

휴대폰 상에서 자바를 이용한 웹 콘텐츠의 설계 및 구현

이윤호, 박진형, 윤인모, 김찬수
호남대학교 소프트웨어공학과
e-mail : iyoon@honam.honam.ac.kr

Design and Implementation of Web Contents with Java on the Mobile Phone

Jin-Hyoung Park, Yun-ho Lee, Immo Yoon, Chansoo Kim
Dept. of Software Engineering, Honam University

요약

본 논문에서는 SUN사에서 발표된 Java 기반의 J2ME 이라는 Java 플랫폼을 이용하여 단순히 휴대폰 제조업체에서 지원하는 수동적인 프로그램이 아니라 개발자를 통하여 좀더 능동적인 프로그램을 제시하고자 한다. 현재 사회적으로 큰 관심사인 다이어트에 관한 정보와 지역에 따른 음식점 정보제공 서비스를 제공하는 콘텐츠를 구현하였다. 그밖에 실생활에서 필요한 정보를 웹서버에서 다운로드하여 언제나 자신이 필요할 때 사용할 수 있는 요소를 고려한 콘텐츠를 설계 및 구현하였다.

1. 서론

선 마이크로시스템즈에서 발표된 J2ME(Java2 platform Micro Edition)를 보면 J2ME의 Object device는 mobile와 embedded이다. mobile device는 사람이 들고 이동할 수 있는 반면 embedded는 움직이지 못하는 것으로 나눌 수가 있는데[1] 본 연구에서는 가장 대표적인 mobile device인 휴대폰에서 사용하는 웹 콘텐츠를 구현한다. J2ME는 모바일이라는 일반 컴퓨터 환경이 아닌 이동기기라는 특징으로 일반적인 Java 프로그램을 만드는 과정과는 조금 다르다. 일반 어플리케이션은 컴파일하고 인터프리터에 의해 실행하면 되지만 모바일용 어플리케이션은 컴파일하고 검증하고(휴대폰 스펙에 맞는가를 확인) 다시 검증된 클래스파일을 JAR로 압축하고 실행 환경파일인 JAD를 작성해야만 실행이 가능하다. 현재 휴대폰은 원래 목적인 통화 서비스보다 그 외의 문자 서비스나 웹을 통한 정보검색 서비스들도 많이 사용되고 있다. 하지만 그런 서비스들은 웹브라우저를 통한 실시간 서비스가 주를 이룬다. 그래서 사용시간에 따른 사용료도 만만치 않다. 이에 사용자가 원하는 서비스에 대해서도 요구나 요청을 할 수 없다. J2ME는 자바 어플리케이션을 실행 할 수 있는 VM(Virtual Machine)을 휴대폰에 맞게 장착

해 휴대폰에서 자바 어플리케이션을 구현할 수 있게 해 준다. 즉 사용자만의 어플리케이션을 이동기기에 직접 넣어 가지고 다닐 수가 있다는 것이다. 이러한 특성 때문에 향후 커다란 시장을 형성할 것으로 예상되고 있다.

본 논문에서는 자바의 플랫폼에 대해서 살펴본 후 요구조건에 따라 웹 콘텐츠를 설계하고, J2ME의 특성을 살려 웹 콘텐츠를 구현하고 적용시키기로 한다.

2. 관련연구

망사업자들과 단말기 제조업체들이 자바를 선택하는 이유로는 다음과 같은 다섯 가지를 들 수 있다.[2]

- 1) 동적인 어플리케이션의 다운로드 : 향후의 무선 어플리케이션과 서비스는 실시간으로, 동적으로, 그리고 안전하게 다운로드되며 실행 코드의 네트워크 이동성을 보장하는 솔루션이다.
- 2) 크로스-플랫폼 호환성 : 플랫폼간의 호환성은 차세대 모바일 솔루션의 기본 조건이다. 자바는 플랫폼 호환성에 관해서는 이미 강력한 경쟁력을 가지고 있다.
- 3) 향상된 사용자 경험과 역동성 : 기존의 브라우저기반의 서비스는 유저 인터페이스에 근본적

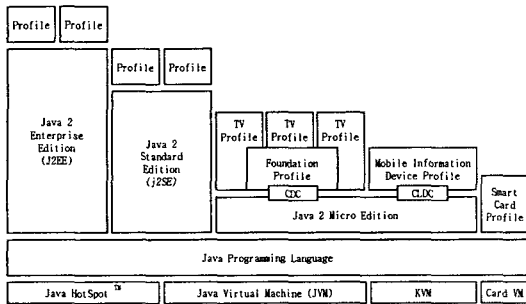
인 한계가 있다. 동적으로 다운로드된 자바 애플리케이션은 보다 향상된 그래픽과 더욱 빠른 응답속도를 통해 사용자들의 만족도를 높여 준다.

- 4) 비연결성 : 항상 네트워크에 연결되어 있을 수는 없다. 자바 애플리케이션을 사용함으로써, 네트워크에 연결되지 않았을 때 작업한 내용을 네트워크에 연결이 가능한 상태에서 동기화하는 것이 가능하다.
- 5) 보안문제 : 아직도 무선인터넷에서의 보안 문제는 해결되지 않고 있다. WAP과 MME, 그리고, iMode는 중단간 보안에 관한 한 완벽한 솔루션을 제시하지 못하고 있다. 자바는 이미 훌륭한 보안 모델을 가지고 있으며, 무선네트워크에서의 보안 문제를 해결할 수 있다.

자바의 이러한 장점들을 바탕으로 선보인 플랫폼이 CLDC(Connected Limited Device Configuration)과 MIDP(Mobile Information Device Profile)이다.

2.1 무선단말을 위한 자바 플랫폼의 개요.

자바 플랫폼은 크게는 두 가지 요소로 구성된다. 자바 언어로 작성되고, 컴파일된 자바 실행 코드를 수행해 주는 자바 가상머신과 자바 프로그램을 작성하기 위해서 기본적으로 사용할 수 있는 클래스 라이브러리, 즉 표준 API 집합이 그 두 가지 요소이다. 그리고, 이러한 자바 플랫폼은 자바2 플랫폼이라는 새로운 마케팅 개념이 가미된 플랫폼으로 진화하면서 J2EE, J2SE, J2ME라는 세 가지의 영역으로 분할되었다. 이것은 각각 엔터프라이즈 서버 시장, 데스크탑시장, 소비자/임베디드 디바이스 시장을 목표로 한 역할 분담이다.



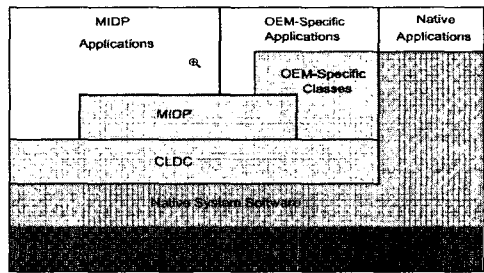
[그림1] 자바 2 플랫폼

[그림1] 은 이러한 자바2 플랫폼의 분할된 형태를 보여주고 있다.[1]

2.2 J2ME 의 개요

MIDP는 모바일 인포메이션 디바이스(MID)를 목표로 CLDC 컨피규레이션을 기반으로 설계된 자바 클래스 라이브러리에 대한 명세이다. 그러므로, MIDP 명세는 CLDC의 명세를 확장하거나, 구체화하거나, 변경을 가하고 있다.[3]

자바실행환경에서 가장 중요한 부분은 플랫폼으로 CLDC(Connected, Limited Device Configuration)와 MIDP(Mobile Information Device Profile)로 정의되는 Java2 Micro Edition (J2ME)이다. 우선 J2ME의 구조를 살펴보면 다음과 같다.[4]



[그림2] J2ME의 구조

[그림2]에서 MID 부분은 Mobile Information Device의 하드웨어를 의미하며, 그 위의 Native System Software는 그 디바이스에서 사용되는 OS와 각종 라이브러리를 의미한다. Native System Software층의 상위에는 차례대로 CLDC와 MIDP, 그리고 OEM-Specific Classes라는 별도의 API가 탑재되는데, 이상과 같은 API를 기반으로 하여 휴대폰 단말기가 실행시킬 수 있는 프로그램들은 다음의 세 가지로 분류된다.

- 1) MIDP Applications : 자바 프로그램으로서 CLDC와 MIDP에서 정의한 클래스만을 사용하여 개발된 응용프로그램
- 2) OEM-Specific Applications : 자바 CLDC와 MIDP에서 정의한 클래스와 개발업체 및 서비스 사업자가 정의한 클래스를(OEM-specific Classes 또는 OEM-specific service API) 사용하여 개발된 응용프로그램
- 3) Native Applications : J2ME와는 별도로 휴대폰단말기 상에서 직접 개발된 응용프로그램으로 WAP 브라우저 등이 포함된다.

3. 설계

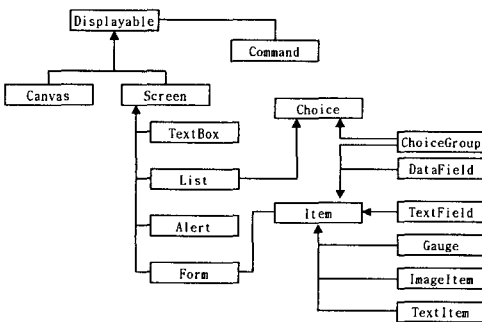
3.1 구체화된 하드웨어의 특징

우선 MIDP에서 좀 더 구체화된 하드웨어 디바이스들의 특징을 살펴보면 다음과 같다.

- 1) 디스플레이 요구사항
 - 96x54 이상의 스크린 사이즈
 - 1비트 이상의 디스플레이(모노크롬)
- 2) 입출력 요구사항
 - 한 손 또는 두손으로 입력가능
- 3) 메모리 요구사항
 - MIDP 컴포넌트를 위한 128K의 비휘발성 메모리 공간
 - 애플리케이션이 생성하는 데이터의 저장을 위한 8K의 비휘발성 메모리 공간
 - 자바 런타임을 위한 32K의 휘발성 메모리 공간
- 4) 네트워킹 요구사항
 - 양방향이고, 무선이며, 제한된 대역폭에, 간헐적으로 연결될 수 있음

3.2 Java상에서 구현을 위한 인터페이스

MIDP는 스크린 기반의 디자인을 가지고 있는데, 스크린은 사용자 입력을 렌더링하는 특정 디바이스 그래픽스를 캡슐화하는 객체라고 정의하고 있다. 다시 말하면 스크린이라는 것은 사용자에게 나타나는 화면이며, 사용자는 한 번에 하나의 스크린만을 볼 수 있고, 사용자 입력에 대한 처리는 이 스크린에 의해서 이루어진다고 할 수 있을 것이다. [그림3]에서는 두 종류의 스크린이 있음을 알 수 있는데, Screen 클래스와 Canvas 클래스가 그것이다.



[그림 3] 유저 인터페이스 계층도

Canvas 클래스는 게임과 같은 하위 레벨 API를 사용하는 애플리케이션을 위한 스크린을 제공하고, Screen 클래스는 텍스트 박스, 리스트, 등의 상위 레벨 API를 위한 최상위 추상화를 제공한다. Screen 클래스를 상속하는 List, TextBox, Alert와 같은 클래스들은 사전에 정의된 구조를 사용할 수 있는 컴

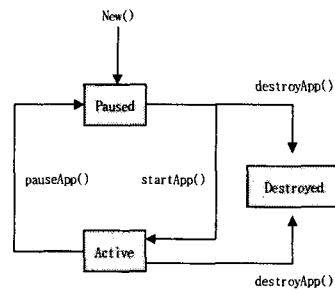
포넌트들이고, 이렇게 사전에 정의된 구조를 사용할 수 없는 TextField, ImageItem, Gauge 등의 클래스들은 Screen 클래스를 상속하는 Form 클래스를 상속하도록 설계되어 있다.

이벤트 처리에서도 상위 레벨 API와 하위 레벨 API에 대한 구분이 있다. 이벤트 처리에서는 [그림 3]의 제일 윗쪽에 놓여진 Command 클래스로 구현된 추상 커맨드에 대한 이해가 필요하다. 대부분의 신형 단말기에는 타원 안에 있는 것과 같은 소프트웨어 키가 있다. 그리고 이 소프트웨어 키를 이용해서 스크린의 아래 부분에 있는 'About'이라는 커맨드를 선택하여 실행시키거나, 위 아래로 화면을 스크롤할 수 있다. 이렇게 추상 커맨드와 소프트웨어 키를 통한 이벤트 처리는 상위 레벨 API를 통해 구현이 이루어진다.

3.3 기본 MIDlet의 인터페이스

MIDlet 애플리케이션은 MIDlet 클래스를 상속받아서 만든다. MIDlet 클래스 안에는 다음과 같은 추상 메소드가 존재한다.

- 1) startApp() 필요한 리소스에 대한 점유 획득. Active 상태로 전이
- 2) pauseApp() 임시적인 리소스들에 대한 점유 해제. Pause 상태로 전이
- 3) destroyApp() MIDlet이 종료한 상태. 공유 리소스를 해제한다. Destroyed 상태로 전이



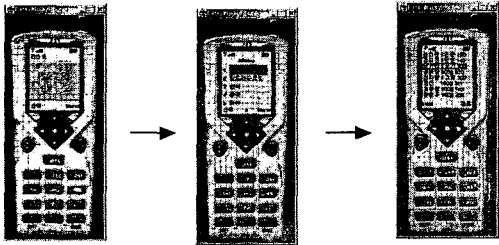
[그림 4] 유저 인터페이스 계층도

4. 콘텐츠 구현

본 논문에서 구현하고자하는 콘텐츠는 단순히 제조회사에서 제공하는 정보가 아닌 시간과 장소에 따라 사용자가 필요로 할 수 있는 정보를 제시하고자 칼로리 정보와 각 지역의 음식점 정보를 구현하였다.

4.1 음식에 대한 칼로리 정보 제공 서비스

다이어트는 여성들만의 문제를 벗어나서 사회적으로 큰 관심사로 떠올랐는데 이 서비스는 장소에 관계없이 음식에 대한 칼로리 정보를 제공받음으로써 다이어트의 효과를 극대화 할 수 있다. [그림5]는 컨텐츠 구현 화면이다.



메인화면 리스트화면 데이터화면
[그림 5] 음식에 대한 칼로리 정보제공 서비스

4.1.1 메인 화면

GIF 나 BMP 이미지는 지원을 안하고 오직 png 이미지만을 지원한다.

```
// 소스 코드 - 메인화면
public class TestList extends MIDlet implements CommandListener
{
    Image img; // list에 추가할 이미지 선언.
    public void startApp()
    {
        try
        {
            img=Image.createImage("t.png"); // 이미지 생성
        } catch(Exception e) {} // 파일이 없을 때 예외발생.
        l.append("", img); // 이미지 추가.
    }
}
```

4.1.2 리스트

이름과 칼로리순으로 검색할 수 있도록 하였으며 한식, 중식, 양식으로 데이터를 분류하여 검색의 용이성을 높였다.

```
// 소스 코드 - 리스트
public void startApp()
{
    l=new List("리스트", List.IMPLICIT); // 리스트 생성
    c1=new Command("선택", Command.SCREEN, 1);
    c2=new Command("이름순으로", Command.SCREEN, 1);
    c3=new Command("칼로리순으로", Command.SCREEN, 1);
    // 리스트 안에 들어 올 메뉴들
    rc=new Command("종료", Command.SCREEN, 1);
    lc=new Command("next", Command.SCREEN, 1);
    l.addCommand(c1); // 생성된 메뉴를 리스트에 붙임.
    l.addCommand(c2);
    l.addCommand(c3);
    l.setCommandListener(this); // 리스트에 Listener 추가
    display.setCurrent(l); // 리스트 화면에 출력.
}
```

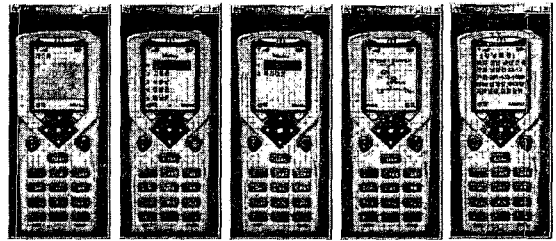
4.1.3 데이터화면

MIDlet 상에서는 파일을 입출력을 할 수 없다. 그래서 정보는 모두 String으로 처리한다. [그림5]의 데이터화면은 String의 배열을 Canvas로 넘겨주어

Canvas배열로 받고 그 Canvas배열을 화면에 출력한다.

4.2 지역에 따른 유명한 음식점 위치 제공 서비스

신혼여행, 배낭여행 등 타 지역으로 이동시에 휴대폰을 이용하여 그 지역만의 별미를 맛볼 수 있는 정보제공서비스로써 정확한 위치 전달과 신뢰도가 높은 정보가 요구된다.



(6-1) (6-2) (6-3) (6-4) (6-5)

[그림 6] 지역에 따른 유명한 음식점위치 정보제공 서비스

[그림 5]와 [그림 6]의 화면은 Top-Down 형식으로 구현하였으며 [그림 6-4]에서는 위치정보를 좀더 정확하게 전달하기 위해 문자가 아닌 이미지로 구현하였다.

5. 결론

웹 컨텐츠 구현에서는 이동환경과 사용요금, 화면구성의 제약조건 등 이동전화 서비스 환경을 고려하여 구현하는 것이 무엇보다도 중요하다. 이러한 사항들을 바탕으로 본 논문에서는 MIDP 명세를 분석/설계하여 이를 이용한 칼로리 정보 제공 서비스와 지역에 따른 음식점 위치 정보제공 서비스를 구현하였다.

앞으로 계속 연구되어야할 과제는 이미지 로딩시 속도향상에 관한 연구와 변화하는 정보를 지속적으로 개선하기 위한 단순화된 데이터 구조에 대한 연구이다.

참고문헌

[1] Java™ 2 Platform, Micro Edition (J2ME™ Platform) <http://java.sun.com/j2me>
 [2] Eric Giguere "Java™2 Micro Edition" by John Wiley & Sons Inc 2000
 [3] Mobile Information Device Profile(MIDP) <http://java.sun.com/products/midp>
 [4] JSR-000037 Mobile Information Device Profile for J2ME™ <http://jcp.org/aboutJava/communityprocess/~review/jsr037/index.html>