

논문 개요

최근 모바일 기기의 대중화에 따라 교육 분야에서도 휴대전화 단말기를 비롯한 모바일 디바이스 기반의 다양한 응용 소프트웨어 연구가 활발하다.

본 연구는 수기용 교무수첩을 대체하여 교육현장에서 교육업무에 활용할 수 있도록 설계된 전자교무수첩 시스템을 제안한다. 이 시스템은 학교업무에서 주로 사용되는 교무수첩의 기능을 심층 분석하여 교사들의 요구를 충분히 반영하도록 한 것이다. 또한 이미 대중화되고 일반화되어 있는 휴대전화 단말기를 교사용 교무수첩의 주 도구로 제안함으로써 학교업무의 효율성을 극대화하고자 한다. 휴대전화 단말기는 언제 어느 곳에서든 휴대가 간편한 휴대성과 이동성, 개인성의 장점을 가지고 있어 교사들이 시간과 장소의 구애를 받지 않고 필요한 정보를 즉시 수정, 삭제, 생성, 조회가 가능하다.

한편 또 다른 모바일 기기로서 기존에 연구되었던 PDA 기반의 전자교무수첩은 유용성과 대중성의 면에서 휴대폰 단말기와 비교해 볼 때 개선이 필요하다. 또한 PDA용 플랫폼들도 다양하여 개발자들의 전문성이 요구되고 있는 것이 현실이다. 이에 본 연구에서는 PDA용 전자교무수첩시스템을 분석하여 PDA용보다 더 실용적이고 개선된 휴대전화 단말기용 전자교무수첩 시스템을 제안하게 되었다. 이를 위해 휴대전화 단말기용 플랫폼의 연구를 선행하게 되었고 그 연구 결과 현재 한국 무선 인터넷의 표준 플랫폼인 WIPI를 기반으로 한 전자교무수첩을 설계하고 구현하였다. WIPI 플랫폼 응용프로그램인 전자교무수첩은 Java 모바일 언어인 Jlet으로 구현하였다.

따라서 본 연구는 기존 사용하던 개인(교사 또는 교육관련 종사자)의 휴대전화 단말기에 그대로 전자교무수첩 시스템을 다운받아 사용하도록 함으로써 교사들이 보다 친근하게 사용할 수 있도록 일관된 사용자 인터페이스 환경을 제공하는 시스템이다.

목 차

논문개요

I. 서론	1
II. 관련연구	3
1. 기존 전자교무시스템 연구	3
1.1 Palm OS 기반 전자교무수첩	4
1.2 Windows CE 기반 전자교무수첩	5
1.3 Embedded Linux 기반 전자교무수첩	5
2. 무선인터넷 표준 플랫폼 WIPI	6
2.1 무선인터넷의 역사	7
2.2 WIPI의 정의 및 탄생배경	9
2.3 WIPI의 구조 및 특징	13
2.4 최근 WIPI의 동향	15
III. 전자교무수첩 시스템의 설계	18
1. 시스템 개발 배경	18
2. 시스템 개발 흐름	18
3. 시스템 설계	22
IV. 구현	26
1. 시스템 구현 환경	26
2. 시스템 구현	28
2.1 학급경영	30

2.2 교무일지	31
2.3 학생일람	33
2.4 기타사항	34
3. WIPI 단말기 포팅 및 상용화	37
V. 결론 및 제언	38

참고문헌

ABSTRACT

표 목 차

<표 1> 이동통신 3사의 다양한 플랫폼	10
<표 2> DMB 데이터 방송 서비스 유형	16
<표 3 > Desktop PC 사양	26
<표 4> 휴대폰 단말기 H/W 요구 사항(최소 권장 사양)	27

그림 목차

[그림 1] 기기별/플랫폼별 분류	4
[그림 2] WIPI 플랫폼 탄생 배경	11
[그림 3] WIPI 표준화 과정	12
[그림 4] WIPI 표준화 범위	12
[그림 5] WIPI 플랫폼 구조도	13
[그림 6] 시스템 개발 흐름도	19
[그림 7] COD서비스 구조	20
[그림 8] COD프레임 워크	21
[그림 9] 시스템 전체 구조	22
[그림 10] 학급경영 업무	23
[그림 11] 교무일지 업무	24
[그림 12] 학생일람 업무	25
[그림 13] 기타사항 업무	25
[그림 14] Jlet의 생명주기	28
[그림 15] 메인 화면	29
[그림 16] 메인 화면 소스 일부	29
[그림 17] 학급경영 화면	30
[그림 18] 교무일지 화면과 하위 메뉴	31
[그림 19] 교무일지 화면의 월간계획표 선택	32
[그림 20] 교무일지 메뉴 소스 일부	32
[그림 21] 학생일람 화면과 하위 메뉴	33
[그림 22] 기타사항 화면과 하위 메뉴	34
[그림 23] 교사주소록 입력 화면	35
[그림 24] 주소록 메뉴 소스 일부	36

I. 서론

최근의 컴퓨팅 환경은 무선 인터넷의 발달과 더불어 데스크탑에서 모바일로 그 무게 중심이 많은 부분 이전되어 활용되고 있다. 이러한 상황은 대부분의 사람들이 소유하고 있는 대표적 모바일 기기인 휴대전화 사용자가 폭발적으로 증가하고 그 이용 범위가 무궁무진하여 서비스가 다양해진 사실만으로도 알 수 있다. 단순히 통신 수단을 위해 개발된 휴대전화 단말기가 무선 인터넷의 발달로 다양한 서비스 개발이 되면서 휴대전화 단말기에 이제는 서비스되지 않는 것이 없을 정도이다[1].

본 연구는 지금까지 교사들의 교무업무와 교육 현장에서의 모든 일들에 있어 간편하고 이동성 있는 프로그램의 필요성을 발견하여 교사들이 교육 현장에서 즉시 활용할 수 있는 이동성을 강조한 모바일 휴대폰 단말기 활용을 제안하고자 한다.

휴대폰 단말기는 이미 일반화되어 있는 모바일 기기 중의 하나로서, 다양한 기능들을 간편하게 사용할 수 있으며 무선통신과 결합하여 차세대 모바일 환경에 적합한 정보 장치로 이미 그 위치가 확고하다. 또한 남녀노소 누구나 쉽게 사용할 수 있는 모바일 기기라는 장점이 있다. 이를 어떻게 활용하느냐에 따라 최고의 교육용 정보 장치로서도 이용이 가능하다.

한편, 휴대폰 단말기 개발환경을 위한 플랫폼은 실질적인 표준 없이 국내 서비스 사업자들이 응용프로그램 개발을 위한 자사의 플랫폼을 사용하고 있어서, 그로 인한 소모적 비용이 적지 않았다[2]. 본 논문에서 사용될 WIPI 플랫폼은 국내 무선 인터넷 표준화 포럼에서 국내 통신 사업자들의 모바일 표준 플랫폼에 대한 요구사항을 기본으로 탄생한 공통의 모바일 표준 플랫폼 규격이다[3].

이에 본 연구에서는 교사 업무보조를 위한 도구로 휴대폰 단말기를 제안함으로써 교사의 업무를 효율화시킬 수 있는 휴대폰 단말기용 소프트웨어를 개발하

는데 그 목적이 있다. 또한 설계에 있어 사용자 중심적인 인터페이스로 디자인 하여 실제 교육현장에서 활용 가능하도록 구현하였다. 이와 같은 본 연구의 목적을 세 가지로 나누어 제시하고자 한다.

첫째, 현재 국내 무선인터넷 모바일 플랫폼으로 표준화되어 있는 WIPI기반 휴대폰 단말기를 교사 업무 보조도구로 제안함으로써 교사 업무의 효율성을 극대화하고자 한다.

둘째, 학교 업무를 깊이 있게 파악하여 일반적인 개인정보관리가 아닌 교사를 위한 소프트웨어를 개발하여 교사들의 여러 불편사항들을 보안할 수 있도록 한다.

셋째, 전자교무수첩 시스템의 내용을 교무업무지원시스템의 입력자료로 활용하여 교사들의 자료 입력의 중복을 최소화하도록 설계, 구현하고자 한다.

본 논문의 구성은 다음과 같이 진행된다. 먼저 2장은 관련연구로서 본 논문을 연구하게 된 배경이 되는 기존 전자교무수첩 시스템의 분석결과이다. 아울러 본 연구에서 이용할 플랫폼인 WIPI에 대하여 구체적으로 살펴본다. 3장에서는 WIPI 기반 휴대폰 단말기의 설계과정을 진술하고 4장에서 구현과정을 기술한다. 마지막 5장에서는 결론을 내리고 향후 과제를 제시하겠다.

II. 관련연구

1. 기존 전자교무시스템 연구

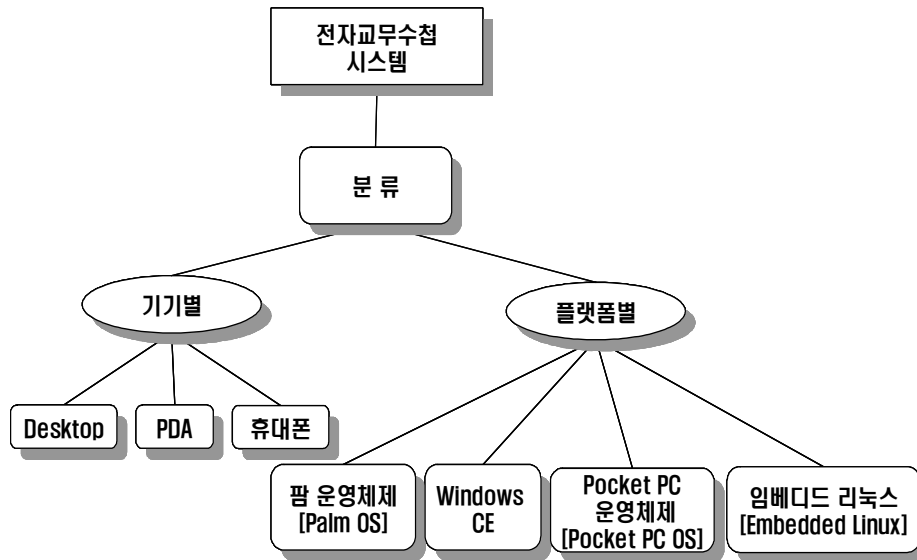
기존 전자교무시스템은 크게 데스크탑용과 모바일용으로 구분할 수 있다. 데스크탑용 시스템은 휴대성이 없고, 유선으로 연결된 장비와 장소가 필요하며 디자인이 다양하지 않고 소형화할 수 없다는 단점이 있다.

반면, 모바일용은 데스크탑용과 비교하여 많은 장점이 존재한다. 첫째, 휴대성과 이동성이 있어 언제 어디서나 정보의 관리가 가능하다. 둘째, 시간과 공간의 제약을 받지 않고 이용 범위를 확장할 수 있다. 셋째, 데스크탑용과 모바일용을 비교할 때 모바일용은 개인성과 편리성이 데스크탑에 비하여 단연 돋보인다.

기존의 전자교무수첩 시스템을 살펴보면 [그림 1]에서와 같이 플랫폼별과 기기별로 분류된다. 기기별로는 Desktop, PDA, 휴대폰 등으로 분류할 수 있으며, 플랫폼별로는 Palm OS, Windows CE, Pocket PC OS, Embedded Linux 등으로 분류할 수 있다. 중시할 점은 이 플랫폼들은 모두 PDA용 운영체제라는 점이다. 결과적으로 기존 전자교무시스템은 모두 PDA용 운영체제에서만 연구되어 활용하도록 제안되어 있다.

이와 같이 PDA용 플랫폼에서만 연구되어온 점을 보완하여 본 연구에서는 휴대폰 단말기를 이용한 시스템을 연구하여 구현할 것이다. 휴대폰 단말기는 타 기기에 비해 대중성과 친밀감이 있고 교사들이 추가 비용 및 교육 없이 사용할 수 있기 때문이다. 플랫폼면에서는 WIPI를 사용할 것인데, 그 이유는 2005년 상반기에 정보통신부 주관으로 탑재가 의무화되어 앞으로는 WIPI 휴대폰이 일반화될 전망이다[4]. 이러한 WIPI 기반 휴대폰단말기용 전자교무수첩의 연구 배경으로 각각의 플랫폼별로 Palm OS 기반 전자교무수첩[5]과 Windows CE

기반 전자교무수첩[6] 그리고 Embedded Linux기반 전자교무수첩[7]에 대하여 상세히 살펴보고자 한다.



[그림 1] 기기별/플랫폼별 분류

1.1 Palm OS 기반 전자교무수첩

Palm OS는 주요 PDA 운영체제로 제품군을 이루고 있는 Embedded Linux, Windows CE와 함께 임베디드 운영체제 중의 하나이다. Palm OS는 PDA 분야에서 인지도가 높은 3Com의 Palm을 대부분 사용한다. 그리고 모토롤라의 드레곤볼 칩을 쓰는 독자적인 방식을 사용한다. Palm OS가 많은 사람들에게 인기를 누리고 있는 이유 중 하나는 수년 동안 업그레이드되면서 안정화된 Palm OS를 기반으로 하는 수천 종류의 다양한 소프트웨어를 제공한다는 데에 있다.

현재 Palm OS 기반으로 연구된 전자교무수첩 시스템은 Metrowerks 사의 CodeWarrior Tool을 이용하여 개발한 시스템이 있다[5]. 이 시스템 역시 교사업무 보조를 위하여 PDA용 소프트웨어를 연구하여 적용한 것이다. 하지만 실제로

교사들이 교육현장에서 사용되기에는 몇 가지 문제점이 존재한다. 그것은 첫째 기술지원 문제, 둘째 가격이 비싸다는 것, 셋째 문자인식이 자유롭지 못하다는 문제 등이다.

1.2 Windows CE 기반 전자교무수첩

Windows CE는 PDA용 OS를 말한다. 따라서 Windows CE 기반 전자교무수첩 시스템은 PDA 기기를 목적으로 연구된 시스템이다. Windows CE 기반 전자교무수첩 시스템은 Microsoft사에서 개발한 임베디드 시스템을 위한 플랫폼인 Windows CE를 플랫폼으로 하고 기기는 PDA를 사용한 시스템이다. Windows CE를 주로 PDA에 탑재해 온 Microsoft사는 멀티미디어 기능을 강화한 Windows CE 3.0의 발표를 계기로 임베디드 운영체제 시장에서 높은 점유율을 차지한다. Windows CE의 장점은 Windows 및 Windows 응용프로그램의 사용 친숙도가 높고, 데스크탑 Windows와 응용프로그램 및 데이터의 호환성이 우수하며, 뛰어난 멀티미디어 처리 기능이 있으며, 응용 프로그램 개발 환경 또한 잘 갖추어져 있다는 것이다. 그러나 Windows CE는 첫째 무엇보다 가격이 비싸며, 둘째 사용시 많은 메모리를 요구하고, 셋째 상대적으로 속도가 느리며, 넷째 자유로운 재구성이 불가능하다는 단점이 있다[6]. PDA를 교사업무 도구로 제안한 연구에서 전자교무수첩 시스템을 제안하고 있으나, PDA에 익숙하지 않은 교사들이 교육현장에서 PDA를 사용하기 위해서는 그에 따른 비용과 시간이 요구되며 또한, 교육현장의 모든 교사들에게 PDA를 이용하여 교사업무를 운영하도록 한다는 것은 현실적으로 한계가 있다.

1.3 Embedded Linux 기반 전자교무수첩

Linux는 Unix 계열의 공개 운영체제이다. 핀란드의 Linus Torvalds가 창시자인

Linux는 세계 각국의 많은 개발자들에 의해 계속 발전하고 있다. 상용 Linux는 '고성능 프로세서와 대용량의 메모리를 장착하고 있는 범용 컴퓨터 시스템용으로 개발된 Linux'로 정의되며 데스크탑 PC용 Linux, 서버용 Linux 등이 이 범주에 해당된다[7].

반면에 Embedded Linux는 '낮은 성능의 프로세서와 작은 크기의 메모리를 가진 내장형 시스템용으로 개발된 Linux'이다. 그러므로 Embedded Linux는 다음과 같은 조건이 맞아야 한다. 첫째, 임베디드 장치는 작은 크기의 메모리 밖에 장착할 수 없다는 제약으로 인하여 Linux 자체의 크기와 기능이 최소화, 경량화, 그리고 맞춤화 되어야 한다. 이 조건은 Embedded Linux가 가져야 하는 필수 조건이다. 둘째, 낮은 성능의 프로세서를 사용하는 제약을 극복하기 위하여 성능이 최적화 되어야 한다[8].

현재 Embedded Linux 환경에서 Application의 개발은 Gui Toolkit 선택이 매우 중요하다. 이유는 각각의 임베디드 기기마다 환경이 다르기 때문이다. 그러므로 그 기기에 맞는 GUI Toolkit을 선택해야 되는 것이다. Embedded GUI Toolkit으로는 Embedded Qt, Microwindows, MiniCUI, Tiny-x 등이 있다. 이와 같은 Embedded Linux 기반의 플랫폼에서 전자교무수첩을 개발하기 위해서는 Linux에 대한 이해가 필수적이다. 또한, 시스템 설정이 어려우며, GUI 환경이 기존 Windows에 적응된 사용자들에게 이질감을 주어, 응용프로그램 개발환경이 다른 OS환경보다 상대적으로 좋지 않다. 플랫폼면에서 뿐만 아니라 기기면에서도 PDA를 사용하여 전자교무수첩 시스템을 활성화하기에는 Windows CE에서 살펴본 것처럼 한계가 있다.

2. 무선인터넷 표준 플랫폼 WIPI

무선인터넷 표준 플랫폼 WIPI를 살펴보기에 앞서, 무선인터넷의 발전과정을 개략적으로 살펴보는 것이 WIPI를 이해하는데 도움이 될 수 있다.

2.1 무선인터넷의 역사

무선인터넷의 발전과정은 1세대와 2세대 무선인터넷 기술로 나뉘지며, 각각은 다음과 같다.

1세대 무선인터넷은 단말기에 내장된 브라우저를 이용하여 무선인터넷에 접속하는 기술을 말한다. PC에서 인터넷에 접속하기 위해서 넷스케이프나 익스플로러와 같은 웹 브라우저가 필요한 것과 같이 이동전화에서 무선인터넷을 사용하기 위해서도 전용 브라우저가 필요하다[9].

현재 가장 많이 보급되어 있는 브라우저는 WAP 브라우저, ME브라우저, I-mode 브라우저 등이 있다.

WAP(Wireless Application Protocol)은 97년 Ericsson, Motorola, Nokia, Unwired Planet 등 4개사가 처음으로 제안한 프로토콜로서 이동전화나 PDA등 소형 무선 단말기에서 무선 인터넷을 사용하기 위해 만든 프로토콜이다.

ME(Mobile Explorer)는 무선단말기를 통해 유선인터넷에 접속하기 위해서 마이크로소프트사가 만든 브라우저이다. ME는 이동단말기에 중점을 둔 WAP과는 달리 다양한 데이터 단말을 지원하기 위하여 개발되었다.

I-mode는 일본의 이동통신 사업자인 NTT Docomo가 세계 최초로 자체 개발한 것으로서, 1999년 2월에 상용서비스를 시작하였으며, 유선 인터넷에서 사용 중인 HTML이나 HTTP 등의 인터넷 표준을 기반으로 하여 독자적으로 개발한 기술언어와 프로토콜을 사용한다.

이러한 1세대 무선인터넷 기술은 사용자 인터페이스가 단순하고, 요금이 비싸며, 속도 및 보안에 문제점을 가지고 있다. 이를 해결하기 위해서 무선인터넷 플랫폼이라는 기술이 출현하게 되었는데, 무선인터넷 플랫폼이란 응용서비스를 실행하기 위한 기반 환경을 말한다.

이는 이식성(Portability)과 API(Application Programming Interface)의 추상화(Abstraction)를 통해서 모든 단말기에 동일한 API를 제공하므로 하드웨어에 대한

독립성을 보장한다[10]. 무선인터넷 플랫폼을 도입함에 따라 비연결성 지원을 통해 콘텐츠를 다운로드하여 단말기에 저장할 수 있게 되었다.

국내에서 사용되고 있는 무선인터넷 플랫폼의 종류는 GVM(General Virtual Machine), MAP(Mobile Application S/W Plug-in), SK-VM(SK-Virtual Machine), KVM(Kilobyte Virtual Machine), BREW(Binary Runtime Environment for Wireless) 등이 있다[11].

GVM은 신지소프트에서 개발한 순수 국내 기술로 제작하여 상용화된 최초의 플랫폼이다. 국내 기술로 제작되었기 때문에 라이선스비의 부담이 없는 것이 큰 장점이다. 사용언어는 자체 개발된 Mobile-C를 사용하였으며, 무선인터넷 게임에서 많이 활용되고 있다.

SK-VM은 SK텔레콤의 사내 벤처인 XCE에서 개발된 플랫폼으로서, 선 마이크로시스템즈의 소스코드를 사용하지 않고 J2ME(Java 2 Micro Edition) 스펙을 이용하여 독자적인 MIDP(Mobile Information Device Profile)기반의 SK-VM을 개발하였다. SK텔레콤에서는 자바 진영과 C 언어 진영간의 시장 흐름에 따라 대처하기 위해 SK-VM을 도입하여 '네이트'라는 명칭으로 서비스 중에 있다.

KVM은 선 마이크로시스템즈에서 개발한 미들웨어 플랫폼으로서, 자바언어를 사용하였다. 이는 자바 가상머신(Java Virtual Machine) 상에서 스크린폰(Screen Phone), PDA(Personal Device Assistant), 셋톱박스, 휴대폰 등에 탑재하기 위한 가용메모리가 128Kbyte 정도의 제품을 겨냥해 개발된 기술로서, 현재 LGT에서 서비스되고 있다. KVM은 다양한 디바이스 간의 기본적인 기능들을 수용할 수 있는 최소한의 자바 가상머신과 자바 API 구성을 제공하며, 16비트 또는 32비트 프로세서의 전체 메모리가 256K 정도인 디바이스를 대상으로 하지만 디바이스의 기능에 따라 유동적으로 적용 가능하다.

MAP은 모빌탑(Mobile Top)에서 C언어를 사용하여 개발하였고, 바이너리 다운로드 방식을 채택한 최초의 무선인터넷 플랫폼이다. 현재의 플랫폼 중에 가장 적은 메모리를 사용하고 있으며, 플러그 인(Plug-in) 방식으로 단말기에 게임이나

응용프로그램을 다운로드 받아 구동하는 방식을 취하고 있다. KTF에서 2001년 3월부터 ‘매직[®]’ 서비스로 제공되고 있다. MAP은 단말기 내 펌웨어(firmware), 무선인터넷을 이용하는 서버, 그리고 이에 적용되는 프로그래밍 규격과 소프트웨어 개발 키트(SDK)를 제공하고 있다[12].

BREW는 킷컴에서 개방형인 C언어를 사용하여 개발한 플랫폼으로서, 바이너리 다운로드 방식을 채택하였으며, ME(Mobile Explore)브라우저 및 자바 가상 머신(Java Virtual Machine) 및 다른 언어로 구현된 애플리케이션과도 연동이 가능한 구조를 가지고 있다. BREW는 다운로드 및 실행에 있어 자바를 사용하는 GVM, KVM보다 빠른 속도로 실행되는 장점을 가지고 있다. KTF에서 2001년 9월부터 컬러단말기에 BREW를 탑재하여 서비스 중이며, 무선인터넷 플랫폼 솔루션 중에서 가장 강력한 실행기능을 내장하고 있다.

2.2 WIPI의 정의 및 탄생배경

WIPI는 영어 ‘Wireless Internet Platform for Interoperability’의 머리글자를 딴 것으로 휴대전화 단말기에 내장되어 무선 인터넷 서비스 기능을 지원하는 플랫폼 규격을 지칭하며, 일종의 휴대폰 단말기용 OS(Operating System)로 볼 수 있다[13].

2.1에서 살펴본 바와 같이 국내에서 서비스되고 있는 무선인터넷 플랫폼은 GVM, MAP, SK-VM, KVM, BREW 등의 5가지로 요약될 있으며 각각의 플랫폼은 독자적인 기술을 이용하여 구현되어 있다. <표 2>는 이동통신 사업자들이 서로 다른 무선 응용 플랫폼을 사용하고 있는 것을 나타낸다.

<표 1> 이동통신 3사의 다양한 플랫폼

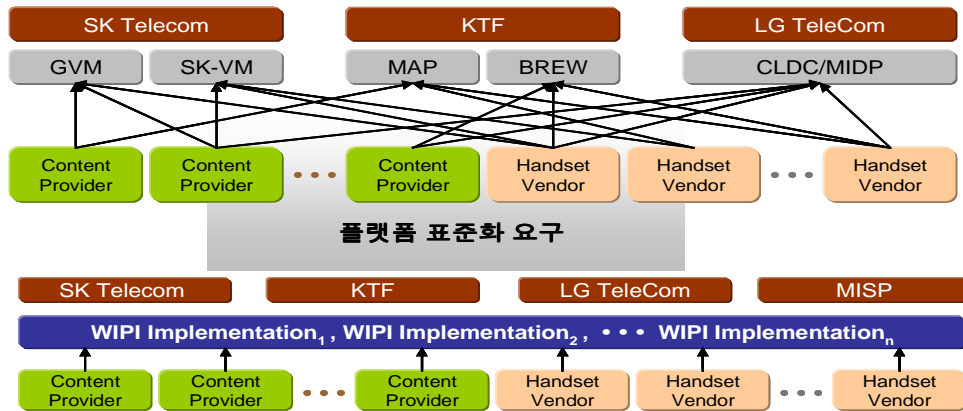
추진사	플랫폼	개발언어	수행방식
LG(썬)	KVM	자바	인터프리터(VM)
LG(아로마소프트)	Kitty Hawk	자바	인터프리터(VM)
SK텔레콤(XCE)	SK-VM	자바	인터프리터(VM)
SK텔레콤(신지소프트)	GVM	C/C++	인터프리터(VM)
SK텔레콤	WITOP	자바, C/C++	인터프리터(VM)
KTF(모빌탑)	MAP	C/C++	바이너리(네이티브)
KTF(퀄컴)	BREW	C/C++	바이너리(네이티브)
KWISF(이통3사),TTA	WIPI	자바, C/C++	바이너리,컴파일러

이 같은 모바일 디바이스의 증가에도 불구하고 모바일 환경용 플랫폼에 대한 실질적인 표준 없이 각 서비스 사업자들이 응용프로그램 개발을 위한 자사의 플랫폼을 사용하고 있어서, 그로 인한 소모적 비용이 적지 않았다[1][14]. 휴대전화 단말기의 공통적인 기능만을 활용하는 프로그램을 개발, 공급하고 싶어도 모바일 플랫폼의 표준화 노력 부족으로 무선망 사업자, 단말기 제조업자, 콘텐츠 제공업자 모두 모바일 플랫폼의 잠재성을 개발하고 활용하는 데 많은 어려움을 겪어온 것이 사실이다. 또한, 국내의 각종 휴대전화단말기 제조업체 및 사업자별로 플랫폼이 상이하여 응용프로그램 제작에 과다한 비용이 소모되고 있었으며, 사용자들은 제한된 서비스를 제공받는 문제점이 있었다[14].

이러한 요구에 부응하여 국내 무선 인터넷 표준화 포럼에서 국내 통신사업자들의 모바일 표준 플랫폼에 대한 요구사항을 기본으로 하여 공통의 모바일 표준 플랫폼(Common Mobile Platform) 개발을 위한 표준 규격을 정의하게 되고 이 요구에 부응하여 탄생된 것이 WIPI이다[15][16].

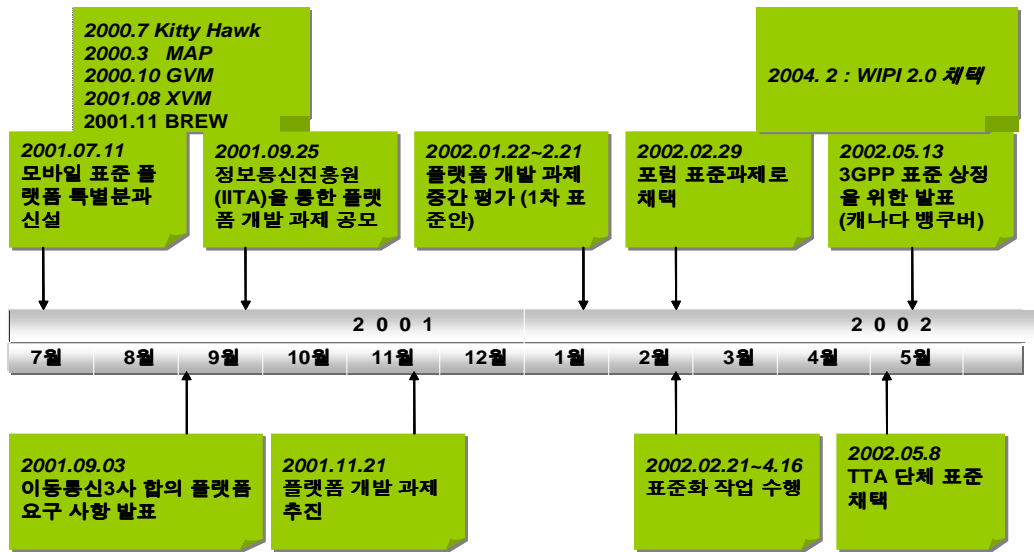
WIPI의 탄생은 다음과 같은 요구들을 만족시키고자 했다. 첫째 이동통신사 관점에서는 다양한 콘텐츠 공급과 폭넓은 사용자 층의 확대를 위한 기반 조성이

필요하고, 고품질의 서비스를 통한 고객 만족 실현이 이루어져야 하는 점이다. 둘째로 단말기 제조사 관점에서는 엔지니어링 시간과 비용의 최소화 그리고 빠른 단말기 개발 공정으로 원활한 단말기 공급 가능성이 절실하였다. 셋째로 Content 및 응용 프로그램 개발사 관점에서는 콘텐츠 개발 시간 및 비용의 최소화가 관건이며 1회의 콘텐츠 개발로 모든 이동통신 사업자에 공급이 가능해야 하였다. 다음 [그림 2]는 이러한 요구들 즉, 응용 프로그램 실행 환경들이 혼재된 복잡한 서비스 형태에서 플랫폼의 표준화가 요구되어 WIPI가 탄생되는 배경을 나타낸 것이다.



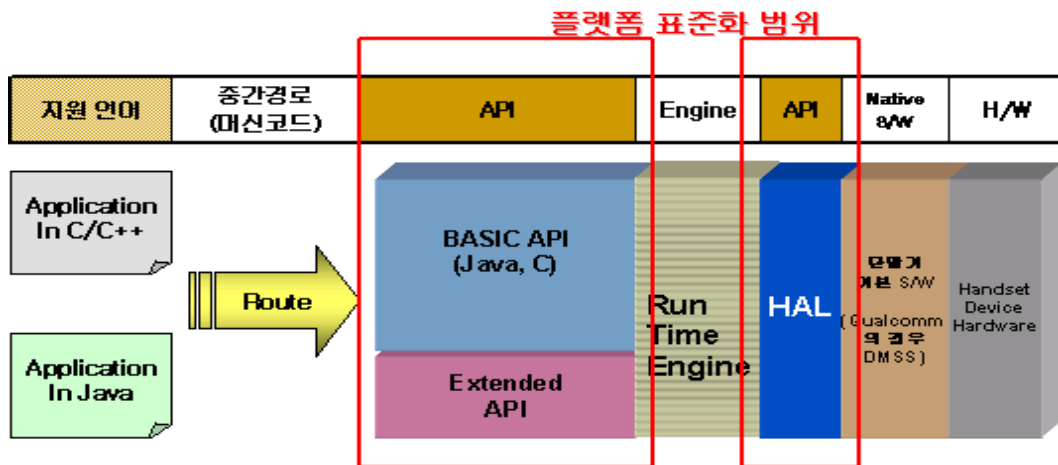
[그림 2] WIPI 플랫폼 탄생 배경

이처럼 WIPI는 기존 플랫폼의 단점들을 보완하고, 차세대 서비스 기술을 반영하기 위하여 한국무선인터넷 표준화 포럼(KWISF : Korea Wireless Internet standardization Forum)[3]과 한국전자통신연구원(ETRI : Electronics and Telecommunications Research Institute)[15]의 활동에 의해 개발된 이동통신 단말기용 응용 프로그램 실행 환경을 표준화한 규격이다. 이후, 2002년 5월 7일 한국정보통신기술협회(TTA : Telecommunication Technology Association)를 통해 단체 표준으로 채택되었다[16]. 다음의 [그림 3]은 WIPI의 표준화 과정을 나타낸 것이다.



[그림 3] WIPI 표준화 과정

[그림 4]에서는 WIPI의 표준화 범위를 나타내었다. WIPI의 표준규격의 의미는 그림에서 나타낸 것과 같이 플랫폼 전체 구조에 대한 규격화를 의미하는 것이 아니라 콘텐츠의 호환을 유지하기 위한 최소한의 API 세트(HAL 포함)만을 의미한다.

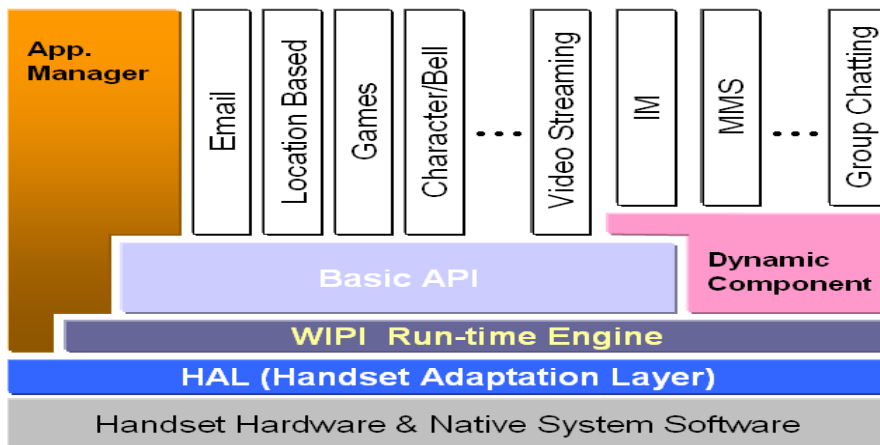


[그림 4] WIPI 표준화 범위

2.3 WIPI의 구조 및 특징

2.2절에서 살펴본 것처럼 WIPI 규격은 이동 통신 단말기에 탑재되어 응용 프로그램을 수행할 수 있는 환경을 제공하는 모바일 표준 플랫폼 규격이다.

이 규격을 만족하는 모바일 플랫폼은 단말기용 응용 프로그램 개발자에게는 플랫폼 간 콘텐츠의 호환성을 보장하고, 단말기 개발자에게는 플랫폼 이식의 용이성을 제공하며, 일반 이용자에게는 다양하고 풍부한 콘텐츠 제공을 목적으로 한다. WIPI는 앞서 살펴본 BREW나 GVM, SK-VM 등과 는 달리 플랫폼 자체가 아닌 플랫폼을 구성하기 위한 규약이다. 이 말은 WIPI의 규격을 따르는 다양한 플랫폼들이 만들어 질 수 있다는 뜻이다. 그러므로 WIPI 플랫폼의 보다 더 상세한 구조와 특징을 살펴볼 필요가 있다.



[그림 5] WIPI 플랫폼 구조도

[그림 5]는 WIPI 플랫폼 구조도를 나타내고 있다. App. Manager는 응용프로그램의 Download, 설치, 삭제 등의 응용프로그램을 관리하며, API 및 Component들의 추가/갱신의 역할을 한다. 본 논문에서 구현되는 시스템은 App. Manager 부분에서 실행된다. WIPI의 구조를 좀 더 상세히 살펴보자.

·API Manager : 응용프로그램의 Download, 설치, 삭제, 등의 응용프로그램 관리를 담당하는 부분이다. 또한, API 및 Component들의 추가와 갱신을 할 수 있으며, Dynamic Linking Library로서 WIPI의 일부가 된다.

·Basic API : 응용 프로그램에서 개발자를 위한 기본 API로서 C와 Java언어를 모두 지원한다.

·Dynamic Component : App. Manager를 통하여 추가/갱신된 API 및 Component들을 나타낸다.

·WIPI Run-time Engine : 응용프로그램 개발자를 위한 API로서 C와 Java 언어를 모두 지원한다.

·HAL(Handset Adaptation Layer) : 단말기 제조사를 위한 규격으로 플랫폼에서 단말기 하드웨어 및 OS와 인터페이스 하는 부분에 대한 규격이다.

·Handset Hardware & Native System Software : 플랫폼의 하드웨어 독립성을 유지하기 위한 추상화 계층으로 상위 Layer들은 HAL위에서 Native System과 무관하게 동작하도록 지원한다.

자바를 이용한 WIPI 애플리케이션에서는 AOTC(Ahead Of Time Compiler)라는 기술을 사용한다. 기존에 사용했던 핫스팟(Hotspot)이라는 수행중의 코드의 어느 부분이 자주 사용되며 병목현상이 나타나는지 분석 후 필요한 부분만을 네이티브로 컴파일 하는 기술이 있었으나 메모리 사용량이 많아 모바일에서는 부적합하였다[17].

자바로 프로그램을 만들고 컴파일하려면 바이트 코드를 생성하고 각 휴대폰의 CPU에서 돌아가는 네이티브 바이너리로 다시 컴파일 한다. AOTC 기술은 자바의 간결성과 안전성 등의 장점과 C의 빠른 속도를 충족해 주는 기술이다[17]. 그러면 WIPI의 특징을 구체적으로 살펴보자.

- 복수 개발 언어 지원 : C/C++, Java™
- 모든 Content가 binary code로 다운로드 되어 실행
- PC처럼 다중 응용프로그램을 독립적으로 동시에 수행
- 플랫폼 보안 (일반수준, CP수준, 시스템 수준)
- API 보안 (NO_ACCESS, READ_ONLY, WRITE_ONLY, READ_WRITE) : 이동통신사에 의해 플랫폼 이식시점에서 결정
- 디렉터리 보안 (Private, Application Shared, System Shared) : 개인 디렉터리, 응용프로그램 공유 디렉터리, 시스템 공유 디렉터리
- 서비스 보안 : 신뢰할 수 있는 기관(TTA)으로부터의 서명

2.4 최근 WIPI의 동향

무선인터넷 표준 플랫폼인 WIPI 규격이 2.0버전까지 완료된 가운데 한국전자통신연구원(ETRI)과 한국무선인터넷표준화포럼(KWISF)에서는 WIPI의 장기적인 발전방향(Roadmap)을 다시 수립하려 하고 있다[18]. 이 과정에서 WIPI 규격에 다양해지는 휴대폰의 각종 응용 서비스(Application)까지 포함시킬 것인지 아니면 최소한의 표준 규격만 제정하고 나머지는 시장의 자율경쟁에 맡겨야 하는지의 의견을 조정하고 있는 상태이다. 이러한 가운데 중시할 점은 최근 각광을 받는 기술인 DMB 기술과 WIPI와의 접목이다[19].

DMB(Digital Multimedia Broadcasting)가 기존의 타 방송 매체와 구별되는 이유는 통신·방송 융합형 매체라는 데 있으며, 이러한 DMB의 매체적 특성을 단적으로 규정지을 수 있는 서비스 특징이 데이터 방송 서비스이다. DMB는 데이터 방송 서비스가 기존 여타의 매체와의 차별화를 결정하는 주요 요인이 된다. DMB 데이터 방송은 유럽 등지에서 DAB(Digital Audio Broadcasting)의 부가 서비스로 시도되고 있는 r-Commerce 또는 Visual Radio 등에서 그 유래를 찾

을 수 있다. 기본적으로 오디오에 치중한 해외의 DAB기반 데이터 방송 서비스에 반해, 국내의 DMB 데이터 방송 서비스는 동영상과 멀티미디어 데이터 서비스를 강조하고 있는 DMB를 근간으로 한다는 측면에서 보다 본격적이고 진보된 데이터 방송 서비스라 할 수 있다. IP기반의 리턴채널을 적용한 보다 응용화된 데이터 방송 서비스 모델 개발이 가능하도록 DMB 데이터 방송 서비스 플랫폼 자체를 휴대인터넷 규격인 WIPI(Wireless Internet Platform for Interoperability)로 도입하는 것을 적극적으로 검토하고 있다[19]. <표 2>에서는 DMB 데이터 방송 서비스 유형을 구체적으로 보여주고 있다.

<표 2> DMB 데이터 방송 서비스 유형

서비스 유형	정의 내용
Visual Radio	비디오 방송 프로그램의 전송 오디오 방송의 경우 DBM 오디오 채널로 방송
r-Commerce	방송되는 음악의 CD 라벨을 이미지로 전송 청취중인 음악의 CD 등을 DMB 단말에서 직접 구매
MEMO(Multimedia Environment for Mobiles)	DMB에서의 양방향성 기능을 부여하기 위한 프로젝트 유럽에서는 리턴채널로 GSM을 사용하여 이용자의 피드백을 전송

한편, 최근 쟁점화되고 있는 미들웨어의 표준안들 중에서 국내 휴대인터넷 무선 플랫폼으로 제안된 WIPI와 유럽의 멀티미디어 데이터 방송 규격인 DVB-MHP(Digital Video Broadcasting-Multimedia Home Platform)가 있다. WIPI 도입에 의한 기대효과는 먼저 기존 이동통신 사업자들간의 이질적인 무선인터넷 플랫폼을 단일 표준안으로 하여 중복 투자를 방지할 수 있다는 점이다. 즉, 이종(異種) 서비스 플랫폼은 서로 상이한 서비스 환경에서 구현이 가능한 애플리케이션과 콘텐츠를 각각 요구하는 반면, WIPI는 통합된 무선인터넷 서비스 플랫폼 환

경을 구축하여, 중복 투자의 폐해를 방지 가능하고 콘텐츠 및 애플리케이션 공유를 촉진할 수 있다는 이점이 있다.

또한, 국내 독자 표준안 채택을 통한 해외 로열티 유출액 감소 효과도 WIPI에 의한 기대 사항이다. 퀄컴사의 BREW를 기반으로 한 무선 인터넷 단말기의 경우 단말기 1대당 약 3달러가 로열티로 지출되는 반면, WIPI의 경우 선마이크로시스템사의 자바 기술 사용에 대한 대가로 1대당 약 20센트만 지불하면 된다[20]. 이러한 측면에서 한국정보통신기술협회(TTA) 내 DMB PG 일각에서는 DMB 데이터 방송 서비스 규격으로 WIPI를 적극 고려하고 있다.

정부와 업계는 이동통신 단말기에 있어 퀄컴에 대한 기술료 지불사례를 통해 이미 DMB 데이터 방송에 있어, 국내 독자 표준안 개발 및 국제 표준화의 중요성에 대한 충분한 공감을 하고 있다. 일례로 무선 인터넷 플랫폼 규격으로 국내에서 독자적으로 개발한 WIPI를 채택한 것은 이러한 교훈에서 비롯된 조치라 할 수 있을 것이다[20].

현재 DMB 미들웨어 표준안 선정을 두고 벌어지는 논쟁의 핵심 쟁점사안은 바로 이 독자 표준 문제에서 비롯된다. 정부 측 국내 DMB 기술규격 제정 작업을 전담하고 있는 정보통신기술협회(TTA) DMB PG에서는 WIPI를 서비스 플랫폼으로 적용하는 방안을 적극 검토 중에 있는데, 이는 국내 독자 표준의 사용을 통한 해외 기술료 지출 감소 효과 및 한국 주도의 국제 표준화 효과, 나아가 DMB 도입의 경제적 효과 극대화 등을 기대한 것에서 비롯된 판단으로 보인다.

Ⅲ. 전자교무수첩 시스템 설계

본 장에서는 WIPI플랫폼에 교사업무 보조를 위한 WIPI 휴대폰용 애플리케이션 시스템의 개발 배경과 설계 과정에 대하여 서술하고자 한다.

1. 시스템 개발 배경

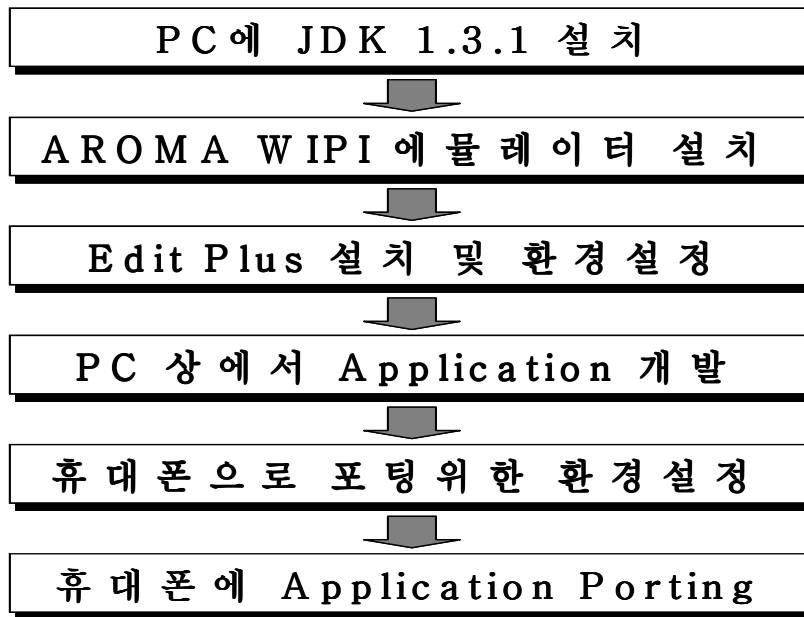
교육정보화에 따른 컴퓨터의 보급과 함께 많은 교사용 업무지원 프로그램이 등장하여 활용되고 있다. 그럼에도 불구하고 교사의 업무에 능률이 오르지 않고, 컴퓨터 사용에 불편함을 느낀다면 이것은 오히려 정보화 추진계획의 역효과라 할 수 있다.

본 연구에서 교사업무보조도구로 휴대폰을 설정한 이유는 서론에서 밝힌 것처럼 이미 일반화되어 있어 어디서나 휴대가 가능해서 이동 중에도 정보 입력이 가능한 편리함을 제공할 뿐만 아니라, 다양한 장치가 있으며, 활용용도에 따라 얼마든지 서비스를 응용할 수 있기 때문이다.

또한, 타 플랫폼에 비해서 유연성과 호환성이 장점이며, 이 특징으로 인해 Java 언어와 C언어를 사용하여 WIPI로 전환하기가 용이하다.

2. 시스템 개발 흐름

WIPI 기반 전자교무수첩 애플리케이션의 전체적인 개발 흐름은 [그림 6]과 같다.



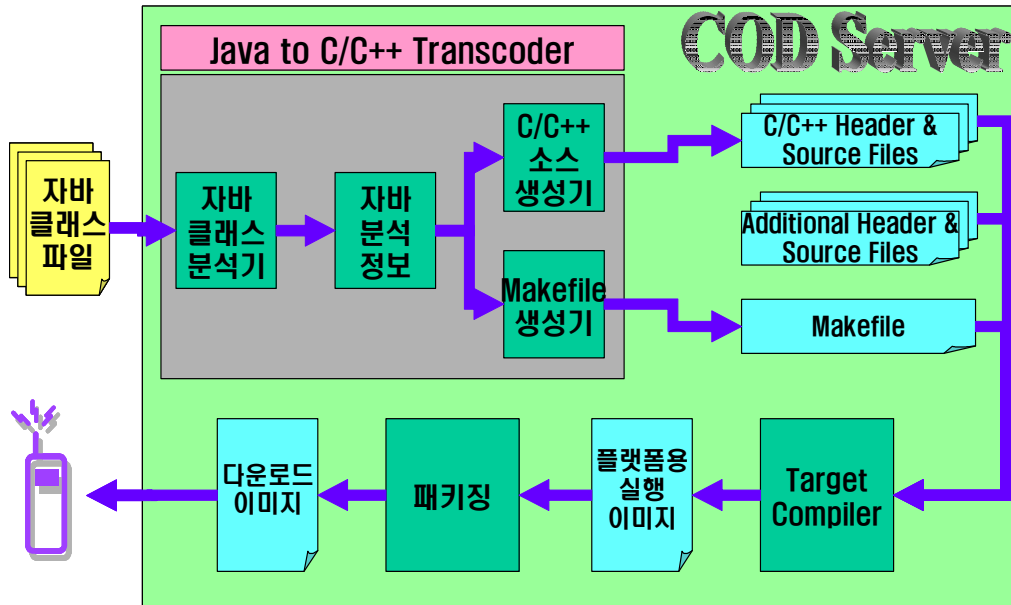
[그림 6] 시스템 개발 흐름도

우선 PC에서 프로그램 개발환경을 갖추기 위해 JDK와 WIPI 에뮬레이터의 설치가 필요하다. WIPI 에뮬레이터 설치 후 개발에 필요한 Editor를 프로그램이 가능한 환경으로 설정을 마친 후, 직접적인 코딩 작업에 들어간다. 모든 코딩을 마쳤다면 에뮬레이터에서 실행시키기 위해 jar 파일로 변환하는 작업이 필요하다. 이 jar 파일은 실제 핸드폰에서 다운로드 하여 테스트하기 위해 바이너리 파일로 만들어야만 한다.

현재 이동통신사 중에서는 KTF에서 테스트를 지원해주고 있다. 이를 위해서는 개발자 지원 사이트에서 작성한 파일을 올려서 바이너리 파일을 만들고, 테스트 폰으로 다운로드 받을 수 있게 되어 있다. 개발자 권한으로는 컴파일과 바이너리 파일만을 만들 수 있고, CP(Content Provider) 회사 소속으로 등록이 되어야 폰으로 다운로드 받을 수 있다.

이 같은 WIPI 응용프로그램 개발 과정을 이해하기 위해서는 COD의 개념이해가 필요하므로 COD에 대해 간단히 살펴보겠다. [그림 7]에서 보는 바와 같이

WIPI 응용프로그램을 작성한 자바 클래스 파일을 Makefile을 생성하여 COD Server를 거치는 COD 서비스 구조를 보여준다.



[그림 7] COD서비스 구조

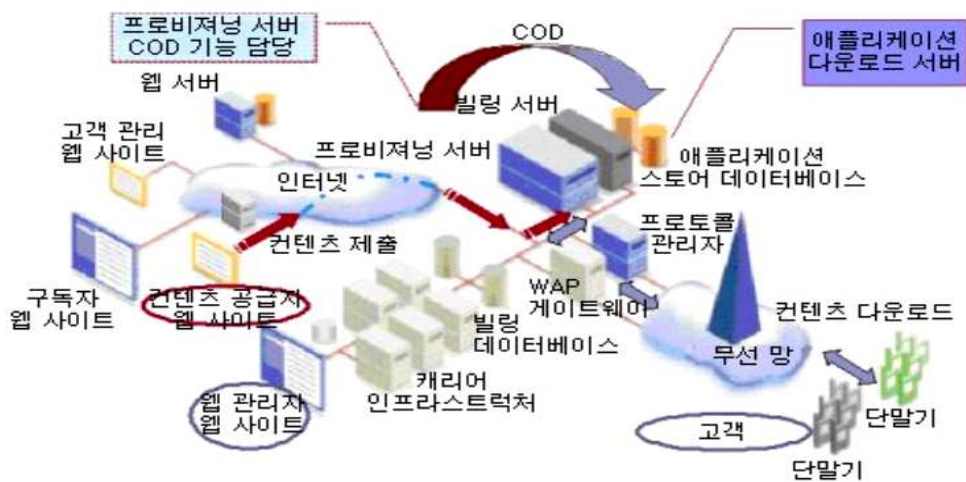
COD는 Compile On Demand의 약자로서 WIPI 규격으로 개발된 코드를 단말기의 Native 코드로 변환하는 기술이다. COD는 VM기술과 Native 바이너리 기술의 장점을 흡수한 기술이다.

VM(Virtual Machine) 기술은 응용프로그램과 하드웨어 사이에 가상의 계층을 두어서 VM 규격을 지키는 하드웨어와 코드끼리의 동작을 보증하게 된다. 이런 동작이 가능한 것은 응용프로그램과 하드웨어 간의 중간 코드가 번역을 수행하기 때문이다. 문제는 이러한 번역과정은 성능의 감소를 불러온다는 것이다 [21].

Native 바이너리 기술은 응용 프로그램의 코드가 기계어로 번역이 되고 기계어는 하드웨어를 직접 제어하게 된다. VM 기술과 달리 중간의 번역 과정이 없

어서 성능이 향상되지만 응용 프로그램의 코드가 하드웨어에 종속되는 결과를 초래한다. COD는 AOTC 기술이 포함된 일종의 프레임 워크로 응용 프로그램 코드의 검증과 변환, 그리고 컴파일과 패키징이 자동으로 수행된다. AOTC는 Ahead Of Time Compiler의 약자로서 Java 언어로 작성된 WIPI 규격에 맞춰서 응용 프로그램을 개발하고 COD 서버에 jar 파일을 올리면 코드를 검증, 변환, 컴파일하고 다운로드 가능한 패키지를 생성하게 된다[21].

[그림 8]은 이러한 COD 프레임 워크의 동작 과정을 보여주고 있다.



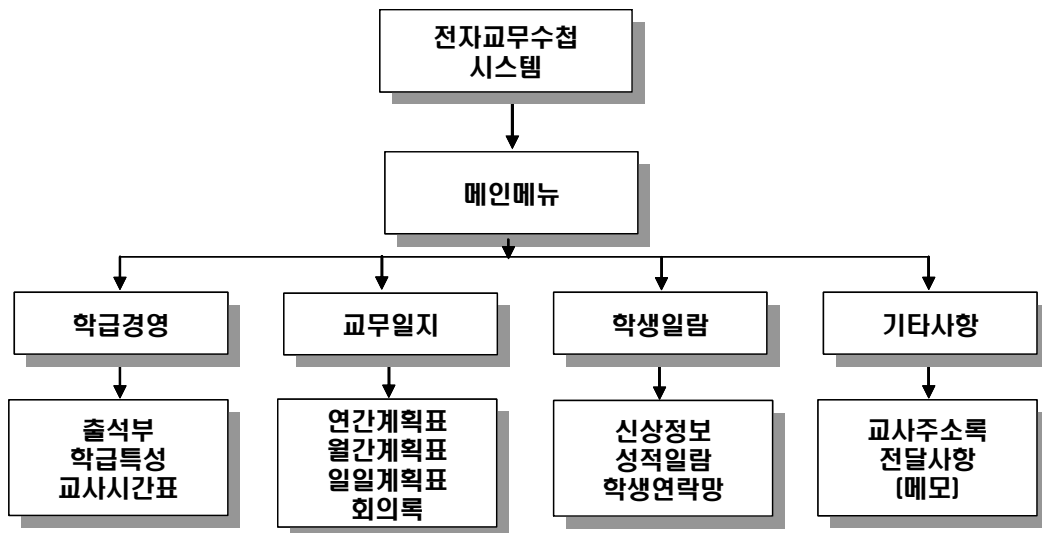
[그림 8] COD프레임 워크

따라서 COD 기술은 WIPI 규격을 통한 개발로 VM의 범용성을 가지며 AOTC 기술을 이용하여 성능을 끌어올릴 수 있는 장점을 갖는다.

3. 시스템 설계

시스템 설계는 여러 교사들의 업무 중 가장 작은 단위인 수첩기능을 활용한 휴대폰용 소프트웨어인 전자교무수첩을 구성하는 작업의 단위를 설계하고자 한다.

전자교무수첩은 학급경영계획, 교무일지, 학생실태에 관한 사항, 기타 사항으로 구성되고 각각은 그에 따른 하위메뉴로 나누어진다. 이러한 설계를 위한 시스템 전체 구조도는 [그림 9]와 같다.



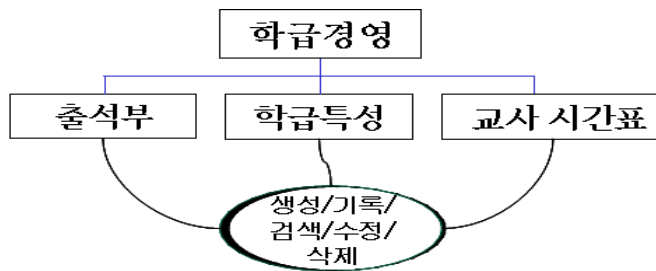
[그림 9] 시스템 전체 구조

본 연구는 교사업무수첩과 관련된 4가지의 응용소프트웨어(학급경영, 교무일지, 학생일람, 기타사항)를 설계하여 구현하고자 한다. 이 기능들은 주 사용자인 교사가 휴대폰에서 응용프로그램을 다운받아 사용할 때 최대한 간편하게 사용되도록 편리성에 주안점을 두고 설계가 이루어졌다. 각각의 기능에 따른 세부 화면

의 구조를 살펴보면 다음과 같다.

1) 학급경영 화면

[그림 10]은 학급경영 메뉴의 하위메뉴를 보여주고 있다. 먼저 출석부 메뉴를 통하여 교사는 담임을 맡고 있는 학급의 출석부를 관리한다. 그리고 교사시간표 메뉴에서는 교사 자신의 교과 시간표를 기록하고 검색, 수정한다. 학급특성 메뉴에서는 담임으로서 꼭 알아야 할 그 학급의 특성들을 기록하여 관리하게 된다. 출석부 메뉴를 이용하여 교사는 매일 교실에 있는 학생들의 출결상황을 확인해야 하며 교무실에서도 학생의 출결상황은 교무업무와 관련하여 교사가 자주 필요로 하는 내용이다. 출석률이 높은 학교에서는 문제가 되지 않지만, 비교적 조퇴·지각·결석이 많은 학교에서의 출석부 관리는 매우 번거로운 일이다. 따라서 이런 어려움을 최소화하기 위하여 출석부 폼을 개발하여 휴대폰에서 어느 곳에서나 변동사항 기록과 수정이 가능하도록 설계하였다.

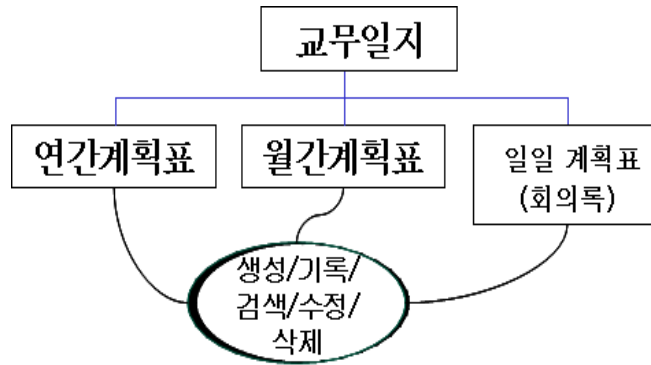


[그림 10] 학급경영 업무

2) 교무일지 화면

교무일지 메뉴는 교사의 교무업무에 대한 연간 계획, 월간 계획, 일일 계획을 기록할 수 있고 조회와 수정이 가능하도록 설계된 메뉴이다. 교사들이 회의 시간의 내용을 기록하거나 전달 사항을 기록하여 필요시에 특정한 날의 회의 내용이

나, 그에 대한 일정 그리고 회의에서의 전달사항을 확인할 수 있다. 검색은 내용이 등록된 날로 검색이 이루어지며 일간/주간/월간 별로 검색할 수 있도록 화면을 설계하였다. [그림 11]은 이러한 교무일지 업무의 메뉴 구조를 보여주고 있다.

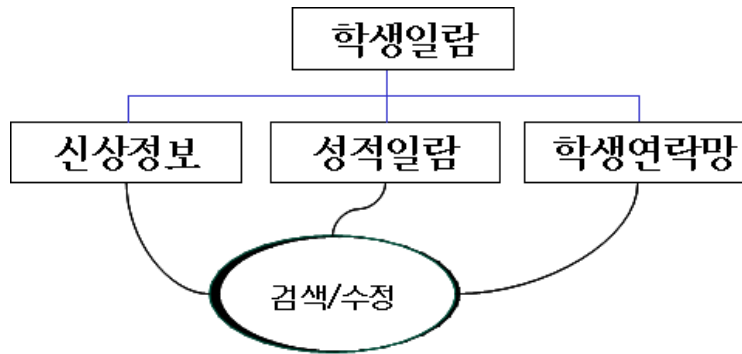


[그림 11] 교무일지 업무

3) 학생일람 화면

학생 일람 화면에서는 학생의 기본 신상정보를 조회, 검색할 수 있고, 변동사항이 발생하면 그 즉시 어느 곳에서도 수정이 가능하도록 설계하였다.

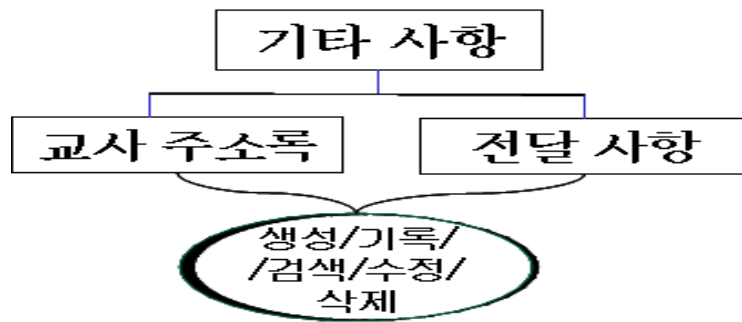
학생 신상정보는 학생의 정보와 그 학생의 보호자, 교우관계, 진로상담을 기록할 수 있다. 그리고 성적일람이라는 메뉴를 이용하여 교사는 학생 개개인의 성적을 검색하고 간단히 수정할 수 있다. 학생 비상연락망 메뉴에서는 학급의 학생들 연락처들을 저장하여 검색할 수 있도록 하였다. [그림 12]는 이러한 학생일람 업무의 구조를 설계한 것이다.



[그림 12] 학생일람 업무

4) 기타 사항 화면

[그림 13]은 기타사항 업무의 구조를 나타낸다. 기타사항 메뉴에서는 교사가 다른 교사들의 비상연락망을 입력, 조회할 수 있도록 교사 주소록 메뉴를 만들었다. 또한, 전달사항 메뉴에서는 교사가 어느 곳에서든지 학생들에게 전달하고자 하는 일이 발생하거나 아이디어가 떠오를 때 즉시 입력할 수 있도록 하였다.



[그림 13] 기타사항 업무

IV. 구현

본 장에서는 전자교무수첩 시스템의 구현 환경과 구현 과정에 대하여 서술하고자 한다.

1. 시스템 구현 환경

본 연구의 시스템 구현을 위한 데스크탑 개발 환경은 크게 하드웨어환경과 소프트웨어 환경으로 나눌 수 있다. <표 3>은 Desktop PC 사양을 나타낸 것이다. 먼저, 하드웨어 환경은 CPU Pentium IV 2.8GHz(Memory 512)이며, Windows-XP Professional의 운영체제 환경에서 진행되었다. 소프트웨어 환경으로는 아로마 WIPI 에뮬레이터를 이용하였다. 또는 KTF WIPI 에뮬레이터를 사용하여도 같은 환경으로 진행할 수 있다. 에뮬레이터를 이용한 이유는 Desktop 컴퓨터에서 소프트웨어의 구현 결과를 직접 확인해 볼 수 있어야 하기 때문이다. 개발 도구로는 EditPlus 편집기와 Java SDK (jdk1.3.1_09)를 이용하여 사용자 인터페이스와 소스 코드를 작성하였다.

<표 3> Desktop PC 사양

System	Pentium IV 2.8GHz(Memory 512)
OS	Windows-XP Professional
에뮬레이터	아로마 WIPI 에뮬레이터, KTF WIPI 에뮬레이터
Editor	EditPlus
Program	Java SDK (jdk1.3.1_09)

WIPI 에뮬레이터를 이용하여 구현 결과를 확인하지만 이러한 에뮬레이터에 보여지는 디스플레이 화면도 다음 <표 4>에서와 같은 기본적인 휴대폰 단말기 하드웨어 요구사항 조건을 충족시키는 에뮬레이터 환경이 된다.

<표 4> 휴대폰 단말기 H/W 요구 사항(최소 권장 사양)

하드웨어	요구사항
디스플레이	스크린 크기 : 96 * 54 이상 색상 : 회색조 4가지 이상 또는 천연색 256 이상
입출력장치	입력 장치 : 한손 키패드, 두 손 키패드 혹은 터치 스크린 출력 장치 : 스크린
사운드 장치	진동 및 비프음
네트워크	무선 및 시리얼을 통한 전송
비 휘발성 메모리	API 라이브러리: 600KB 이상 응용프로그램 관리자 및 기본 응용프로그램: 400KB 이상 응용프로그램: 파일 시스템 공간으로 500KB 이상
휘발성 메모리	응용프로그램에서 사용 가능한 HEAP 영역: 300KB 이상 API 라이브러리에서 사용 가능한 영역: 20KB 이상 지원

본 연구의 소프트웨어 구현은 자바 모바일용 언어인 Jlet을 이용하여 프로그램을 작성하였다. 작성된 소스들을 컴파일하여 jar 파일을 생성한 후, 이 jar 파일을 각 이동통신사들의 에뮬레이터를 사용하여 구현한 결과를 확인한다. Jlet의 구체적인 실행과정을 살펴보면 다음과 같다.

Jlet은 다음의 [그림 14]와 같은 3가지의 라이프사이클로 동작한다. [그림 14]에서 Jlet은 각 상태로 이동하기 위해 startApp()와 pauseApp(), resumeApp(), destroyApp()의 4가지 함수를 사용한다. 먼저 startApp()는 protected abstract void startApp(String[] args)로 표현하며 이는 Jlet이 시작될 때 호출된다. pauseApp()

함수는 protected void pauseApp()로 구현할수 있으며, 프로그램을 일시 정지시킬 때 호출하는 함수이다. resumeApp() 함수는 protected void resumeApp()로 표현하고 일시 정지된 프로그램이 재개될 때 호출된다.



[그림 14] Jlet의 생명주기

destroyApp() 함수는 protected abstract destroyApp(boolean unconditional)로 표현하여 사용하고 프로그램이 종료될 때 호출된다. 인자 값이 true면 무조건 종료. False이면 JletStateChangeException 발생 후 종료된다. 이와 같이 모든 Jlet App들은 이 4가지 함수를 전부 구현해야 한다.

2. 시스템 구현

본 연구는 교사업무수첩과 관련된 4가지의 응용소프트웨어를 구현하였다.

[그림 15]는 WIPi 휴대폰 단말기용 에뮬레이터에서 실행된 전자교무수첩의 메인 화면 모습이다. [그림 9] 시스템 전체 구조에서 보는 바와 같이 메인 화면은 학급경영, 교무일지, 학생일람, 기타사항의 4개 메뉴로 구성되어 있다.



[그림 15] 메인 화면

다음의 [그림 16]은 메인 화면을 나타내는 소스의 일부이다.

```

displayLCD = Display.getDefaultDisplay();
shell = new ShellComponent();

list = new ListComponent(ListComponent.SELECT_IMPLICIT);
ListItemComponent item1 = new ListItemComponent("학급경영");
ListItemComponent item2 = new ListItemComponent("교무일지");
ListItemComponent item3 = new ListItemComponent("학생일람");
ListItemComponent item4 = new ListItemComponent("기타사항");

list.addComponent(item1);
list.addComponent(item2);
list.addComponent(item3);
list.addComponent(item4);

list.setActionListener(new ActionListenerHandler(),null);
list.setChangeListener(new ChangeListenerHandler(), null);
중략
protected void destroyApp(boolean arg0) { }

```

[그림 16] 메인 화면 소스 일부

[그림 16]의 소스를 보면 각 메뉴들을 ListComponent 객체 list에 Change Listener를 등록해 주어 list에서 방향키가 눌렸을 때 Change ListenerHandler의

Changeed() 메소드가 호출되어 동작이 일어나게 한다. 이 소스의 전체를 실행하면 [그림 15]의 메인화면과 같은 결과를 얻을 수 있다.

소스 마지막 부분의 protected void destroyApp(boolean arg0)는 [그림 14]의 Jlet의 생명주기 과정에서처럼 프로그램이 종료될 때 destroyApp() 함수를 사용한 것을 볼 수 있다.

2.1 학급경영

[그림 17]은 학급경영의 하위메뉴인 출석부, 학급특성, 교사시간표를 교사가 필요에 따라 선택하도록 구성된 있는 모습을 보여주고 있다.



[그림 17] 학급경영 화면

답임을 맡은 교사는 학급경영에 필요한 내용 중 출석부와 그 학급의 특성을 기록하게 되고 교사자신의 시간표를 관리하게 된다.

학급 경영은 새로운 학기나 학년마다 그 학급의 교훈, 급훈, 학급의 특성을 기

입할 수 있도록 관리될 것이다. 이러한 실행결과가 구현되는 과정은 메인메뉴 소스와 거의 동일하다.

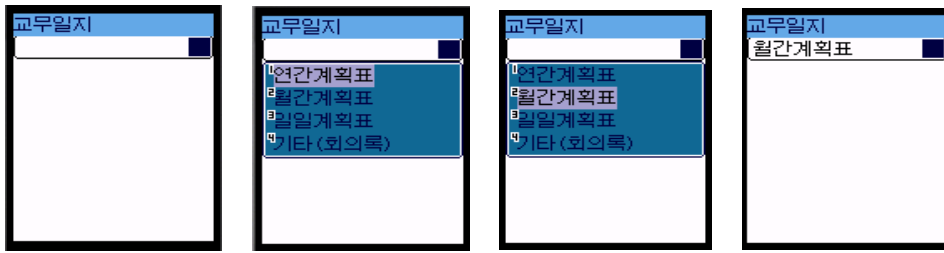
2.2 교무일지

[그림 18]은 메인화면의 두 번째 메뉴인 교무일지를 선택했을 때의 모습을 보여준다. 교무일지의 하위메뉴에는 연간계획표와 월간계획표 그리고 일일계획표를 선택하여 필요에 따라 원하는 사항을 기록하도록 구성되어 있다. 또한 회의록 메뉴에서는 교무회의에 참석하거나 동학년 회의에 참석했을 때의 내용들을 기록하도록 되어 있다.



[그림 18] 교무일지 화면과 하위메뉴

다음의 [그림 19]에서는 교무일지 화면에서 월간계획표를 작성하고자 할 때의 과정을 보여주고 있다.



[그림 19] 교무일지 화면의 월간계획표 선택

교무일지는 교사들이 회의 시간의 내용을 기록하거나 전달 사항을 기록하여 필요시에 언제든지 특정한 날의 회의 내용이나, 그에 대한 일정 그리고 회의에서의 전달사항을 확인할 수 있다. 검색은 내용이 등록된 날로 검색이 이루어지며 일간/주간/월간 별로 검색할 수 있도록 화면을 설계하였다.

```

displayLCD = Display.getDefaultDisplay();
shell = new ShellComponent();

ComboComponent cmb = new ComboComponent();
FormComponent form = new FormComponent();

cmb.append("연간계획표");
cmb.append("월간계획표");
cmb.append("일일계획표");
cmb.append("기타(회의록)");

```

[그림 20] 교무일지 메뉴 소스 일부

위 [그림 20]의 교무일지 메뉴 소스를 보면 팝업메뉴를 구성하기 위해 ComboComponent의 객체 cmb를 생성한 후, 이 cmb의 부모 Component가 될 FormComponent의 객체 form을 생성한다. 그 아래의 소스 cmb.append("연간계획표")처럼 메뉴 항목들을 cmb에 추가시키는 것이다. 이렇게 하면 0번 메뉴 항목

은 “연간 계획표”가 되고 1번 메뉴 항목은 “월간계획표”가 된다. 연간계획표를 비롯한 하위메뉴들에는 이렇게 각 메뉴 항목마다 index가 붙는다.

2.3 학생일람

[그림 21]은 학생일람 화면에서 학생의 신상정보와 성적일람 그리고 학생 연락망을 작성하려고 할 때 사용하는 메뉴이다.



[그림 21] 학생일람 화면과 하위메뉴

학생 일람 화면에서는 학생의 기본 신상정보를 조회, 검색할 수 있고, 변동사항이 발생하면 그 즉시 수정이 가능하도록 하였다.

학생 신상정보는 학생의 정보와 그 학생의 보호자, 교우관계, 진로상담을 기록할 수 있다.

2.4 기타사항

[그림 22]는 기타사항 화면을 나타낸다. 기타사항이라는 화면을 두어 교사가 다른 교사들의 비상연락망을 입력, 조회할 수 있도록 교사 주소록 메뉴를 설계하고 구현하였다. 또한, 전달사항 메뉴에서는 교사가 어느 곳에서든지 학생들에게 전달하고자 하는 일이 발생하거나 아이디어가 떠오를 때 즉시 입력할 수 있도록 하였다.



[그림 22] 기타사항 화면과 하위 메뉴

[그림 23]은 기타사항 화면의 하위메뉴 중에서 교사가 자신의 주소록을 작성하는 과정을 보여주고 있다.



[그림 23] 교사주소록 입력 화면

[그림 23]의 교사주소록 입력 화면은 TextFieldComponent 생성자를 이용하여 구현된 화면이다. TextFieldComponent("", TextComponent.CONSTRAINT_ANY) 부분에서 입력 제한자가 CONSTRAINT_ANY로 아무거나 입력을 받을 수 있는 상태로 하여 textName을 생성하고 "02-", TextComponent.CONSTRAINT_NUMBER로 입력 제한자를 설정하여 textPhone을 생성한다. 마찬가지로 textEmail도 ("@sunshin.ac.kr", TextComponent.CONSTRAINT_EMAILADDRESS)로 입력 제한자를 설정하여 이메일 주소를 입력받도록 하였다.

다음의 [그림 24]는 [그림 23]의 결과 화면을 실행시켜주는 주소록 메뉴 소스의 일부분이다.

```
textName = new
TextFieldComponent("",TextComponent.CONSTRAINT_ANY);
textPhone = new
TextFieldComponent("02-",TextComponent.CONSTRAINT_NUMBER);
textEmail = new
TextFieldComponent("@sungshin.ac.kr",TextComponent.CONSTRAINT
_EMAILADDRESS);
form.addComponent(new LabelComponent("이름"));
form.addComponent(textName);
form.addComponent(new LabelComponent("전화번호"));
form.addComponent(textPhone);
form.addComponent(new LabelComponent("E-mail"));
form.addComponent(textEmail);

button = new ButtonComponent("확인", null);
button.addActionListener(this,button);
shell.setCommand(button,false);

shell.addComponent(form);
shell.setTitle("교사주소록");
shell.show();
```

[그림 24] 주소록 메뉴 소스 일부

3. WIPI 단말기 포팅 및 상용화

WIPI 응용프로그램을 제작하였으면 실제 단말기에 포팅하고 상용화 하는 과정을 거쳐야 비로소 서비스가 가능하게 된다. BREW와 달리 WIPI는 응용프로그램 모듈을 데이터 케이블로 단말기에 업로드하는 것이 아니라 무선 인터넷 망을 이용하여 다운로드 한다[22]. 이를 위해서 단말기 모듈의 컴파일은 COD 서버에서 이루어지고 테스트 ADS(Application Download Server)에 등록된다. 따라서 개발 단계에서는 테스트 단말기로 테스트 ADS에 접속해서 응용프로그램을 설치하게 된다. 개발이 완료되면 상용화 절차를 거치게 되며 모든 상용화 절차가 완료되어야 상용 ADS에서 실제 서비스가 이루어진다. 모바일 응용프로그램의 상용화 과정은 이동통신사의 정책과 서비스 Flow에 따라 매우 상이하다.

V. 결론 및 제언

본 연구는 무선 인터넷 기술의 발전으로 인한 모바일 환경에 발맞추어 교육 현장에서 활용 가능한 휴대폰 단말기용 교육용 콘텐츠를 모색한 결과, 교사들을 위한 전자교무수첩시스템을 설계하고 구현하였다. 이는 기존의 PDA를 이용하였던 전자교무수첩시스템을 분석하고 한계점을 개선한 시스템으로서 현재 보급률과 대중성이 높은 휴대전화에 다운받아서 내장하여 실행되도록 한 것이다. 또한 기존의 PDA용 전자교무수첩 연구의 현실적인 한계를 분석하여 좀 더 실용적으로 활용하도록 개선하였다. 본 연구의 전자교무수첩 시스템은 대중적인 모바일 기기로서 위치가 확고한 휴대폰 단말기용 응용프로그램이므로 교사가 언제 어디서나 추가 비용 없이 휴대폰 단말기에서 학생관리를 할 수 있으며 학생은 장소에 구애 없이 교사에게서 피드백이 가능하다. 본 연구는 한국 무선 인터넷의 표준 플랫폼인 WIPI에 대한 전반적인 연구를 거침으로서, 인터넷 강국이지만 플랫폼면에서는 아직은 해결 과제가 많이 남아있는 국내 현실을 인식하게 되는 연구이기도 하다. 현재 WIPI 2.0 버전이 완료된 가운데 WIPI에 대한 전망은 아직은 혼재된 상황이다. WIPI는 성능과 구현에 있어 미래가 밝은 플랫폼이지만 시장 상황 때문에 개발이 매우 까다롭다는 문제점이 있다. 또한, 플랫폼 보안의 신뢰성 부분도 지원되어야 하는 바, WIPI 플랫폼 보안 기법의 세 단계인 일반(Public) 수준, 콘텐츠 개발자(CP) 수준, 시스템(System) 수준의 보안이 지원되어야 하는 문제도 해결해야 할 과제이다.

이 같은 현실에서 휴대전화 단말기를 이용한 교사용 소프트웨어가 교육현장에서 실제로 활용될 수 있도록 교육인적자원부의 해당 부서나 정보통신부의 적극적 지원도 필요하다 할 수 있겠다.

참 고 문 헌

- [1] 박 성 우 “J2ME와 WIPI의 호환성 연구” 충남대학교 교육대학원 석사논문, 2002.
- [2] 배 석 회, 한 상 흥, 전 영 준, “클릭하세요 WIPI”, 도서출판 대림, 2004, pp.234, pp.277.
- [3] <http://www.kwisforum.org/> (한국무선인터넷표준화포럼)
- [4] 강 상 원, 임 석 진, 심 양 섭, “모바일 플랫폼 천하통일! WIPI 프로그래밍”, 제우미디어, 2004.
- [5] 김 녕 회, “교사업무보조를 위한 휴대용 정보관리 S/W 설계 및 구현”, 연세대학교 교육대학원 석사논문, 1999.
- [6] 류현주, Windows CE 기반 이동단말기용 전자교무수첩 시스템 설계 및 구현, 석사학위논문, 2002
- [7] 이민정, Embedded Linux 기반 전자교무수첩 설계 및 구현, 석사학위논문, 2002
- [8] 주상현, “리눅스용 WIPI 플랫폼을 위한 AOTC 설계 및 구현”, 2005
- [9] 이왕룡, “WIPI 기반의 원격 모니터 및 정보가전 제어 시스템의 설계 및 구현”, 2004
- [10] 정 영 오, “모바일 자바 PDA 핸드폰 프로그래밍”, 도서출판 PCBOOK, 2002, pp.44, pp.77.
- [11] 박수원 외 2인, 위피 모바일 프로그래밍, 한빛미디어, 2003, pp.67, pp.244.
- [12] 김 석 구, 김 한 규, 안 종 현, “WIPI스쿨과 함께하는 WIPI 모바일 프로그래밍”, 영진닷컴, 2004, pp.34, pp.414.
- [13] 배석회, “위피의 탄생과 그 가능성, 마이크로소프트웨어,” 2002-10.
- [14] WIPI@WORLD 2004 컨퍼런스(WIPI 전문가 커뮤니티 활성화를 위한 개

발자 컨퍼런스).

- [15] <http://www.etri.re.kr/> (한국전자통신연구원).
- [16] <http://www.tta.or.kr/> (한국정보통신기술협회).
- [17] 김철우, “고성능 자바 구현하는 WIPI-COD, 마이크로소프트웨어” 2002-10.
- [18] '위피-DMB' 통합 플랫폼 만든다, 전자신문, 2005-05-02.
- [19] 방송위원회, 신규방송서비스 도입이 방송산업에 미치는 영향, 2005.
- [20] “WIPI 국제표준화 시동걸었다”, 전자신문, 2004-08-16.
- [21] 정재민. “모바일 WIPI 플랫폼에서의 적응형 소프트웨어를 위한 컴포넌트의 재 구성 프레임워크, 2004.
- [22] BREW & WIPI 모바일 프로그래밍 김형구, 2005, pp.622, pp.628.
- [23] 한용재 외 3인, “코스 코디네이터의 역할을 하는WIPI 기반 과목 추천 시스템”, 한국정보처리학회 춘계학술대회, 2004.
- [24] “4월부터 「위피폰」 시대 활짝 열린다”, 디지털타임스, 2005-03-21.
- [25] <http://www.mobilejava.co.kr/> (모바일자바).

ABSTRACT

Design and Implementation of an Electronic Pocket System for School-Affairs based on WIPI

Son, In Sook

Major in Computer Science Education

Graduate School of Education

Sungshin Women's University

As mobile devices popularize, there are a lot of researches in education area to provide various application software based on mobile devices such as cellular phones.

In this thesis, we provide an Electronic Pocket System for School-Affairs as a substitute for manual system. This System fully analyze and reflect requirements of school teachers on pocket system. For efficiency, we propose popularized cellular telephone as main tool of school affairs system for teachers. Teachers can modify, delete, create, and reference information ubiquitously, because cellular telephone devices have advantages on portability, mobile, individuality.

Prior Pocket System for School-Affaires which based on PDA needs some improvement on utility and popularity. By our approaches through case studies with Prior PDA System, we suppose more usable school affairs system based on cellular phone.

To design and implement an Electronic Pocket System for School-Affairs which apply on cellular phone device, we research various platforms of cellular

device and we build this system based on WIPI(Wireless Internet Platform for Interoperability), a standard platform of wireless-internet in Korea. And the application program, Electronic Pocket System for School-Affairs, is implemented by Jlet which is a mobile language of Java.

This system provide consistent user interface by using personal cellular phone device that an user(teachers or school employees) already use it.

WIPI 기반의 모바일
전자교무수첩 시스템
설계 및 구현

우 종 정 교수지도

이 논문을 석사학위 논문으로 제출함.

2005년 11월

성신여자대학교 교육대학원
교육학과 전자계산교육전공
손 인 숙