폰을 통한 내 컴퓨터 원격 제어 분야의 연구에 도움을 줄 수 있다는 것이 가장 큰 기여라고 할 것이다. 또한 기존 시스템의 단점인 불편한 사용자 인터페이스를 보완하였고, James 자바 메일 서버를 이용한 SMTP 방식으로 이메일을 전송하는 효과적인 시스템을 구현하였다. 실험을 통해 개발된 시스템이 정상적으로 실행됨을 확인하였으며, 동시 접속 시의 성능도 확인함으로써 다수의 클라이언트들이 동시에 본 서



비스에 접속하더라도 문제가 발생하지 않음을 알 수 있었다.

향후에는 이메일 전송 외에 파일 내용 검색, 수신 이메일 검색 등 다양한 서비스 에이전트를 개발할 예정이다. 휴대폰 같은 모바일 단말기는 아직도 그 계산 능력이나 하드웨어적 제약으로 인해 각종 서비스 프로토콜들의 클라이언트 부분을 구현하고 있지 못하다. 따라서 '휴대폰 --> 내 PC --> 실제 서버'로 이어지는 컴퓨팅 의뢰 관계가 점점 그 필요성을 더해갈 것으로 예상된다.

### 참고문헌

- [1] 전호인, "디지털 홈 기술 및 표준화 동향," TTA 저널, 제 88 호, pp.59-74, 2003.
- [2] H.Schulzrinne, et. al., "SIP: Session Initiation Protocol," RFC 3261, 2002.2.1 원격제어 시스템의 단순화된 모델
- [3] <http://james.apache.org>
- [4] 인투모바일(주) <http://www.i2m.co.kr>
- [5] [http://www.mobilelab.co.kr/programming/up/data\\_brew/Billcom\\_요약.doc](http://www.mobilelab.co.kr/programming/up/data_brew/Billcom_요약.doc)
- [6] <http://www.mysql.org>
- [7] MICHAEL J. DONAHOO 외, TCP/IP 소켓 프로그래밍 - 자바 버전, 사이텍미디어, 2002.
- [8] 토모히로오다카, 기초부터 배우는 TCP/IP - JAVA 네트워크 프로그래밍, 성안당, 2002.
- [9] 강상원, 임석진, 심양섭, 모바일 플랫폼 친화통일 위피 프로그래밍, 제우미디어, 2004.
- [10] 최재영, 최종명, 유재우, 프로그래머를 위한 JAVA, 홍릉과학 출판사, 2003.
- [11] 최종명, 박경우, 안동순, 프로그래머를 위한 서버릿/JSP, FREELEC, 2005.

### 송혜주



2007년 : 숙명여자대학교  
멀티미디어과학(학사)  
2007 - 현 재: 숙명여자대학교  
멀티미디어과학(석사과정)  
관심분야 : DBMS, Information Retrieval, 모바일 소프트웨어

### 김현주



2004 - 현 재: 숙명여자대학교  
멀티미디어과학(학사)  
관심분야 : 멀티미디어 콘텐츠, 모바일 소프트웨어

### 유현정



2004 - 현 재: 숙명여자대학교  
멀티미디어과학(학사)  
관심분야 : 멀티미디어 콘텐츠, 모바일 소프트웨어

### 이종우



1990년 : 서울대학교 컴퓨터공학과 졸업(학사)  
1992년 : 서울대학교 컴퓨터공학과 대학원 석사과정 졸업(석사)  
1996년 : 서울대학교 컴퓨터공학과 대학원 박사과정 졸업(박사)  
1998년 - 1999년 : 현대정보기술㈜ 책임연구원  
1999년 - 2002년 : 한림대학교 정보통신공학부 조교수  
2002년 - 2003년 : 광운대학교 컴퓨터공학부 조교수  
2004년 - 현 재 : 숙명여자대학교 정보과학부 멀티미디어과학전공 조교수  
관심분야 : Mobile System Software, Storage Systems, Computational Finance, Cluster Computing, Parallel and Distributed Operating Systems, Embedded System Software