

위치정보를 위한 J2ME 기반의 모바일 클라이언트 시스템 설계 및 구현 (Mobile Client System Design & Implementation of J2ME Base for Location Information)

이창희(Chang-Hee Lee)* / 전형준(Hyung-Joon Jeon)** / 나극환(Keuk-Hwan Ra)***

요 약

본 논문에서 모바일 단말기를 클라이언트로 사용하는 주차관리 시스템을 설계하였다. 클라이언트는 주차관리 웹 사이트에 접속한 후 주차시간, 주차정보, 주차요금에 관한 정보를 웹 서버에서 제공받으며, 클라이언트의 차에 대한 정보를 감지하는 센서를 통하여 클라이언트의 주차 위치를 정보를 파악 하여 크고 복잡한 대형주차장에서 사용자에게 매우 편리한 인터페이스를 제공하고 원활한 관리를 할 수 있는 시스템을 설계를 구현한다.

Abstract

This paper is a parking management system which uses mobile terminal equipment by the client. From the Web server, the client will be served about the information concerned with the location of parking lot, parking time, and payment after accessing the parking management web site. After the sensor which is set in the parking lot perceives the information about the client's car, it will inform the client of the parking place. This system is designed to control the large and complex parking lot and serves a very convenient interface to the user.

1. 서론

1980년대부터 제조분야에서 많은 화를 겪어 왔다. 저비용 고품질 제품을 위해서 제조자는 대량생산을 포기하고 JIT (Just In Time

:적시생산시스템), EDI(Electronic Data Interchange)등을 선호한다. 이러한 새로운 제조 방법은 거대한 정보와 데이터를 동시에 관리해야 하는 엔지니어에게 변화를 가져왔다. 사실 이전보다 극적으로 된 효과적인 정보를

*정회원 광운대학교 전자공학과 박사수료

**정회원 삼성에스원 기술 연구소

***정회원 광운대학교 전자공학부 교수

논문접수 : 2003. 9. 2.

심사완료 : 2003. 9.23.

*이 논문은 2003년도 광운대학교 교내학술연구비 지원에 의해 연구되었음

어떻게 활용하고 제어하는가 하는 것이 문제가 되었다[1].

인터넷의 출현은 인류가 통신하고 세계를 보는 방법에 기본적인 변화를 가져왔다. 인터넷은 언제 어디서나 정보를 분배하는 세계적인 배관을 제공한다. WWW은 임의의 플랫폼 상에서 (수치적, 그래픽) 정보를 표현할 수 있게 한다. 인터넷에서 실시간 전송을 사용하여 운영자는 기기 세팅이나 제어공정 진행을 급속히 최적화하고 문제가 나타나기 전에 해결하며 현장이 아닌 원격에서 전달한다.

1990년대 이후 유선망을 이용한 인터넷의 비약적 발전으로 네트워크와 연결된 컴퓨터가 있으면 원하는 정보를 얻을 수 있다. 개인컴퓨터는 통신만을 사용하기 위해서는 기능이 복잡하고 큰 부피로 인하여 소형의 장비에 간단한 기능 요구가 대두되었다. 특히 무선전화기나 PDA (Personal Digital Assistant)와 같은 무선 터미널을 이용한 웹 접속이 요구되었다.

현재 이동통신망 가입자중 무선인터넷 이용자는 828만 명으로 전체 이동통신가입자중 32.3%에 이르고 앞으로 더 많은 사용자가 이용될 것이다[2]. 유선인터넷과 무선인터넷을 비교한 설문들을 통하여 유선인터넷과 경쟁하려는 전략은 지양되어야 할 것으로 판단되며, 이동성이라는 무선인터넷의 장점을 최대한 살려 무선인터넷만의 차별화된 서비스를 개발하는 것이 가장 중요한 생존전략으로 판단된다[3]. 무선 인터넷 서비스를 정의하면, 이동통신과 인터넷서비스의 결합으로 Anyway, Anytime, Anydevice, Anymedia를 통하여 편재하는(Ubiquitous) 환경 하에서 인터넷의 이용을 가능하게 하는 서비스를 말하며, 이동중에도 무선으로 다양한 정보를 송/수신할 수 있는 서비스로 환경을 구축하여야 한다.

사람이 인터넷 접속을 위해서는 유선을 사용한 컴퓨터로의 이동이 요구되지만 무선 인터넷을 사용하면 바로 휴대용 기기를 이용하여 접속이 가능하다. 증가하는 무선인터넷 사용자의 수요와 소득증대에 따른 차량의 증가에 따라 많은 주차장이 필요하고 대형시설 등

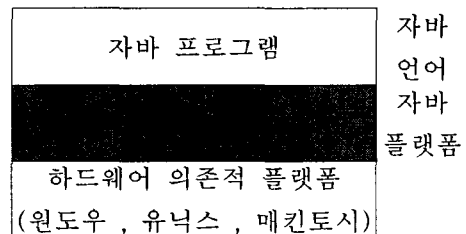
에 부속한 차량의 주차는 큰 불편으로 대두되거나 아직 자동화 체계가 부족하여 입 출구에서의 정체와 주차하기위하여 큰 주차장을 배회하게 되고 일이 끝나고 주차한곳을 기억하기 어렵다. 이에 주차장을 관리하는 부분에서는 신속하고 빠른 정리를 할 수 있고 주차하는 사람들 입장에서는 주차에 허비하는 시간을 줄이고 자신의 주차 위치를 찾는 등의 주차에 따른 불편함을 감소할 수 있게 무선 인터넷 단말기를 사용한 주차 시스템을 개발하여 일반대중으로 하여금 무선 인터넷 서비스로서 유용한 활용가치를 주어 무선 인터넷서비스의 이점을 살려본다.

2. 이론적 배경

2.1. 자바

2.1.1. 자바

자바(Java)는 두 가지로 나누어 볼 수 있다. 먼저, 기계어, 어셈블리어, 포트란, 또는 C 등과 같이 프로그래밍을 하기 위해 사용하는 자바 언어가 있고, 자바 언어를 이용하여 프로그래밍 하기 위해 사용할 수 있는 자바 API (Application Programming Interface)와 자바 프로그램을 실행시켜 주기 위한 자바 가상머신 등을 가리키는 자바 플랫폼이 있다. 다시 말해서, 자바 언어는 Visual C++와 비유될 수 있고, 자바 플랫폼은 윈도우 95/98/NT 및 윈도우 95/98/NT API와 비유될 수 있다.



[그림 1] 자바(Java)

[Fig1] Java

그림1과 같이 자바 언어를 이용하여 작성한 자바 프로그램은 자바 컴파일러를 이용하여

자바 바이트코드로 컴파일 되고, 이 자바 바이트코드는 자바 가상머신에 의해 해석되어 실행되는데, 이때 자바 가상머신은 자바 바이트코드에 대한 해석기 즉 인터프리터로 동작하게 된다.

2.1.2 자바의 특징

- 객체 지향적이다(Object-Oriented)
- 보안성(Security)을 갖는다
- 아키텍처 중립적이고 이식성이 높다
- 다중 스레드(Multi-thread)를 지원 한다

2.1.3. Java 2 플랫폼과 J2ME

(Java 2 Micro Edition)

Java 2 플랫폼은 프로그래밍언어인 Java , 실행환경과 개발 환경을 전체적으로 규정하는 말이다. 현재 Java2 플랫폼에서는 특정 그룹의 애플리케이션에 따라 J2EE ,J2SE,J2ME의 3가지 Edition 으로 나눈다[4-10].

J2ME 는 Hand Held Device 나 PDA, Set-top Box, net TV 와 같은 네트워크로 연결되어 있고, VM 자체가 일반 JVM 보다는 가벼운 VM , Core API 가 올라가게 된다. CDC는 Set-top Box, net TV 와 같은 Device 의 설정 이다. CLDC 는 Hand Held Device 을 위한 설정 이다. PDA 와 같은 경우는 CDC와 CLDC 의 중간 형태라 할 수 있다. CLDC 나 CDC 위에 Profile 이라는 부분이 올라가게 된다.

그림2는 Hand Held Device 에 올라가는 J2ME 환경을 예로 나타낸 것이다

Profile
Configuration
Java Virtual Machine
Host Operation System

[그림 2] Java 2 Micro Edition Software Layer

[Fig2] Java 2 Micro Edition Software Layer

2.1.4. J2ME 설정과 프로파일

① J2ME 설정(Configuration)

단순한 휴대용 무선전화기에서 PDA, HPC(handheld PC), 스마트 폰(휴대용전화기와 PDA의 혼합기종)까지 그 디바이스의 범위가 광범위 하다. 이런 디바이스의 특성을 감안하여 디바이스용 자바가상머신은 여러 가지 범주로 나누어지고 설정과 프로파일이라는 개념이 생겨났다.

설정은 가상머신의 특징을 규정하는 것으로 디바이스의 지원능력에 따라 자바 언어의 명세서인 자바 언어 스펙 에서 어느 정도까지 지원할 것인지를 규정한다, 네트워크나 보안 같은 필수 자바 라이브러리 지원여부에 대하여 최소한의 범위도 규정한다. J2ME의 설정에는 CLDC와 CDC가 있다.

설정 레벨에는 두 가지 형태의 클래스들이 정의된다. J2SE에서 상속받은 클래스와 소형 장치들을 위해 특별히 설계된 클래스이다.

상속받은 CLDC의 클래스 라이브러리들은 CDC에 대해서도 상향 호환성이 있으며 CDC의 상속받은 클래스 라이브러리들은 J2SE에 대해서 상향 호환된다.

프로파일은 이런 설정에서 사용자 인터페이스 등을 포함하는 추가 API 를 규정한다, 이는 작은 틀의 범위를 정하는 점에서 설정이 정해놓은 큰 틀 안에서 각각의 디바이스에 맞도록 세부적인 범위를 정하는 것이라고 볼 수 있다[4,6-8,10].

설정이 정해지면 가상머신의 종류도 설정에 의해 분류된다. 하지만 UI(User Interface)나 추가적인 라이브러리를 규정하는 나머지 연역은 전부 프로파일에서 규정되므로 본격적인 프로그램을 하려면 프로파일에 속하는 API를 사용해야만 한다.

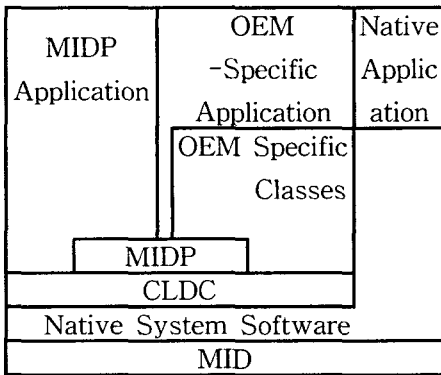
② J2ME Profile

Profile 은 특정 Device type 이나 Vertical Market 을 위해서 설정 의 위단에서 제공되어지는 자바 Technology 베이스의 APIs이다.

설정 기반에서 Target Device 의 Type 에 맞게 Display toolkit이라던가 User input Method, Persistent 데이터 저장 공간, 메시지(SMS, Email, 등), 보안 그리고 무선 환경에서의 네트워크 처리 부분에 대해서 명세하게 된다.

2.2. MIDP(Mobile Information Device Profile)

KVM 기반의 Mobile Device 을 위한 Profile 이다. MIDP 는 Display toolkit 과 User interface methods, Persistent data storage, Messaging (SMS, Email, 등) 과 보안, Wireless Telephony Connection 을 지원한다.



[그림 3] MID인 Device 의 구조
[Fig3] MID Device structure

그림3의 맨 하단에 Device가 존재하고, 그 위에 Native System Function을 제공하는 S/W가 올라간다. 이는 Device 의 Native Method는 System Call을 위해서 사용되기 때문에 애플리케이션 레벨까지 영향이 미치게 된다. 그 위에 KVM 과 Core App 가 있는 CLDC가 올라가고 CLDC의 일부 기능과 Device Profile을 갖는 MIDP 가 올라간고 MIDP 애플리케이션은 그 위에서 수행 된다.

2.3 무선결제서비스 제공방식

무선결제시스템의 처리과정은 일반적으로

(i) 고객확보, (ii) (virtual) POS 에서의 지급, (iii) 결제대금의 과금 및 청구, (iv) 결제서비스 제공기관과 소비자 간의 정산 및 결제 등으로 구분된다. 대부분의 무선결제서비스는 결제시스템에 상관없이 비슷한 처리과정을 거친다. 그러나 (ii)단계인 POS에서의 지급은 결제시스템마다 약간의 기술적 차이를 보인다. 특히, 거래정보의 입출력과 무선망을 이용한 정보전달의 방식에 따라 소프트웨어 방식은 물리적인 카드 접촉 없이 휴대폰 무선망을 통해 실시간 인증과 결제시스템을 연동하는 방식이고, 하드웨어 방식은 휴대폰에 스마트카드나 칩을 장착하여 인증과 결제가 이루어지는 방식이다.

본 논문에서는 무선결제 방식은 휴대폰의 내장 기술로 결제하는 것으로 가정하였다

2.4 LBS (Location-Based Service)

LBS는 미래형 경제시스템인 모바일 커머스의 활성화를 위한 기본 인프라이며, 개인의 위치정보 누출로 인한 사생활 침해, LBS 기술 표준화 와 호환성, 투자비용 부담 문제 등 LBS의 해결되면 위치정보 자체가 자원이 되는 신 개념의 L-커머스(Location -Commerce) 시대가 도래 할 것이다[11].

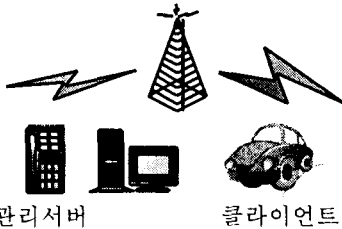
제안된 LBS 방법들은 주차장의 환경에서는 GPS는 밀폐된 주차장 공간에 사용이 어렵고, 핸드폰의 셀 단위를 사용은 작은 범위에서 불필요한 설비증설이 요구되고, 단말기 내장형 유비쿼터스는 동일한 장치를 갖는 문제점등을 지니고 있다. 또한 개인의 프라이버시라는 큰 장애물은 자신은 위치를 알고 싶어 하지만 자신의 위치는 노출을 꺼리는 양면성이 있으나 주차장이라는 조건은 자신의 차량의 위치를 알고자 하는 요구는 크다. 이런 여건상 주차하는 차량주인 자신이 위치를 확인해주고 관리자는 정확한 주차가 이루어졌는지를 확인하는 것이 주차장의 특성상 효율적이고, 서버에서 주차 위치 지도를 제공하고 핸드폰으로 확인하는 작업으로 LBS 기능을 대신할 수 있게 함으로서 복잡한 주차장에서의 자신의 차를

용이하게 찾는 기능을 사용하였다.

3. 모바일서버 클라이언트시스템

3.1. 시스템 구성

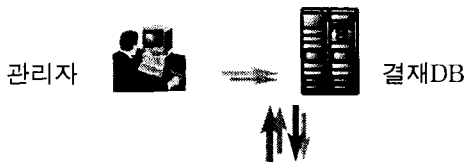
그림 4는 모바일을 사용한 클라이언트의 서버 접속도로 주차장에 진입한 클라이언트는 무선 인터넷을 사용하여 주차장 관리 서버 시스템에 접속하여 서버에서 제공하는 주차 관련 정보를 제공받을 수 있게 된다.



[그림 4] 서버 클라이언트 접속도

[Fig4] Connect with Server and Client Diagram

서버의 주역할은 주차시스템에 접속한 클라이언트를 등록하여 주차시간을 기록하여 출차시 주차시간을 계산하여 줄 수 있는 데이터베이스체계를 갖는다. 주차 초기에 클라이언트에게 주차가능 구역을 알려주어 주차흐름의 원활한 기능을 제공하고, 지하 또는 작은 구역의 주차장이라는 특수한 상황에서 기존의 모바일 장비 또는 GPS를 사용한 위치 추적 시스템이 위치파악 기능을 발휘하기 힘든 점을 보완하여 주차위치를 주차장에 설치한 센서를 사용하여 주차위치를 확인함으로써 위치정보를 요청하는 클라이언트에게 주차 위치 정보를 제공하여 대형 주차장에서 겪는 비슷한 주차환경과 혼동을 차량위치 정보를 제공하여 주차장의 서비스 질을 높인다.



[그림 5] 서버운영도

[Fig5] Operation Server Diagram

1) 센서장비

광센서를 사용한 스위치 신호를 사용하여 차량의 진입 및 출차를 확인하고 센서를 감지하는 CPU에서 각각의 위치정보와 함께 액세스 포인터로 전송하게 한다.

2) 액세스 포인터

각 센서의 데이터를 서버에게 전달하기 위하여 액세스 포인터로 데이터를 수집하여 위치정보를 서버에 기록하게 한다.

3) 주차관리 DB 운영

일별, 주별, 월별로 관리를 백업하여 주차장의 주차내역을 기록하여준다.

4) 결제내역 DB 운영

신용결제인 경우 데이터를 해당 서비스업체에게 전송하기 위하여 데이터베이스를 구축하여 기록 및 전송한다.

4. 시스템 환경 설정

서버의 하드웨어는 펜티엄 III급으로 linux RedHat8.0을 사용하여 구축하였다.

<표1> 설치에 필요한 파일들

<table1> installed Files

파일명
MySQL-4.0.13.tar.gz
Apache.2.0.44.tar.gz
php-4.3.0.tar.gz
ZendOptimizer-1.3.1-Linux_glibc21-i386.tar.gz
Java-j2sdk1.4.1_03
Tomcat-4.1.24
Tomcat-Connector4.1.24

<표 2> 프로그램 위치.
<table2> program location.

프로그램 명	역 할	위 치
Ticket.java	무선 톨킷 프로그램	C:\J2mewtk\apps\Ticket\src\
TicketServerQ.jsp	클라이언트와 서버의 연결 루트 프로그램	*/tomcat/webapps/examples/
TicketServer.jsp	클라이언트와 서버의 연결 루트 프로그램	*/tomcat/webapps/examples/
Id_CheckTicket.java	서버의 DB mySQL에 접속하여 상수전달	*/tomcat/webapps/examples/WEB-INF/classes/
DBConnectTicket.java	서버의 DB mySQL에 접속 연결 프로그램.	*/tomcat/webapps/examples/WEB-INF/classes/
index.html	서버 접속 화면	*/apache/htdocs/cha/nhe/
list.php	주차 리스트	*/apache/htdocs/cha/nhe/parkadmin/
write.php	주차리스트 쓰기	*/apache/htdocs/cha/nhe/parkadmin/

* /usr/local

4-1. J2MEWTK (Java2 Micro Edition Wireless Tool Kit) 사양

J2MEWTK 의 시스템사양은 다음과 같다.
OS : Microsoft Windows 98 Second Edition.
JDK: J2SE, version 1.3.0을 다운로드 받을 수 있다. J2MEWTK는 반드시 JDK1.3이상이 설치되어야 한다[12].

4-2 . 아파치

아파치는 1995년 그 당시 가장 인기 있었던 웹서버 중의 하나인 NCSA HTTPD 1.3버전을 기반으로 탄생하여 그 후 기능을 더욱 향상 시켜 Apache 웹서버를 발표하였다. 현재 인터넷 웹서버 중 가장 인기를 누리는 S/W다. 지속적으로 패치파일을 제공하고 퍼포먼스가 가장 좋기 때문이다. 유닉스 플랫폼의 버전뿐 아니라 윈도우 NT, 95용의 베타 버전도 나와 있어 다양한 운영체제에서 서버구축이

가능하다. 일종의 HTTP 규약(http v1.1이상)을 지원하는 s/w 이며 웹환경을 구현하려면 여러 가지 기술요소 들이 필요한데 그중에 한 가지이다. html과 cgi등을 운영하는 s/w이므로 그 역할이 가장 중요하고 , 현재 웹서버 s/w는 수백 가지가 있으며 이 웹서버 s/w는 각자 일정한 운영체제 하에서 운영이 되게 설계가 되어 있다.

아파치는 리눅스, 선 솔라리스, 기타 유닉스 계열의 플랫폼을 지원하는 웹서버 중에서 가장 안정적이고 기능이 뛰어난 제품이고, 전 세계 인터넷상의 웹서버의 약 45% 이상이 아파치 웹서버로 구현되어 있다.

4-3 소형기기를 위한 프로그래밍

소형기기가 갖고 있는 제한요건은 응용 프로그램을 작성하는 방법을 바꿔야만 하는 이유가 된다. Java의 소형기기를 위한 프로그래밍 방법은 다음사항을 고려해야한다.

- ①. Java에 대한 확신 없이 사용하지 않는다.
- ②. Server에서 계산을 하도록 하라
- ③. 응용 프로그램을 간단하게.
- ④. 작은 프로그램을 만들도록 한다.
- ⑤. Run-Time시에 적은 메모리를 사용한다.
- ⑥. 성능을 고려한 Code 작성을 염두에 둔다.
- ⑦. Model을 분할한다[13].

5. 시뮬레이션

5.1 구조

5.1.1 기본구성

인터넷 통신 핸드폰 소지자가 주차장에 들어오면 주차장 서버에 접속을 시도하여 주차 정보를 전송한다. 서버에서는 주차 가능한 위치를 전송하고 주차시간과 차량번호, 핸드폰 번호 등을 데이터베이스에 저장한다. 입력되는 정보는 순차적으로 정리되고 주차 전까지 대기상태를 알려준다. 사용자는 주차가 되면 인접한 곳의 핸드폰을 접속하여 주차를 알려주며, 주차알림을 하지 않는 경우는 센서를 통하여 주차 상태를 파악한다. 센서는 각각의

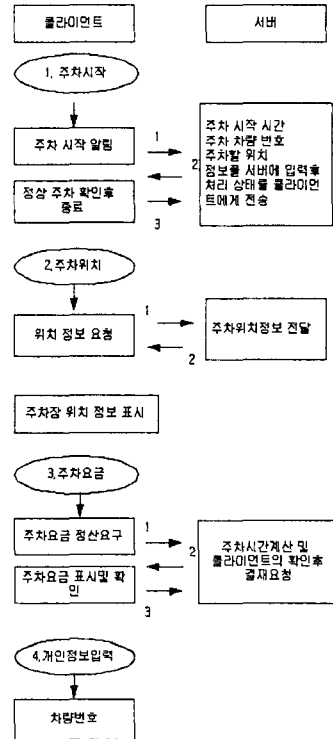
주차입구에 차량감지 스위치를 통하여 주차 상태를 알려준다. 주차 위치를 요구하는 신호가 입력되면 핸드폰으로 정보를 전송한다. 주차를 마치고 출차 할 때 주차 요금을 정산하기 위해서 출차 신호를 보내면 주차장 요금체계에 따른 요금을 계산하여 전송하고 주차시간과 요금을 데이터베이스에 저장한다.

5.1.2 흐름도

흐름도 1.

먼저, 관리화면의 선택할 수 있는 화면들이 나온다. 주차 하고자 하는 사람의 무선 인터넷 사용기기가 클라이언트가 되고 주차장을 관리하는 시스템이 서버가 된다.

클라이언트가 주차 관리 프로그램을 시동하면 초기에 4가지(주차시작, 주차위치, 주차요금, 정보입력) 선택 화면이 표시 된다. 주차가 시작됨을 클라이언트가 서버에게 알려주면 서버의 관리 프로그램은 주차 차량을 데이터베이스에 기록하면서 차량의 번호, 주차 시작 시간 등의 정보를 새 레코드에 입력하고 서버는 주차할 곳을 지정하여 클라이언트에게 전송하면 클라이언트는 주차를 시작함을 알리게 된다. 두 번째 주차 위치의 경우를 선택하여 전체도면과 차량에 가까운 지형을 표시하고 위치를 확인할 수 있다. 주차요금을 선택하면 클라이언트는 주차 종료임을 서버에게 알려주고 서버는 종료시간을 주차시간과 차이를 계산하여 요금으로 환산하여 클라이언트에게 정보를 알려준다. 결제될 주차 요금을 전달 받은 클라이언트는 확인 버튼을 눌러서 결제 처리를 요구하게 되고, 결제요구에 대한 처리가 결제DB의 승인 처리 과정을 거쳐서 정상 처리가 되면 확인 신호를 클라이언트에게 알려준다. 이과정이 이루어지는 동안 관리자는 웹을 통하여 차량의 주차 처리상태 및 위치 등을 수정, 삭제 할 수 있다. 물론 무선 인터넷을 사용하지 않는 차량에 대하여는 관리자 프로그램에서 주차 차량의 정보를 직접 입력하여 처리할 수 있다.

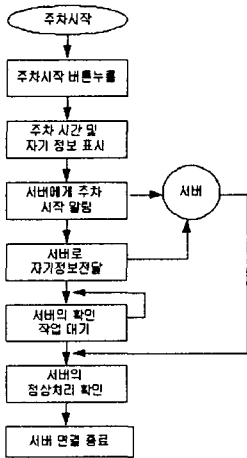


흐름도1. 주화면 흐름도.

Flowchart1. Main display flowchart.

흐름도2.

주차시작을 선택하면 주차하는 시간과 차량 번호를 화면에 표시하여 시작하는 차량의 정보를 주차하는 사용자가 확인할 수 있게 한다. 클라이언트의 주차시작 버튼이 눌리지면 서버에게 차량이 주차한다는 정보를 알려주고 정보의 전달을 간략화하기 위하여 차량번호만을 서버에게 주어서 주차함을 알려준다. 서버가 데이터베이스에 차량정보와 주차 시작 시간을 입력하는 동안 클라이언트는 대기상태로 있으며 서버에서는 클라이언트에게 예상 주차 위치를 전송하고 서버의 등록정보 입력 작업이 종료 되어 승인신호를 주면 클라이언트는 주차가 시작되며 작업이 종료된다.



흐름도2 . 주차시작흐름도(클라이언트).

Flowchart2 . Start parking flowchart(Client).

흐름도3.

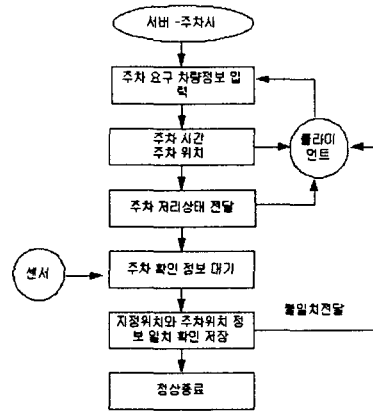
서버는 주차관리를 요청하는 클라이언트의 정보를 받아들여서 자료를 데이터베이스에 요청 순서에 따라서 입력하고 처리결과를 클라이언트에게 알려준다. 주차하고자하는 차량이 서버에서 지정한곳에 주차여부를 외부 센서입력을 통하여 확인하여 못하는 경우에는 예상 주차 지점이 아닌 경우 알려주어 주차 지역을 클라이언트가 수정 할 수 있게 한다.

흐름도4.

클라이언트가 주차한곳의 위치정보를 서버에게 요청하면 전체 주차장 지형도와 세부구역 지형도를 구분하여 선택하게 함으로서 위치를 찾는데 용이하게 한다.

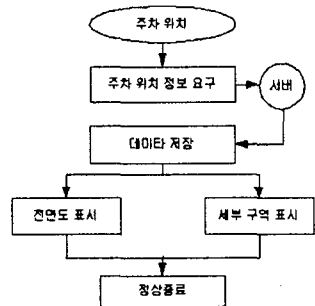
흐름도5.

클라이언트는 주차요금을 선택함으로써 서버에게 차량이 출차 한다는 상황을 알려주면 서버는 현재시간과 주차를 시작한 시점의 시간을 총 주차시간으로 계산하여 결제 예상금액을 클라이언트에게 전달한다. 클라이언트가 금액 확인 후 결제를 요청하면 서버에서 결제 DB로 결제자 정보를 전달하여 주차요금 결제처리를 수행하고 결제가 정상 종료되면 데이터베이스에 정상처리를 기록한 후 클라이언트에게 결제가 정상처리 됨을 알려준다.



흐름도3 . 주차시작흐름도(서버)

Flowchart3 . Start parking flowchart(Server).

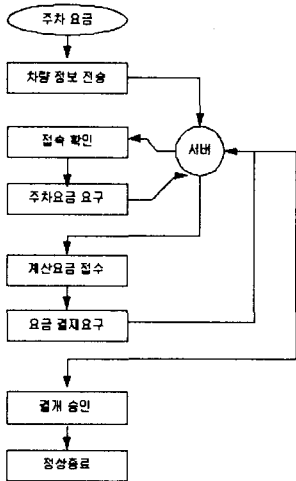


흐름도4 주차위치(클라이언트)

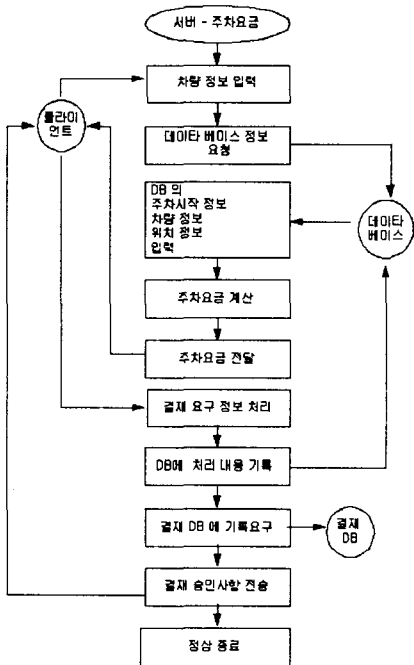
Flowchart4 . Parking Location (Client).

흐름도6.

클라이언트에서 오는 결제요청에 대하여 서버는 데이터베이스에 저장된 입차 시간과 출차 요청한 시간을 분단위로 계산하여 저장하여 두고 주차시간에 따른 요금을 전송하여 클라이언트에게 알려주고 클라이언트가 결제에 확인을 하면 결제 DB로 자료를 전송하여 처리를 종결한다. 결제 DB의 승인이 떨어진 사항에 대하여 클라이언트에게 알려준다.



흐름도5 . 주차요금흐름도(클라이언트)
Flowchart5 . Parking fee flowchart(Client).



흐름도6 . 주차요금흐름도(서버)
Flowchart6 . Parking fee flowchart(Server)

5.2 mysql 데이터베이스 구조

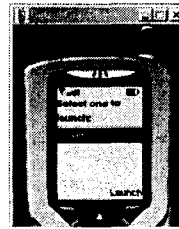
초기 기본 데이터베이스 테이블을 작성하기 위한 mysql(ver.4.0.1) 파일을 작성한다.

<표2> times1 table 의 구조
<Table2> time1 table structer

FIELD	TYPE	설 명
parknum	INT	primarykey(parknum)
tail	varchar(10)	차량번호
startt	datetime	주차시간
stopt	bigint	출차시간
loc	varchar(10)	주차위치
process	varchar(2)	처리상태

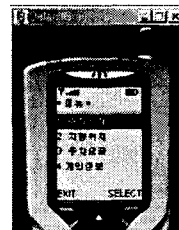
6. 구현 결과

휴대용 전화기 대응으로 SUN사의 Wireless Toolkit을 사용한 에뮬레이터로 구현



[그림6] 시작화면
[Fig6] Start Display

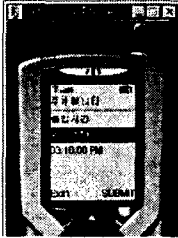
접속을 시작하면 그림6 과 같이 선택을 할 수 있게 하는 화면이 표시되고 Launch버튼을 누르면 기능들을 보여주는 그림7 의 화면으로 전환된다.



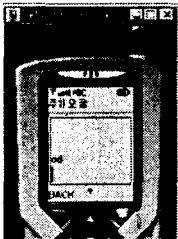
[그림7] 전체 메뉴화면
[Fig7] Full Menu Display

전환된 화면은 4가지 기능을 표시하고 SELECT버튼을 눌러 줌으로서 선택이 된다.

"1주차기능"을 선택하면 현재의 출입시간이 표시되고 제출 버튼을 눌러주면 서버로 자료를 전송하고 승인을 기다린다.



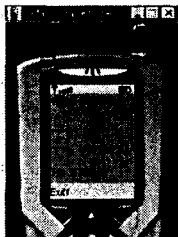
[그림8] 주차시작선택화면 (출입시간을 표시)
[Fig8] Selected Parking start Display



[그림9] 주차 시작확인(그림8의 SUBMIT 선택으로)

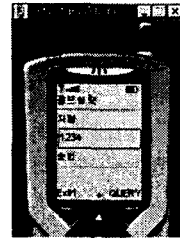
[Fig9] Parking start confirm (SUBMIT selected in Fig8)

서버에서 정상적인 접수가 이루어지면 그림 9 와 같은 회신이오고 정상주차임을 알려준다.



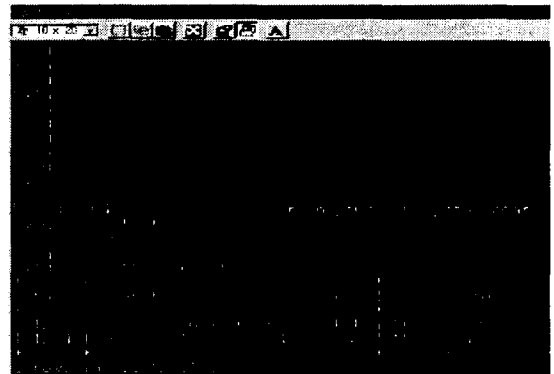
[그림10] 차량 위치화면
[Fig10] Parking location Display

차량 위치를 선택하면 그림10과 같은 주차장의 위치를 선택하여 볼 수 있게 하는 화면이 표시된다.



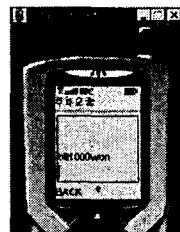
[그림11] 주차 요금 확인 화면
[Fig11] Parking Fee confirm Display

주차요금 버튼을 선택한 경우 그림11 이 나오고 조회 버튼을 선택하면 서버로 정보를 전송하여 주차가 끝났음을 알려주고 계산요금을 기다린다.

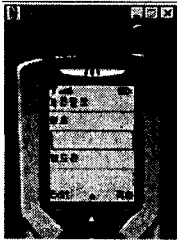


[그림12] mySQL의 데이터 저장
[Fig12] Data storage in mySQL

그림12의 mySQL DB자료에서 요금을 받아서 그림13과 같이 금액을 확인하여 버튼을 누르면 서버에서 요금을 처리 한다.



[그림13] 주차요금 계산표시화면
[Fig13] Parking Fee Result Display



[그림14] 개인정보화면 입력 초기화면
[Fig14] Person Information input Start Display

개인정보를 처음에 입력하고 저장하여 데이터를 갖고 있으면서 필요시 이용한다.

7. 결론

휴대용 전화기와 PDA와 같은 모바일 단말기의 보급의 확산으로 모바일 단말기와 인터넷을 접목한 무선 인터넷 서비스 분야가 새롭게 대두되고 있다. 현재 유선을 이용한 접속이 일반적이지만 향후 모바일 단말기를 이용한 인터넷 접속이 보편화될 것이다. 양방향 페이지, 휴대용 전화기, PDA 등으로 대표되는 모바일 단말기들은 기존의 데스크 탑과 노트북으로 대표되는 컴퓨팅 환경에 커다란 혁명을 불러 올 것이다.

주차장이란 한정된 모델에서 모바일 클라이언트 시스템의 구현이 가능함을 보임으로서 향후 더 나은 콘텐츠 개발이 예상된다. 위치정보 모바일시스템은 장소와 시간에 제약을 받지 않고 이용가능하다는 큰 이점 때문에 향후 실질적인 서비스를 위해 차별화된 모바일 콘텐츠의 개발과 효율적인 콘텐츠 제공을 위한 클라이언트 인터페이스 설계필요, 모바일 단말기의 약점인 한정된 자원의 관리 필요며, 이런 기반 여건이 성숙하면 새로운 개념의 통신 문화가 다가올 것으로 예상된다.

참고 문헌

[1] Alfred C. Weaver, "Monitoring and Control Using the Internet Java"

[2] 정보통신부 "무선인터넷 이용현황 및 실태 조사 보고서" 2002. 10. page21 .

[3] 변상규, 김한주 "국내 무선인터넷 서비스 이용실태 조사 분석" 권호:1029 2002.01.08

(<http://www.itfind.or.kr>)

[4] Eric Giguère , "Java 2 Micro Edition", WILEY,2000

[5] Jeff Nelson, "JAVA SERVER Page", O'REILLY,2001

[6] Round Ashri ,Steve Atkinson etc. "Professional Java Mobile Programming", Wrox,2001

[7] Yu Feng & Dr. Jun Zhu, "Wireless Java Programming with J2ME", SAMS , 2001

[8] 하동욱, "모바일 콘텐츠 활성화의 선봉장 J2ME: 임베디드 분야에서 자바와 닷넷 ", 소프트뱅크미디어,2001

[9] The Source for Java (TM) Technology (<http://java.sun.com>)[10]

<http://www.mobilejava.co.kr>

[11] 장미경 "휴대폰 위치기반 서비스, LBS" 과학동아, 2002

[12] "J2MEWTK를 사용한 MIDlet Httpconnection 프로그래밍" , (2001.09)

<http://www.mobilejava.co.kr>

[13] Eric Giguere , "Java2 Micro Edition" ,WILEY ,Ch3 ,2002

이창희



1985년 2월 : 광운대학교
전자공학과 졸업 (학사)
1987년 8월 : 광운대학교
산업정보대학
전자공학전공 졸업(석사)
2002년 2월 : 광운대학교
대학원 전자공학과 박사수료

<주관심분야> 무선 이동통신 시스템

전형준



1993년 2월 광운대학교
전자공학과
1995년 2월 광운대학교
전자공학과 대학원 공학석사
2003년 2월 광운대학교
전자공학과 대학원 공학박사

1995년 3월~1998년 3월
(주)대양전기공업주식회사

1998년 5월~2000년 12월

(주)창원정보통신

2001년 2월~현재,

삼성에스원 기술연구소

<주관심분야>M/W, RF 회로,

무선 네트워크 시스템

나극환



1981년 7월 : 블란서
ENSEEIH트 국립
종합공과대학 졸업(박사)

1981년 9월~ 현재 :

광운대학교
전자공학과 교수

1987년 1월~1988년 1월: 미국 COSMST

연구소 객원 연구원

<주관심 분야> RF 및 M/W, 밀리미터파통신
시스템 기술, 선형화 및 출력
최적화 기술, 수치 해석