

EJB기반 모바일 전자상거래 컴포넌트의 설계 및 구현

신 동 규[†] · 신 동 일[†] · 차 석 일^{††} · 장 철 수^{†††}
이 경 호^{††††} · 김 중 배^{†††††}

요 약

컴포넌트 기술은 재사용성을 기반으로 어플리케이션 개발의 생산성을 높일 수 있어 소프트웨어를 신속하고 효과적으로 개발할 수 있는 기술로써 채택되고 있으며, 이들 중 EJB 컴포넌트는 자바를 기반으로 하며 산업계에서 널리 이용되고 있는 기술이다. 본 논문에서는 모바일 전자상거래 시스템의 구성 요소중 게시판과 쇼핑몰을 EJB 컴포넌트로 설계, 구현함으로써 재사용성과 상호운용성을 지원하는 구축 방안을 제시하였다.

Design and Implementation of EJB-based Components for Mobile Commerce

Dongkyoo Shin[†] · Dongil Shin[†] · Sukil Cha^{††} · Choulsoo Jang^{†††}
Kyoungho Lee^{††††} · Joongbae Kim^{†††††}

ABSTRACT

Since component-based development technologies support reusability and enhance productivity of application development, they are adopted for rapid and efficient software development, and among them, EJB component is a Java-based software development technology widely used in industry. In this paper, we designed and implemented bulletin board and shopping mall which are major constituents of mobile e-commerce system using EJB components, and suggest construction guideline supporting reusability and interoperability.

키워드 : EJB(Enterprise Java Beans), 모바일 전자상거래(Mobile Commerce), WAP(Wireless Application Protocol), 무선 인터넷(Wireless Internet)

1. 서 론

무선 인터넷을 활용한 전자상거래는 전 세계적으로 큰 폭의 성장을 보일 것으로 예상되며, 이외에도 무선 인터넷을 이용한 서비스는 매우 다양하다[1, 2]. 그러나 효율적인 모바일 전자상거래를 위해서는 전자상거래 플랫폼에서 모바일 콘텐츠를 효과적으로 제작할 수 있는 환경을 고려하여야 하며 컴포넌트 기술이 그 대안 중 하나이다. 컴포넌트 기술은 소프트웨어를 신속하고 효과적으로 개발할 수 있는 기술로써 각광받고 있으며, 이 중 EJB(Enterprise Java Beans)는 자바를 기반으로 한 소프트웨어 개발에 효과적이어서 산업현장에서 많이 이용되고 있다[3].

모바일 전자상거래 시장은 오는 2003년을 분기점으로 PC

를 통한 기존 인터넷 사용자를 능가할 전망이다. 전 세계적으로 볼 때 향후 2004년경에는 미국을 제외한 나머지 국가에서는 무선 인터넷서비스나 이동 컴퓨팅 서비스의 이용자가 인터넷 이용자를 넘어설 것으로 보인다[4]. 현재 모바일 전자상거래 시장은 일본이 가장 앞서있으며, 다른 지역은 시작단계인 실정이다. 국내 무선 인터넷서비스 시장은 무선 인터넷관련 기술적 인프라가 구축되고 단순한 형태의 서비스가 도입되는 단계이다.

컴포넌트 기술은 소프트웨어를 신속하고 효과적으로 개발할 수 있는 대안으로 1990년대 초반부터 각광을 받기 시작하였다. 컴포넌트 기반 기술을 사용하여 어플리케이션을 개발할 경우에는 컴포넌트 기반 기술의 특징인 재사용성을 기반으로 어플리케이션 개발의 생산성을 높일 수 있어서 기하급수적으로 증가하고 있는 소프트웨어의 수요를 충족할 수 있게 된다. 따라서 컴포넌트 기술을 활용하는 컴포넌트 기반 개발 방법론이라는 새로운 패러다임이 미래 소프트웨어 산업에 있어서의 새로운 대안으로 인식되며 많은 사람들의 관심을 모으고 있다[3].

† 중신회원 : 세종대학교 컴퓨터공학과 교수
†† 준회원 : 세종대학교 대학원 컴퓨터공학과
††† 정회원 : 한국전자통신연구원 연구원
†††† 정회원 : 한국전자통신연구원 책임연구원
††††† 정회원 : 한국전자통신연구원 컴퓨터소프트웨어기술연구소
모바일응용서버연구팀장
논문접수 : 2002년 1월 10일, 심사완료 : 2002년 5월 22일

컴포넌트 기술은 규격화되고 검증된 소프트웨어 컴포넌트들의 획득과 조립을 통해 소프트웨어 시스템을 개발하는 기술이다. 소프트웨어 컴포넌트는 재사용성과 상호 운용성이 뛰어난 소프트웨어 부품으로서 애플리케이션 개발자는 마치 하드웨어를 조립하듯이 원하는 기능이나 성능을 가진 소프트웨어 컴포넌트를 선택하여 컴포넌트 시스템을 구성할 수 있게 된다. 이렇게 개발된 컴포넌트 기반 시스템은 일부의 부품을 개선된 부품으로 대체함으로써 품질이나 성능을 개선할 수도 있고 새로운 운용환경에 쉽게 이식할 수도 있다. 따라서 기술 변동에 능동적으로 대응할 수 있으며 타 시스템과의 상호 운용성도 보장된다. 이와 같이 컴포넌트 기반의 소프트웨어 개발은 다양한 측면에서 많은 이점을 제공해 주기 때문에 선진국뿐 아니라 국내에서도 빠르게 성장하는 분야가 될 것이다[5].

현재의 모바일 전자상거래 시스템은 유사한 시스템을 요구함에도 불구하고 개발자의 요구에 따라 새롭게 제작되고 있다. 이러한 방식은 자원과 인력에 커다란 낭비를 초래하게 된다. 무선 전자상거래 기술의 기반 하에 컴포넌트기술이 소프트웨어의 부품화, 분산화, 통합화 및 개방화를 지원하게 되면 효율성이 증가 될 것이다. 본 논문은 체계화된 모바일 전자상거래 시스템의 형식을 제안하고 그 시스템을 컴포넌트로 제작하여 재사용성을 증가시켜 자원과 인력의 낭비를 감소시키는 데 목적을 둔다. 모바일 전자상거래 컴포넌트에 대한 연구는 국내에서는 미개척 분야이고 국외의 경우도 기술 개발 진행정도가 상품화단계에 있지 않기 때문에, 모바일 전자상거래 관련 연구의 활성화 및 시장에서의 제품 적용에 기여할 수 있을 것으로 기대된다. 본 논문에서는 모바일 전자상거래 시스템의 특성에 맞게 개발자가 적용할 수 있는 시스템 컴포넌트를 설계하고 구현하였다.

2. 관련 연구

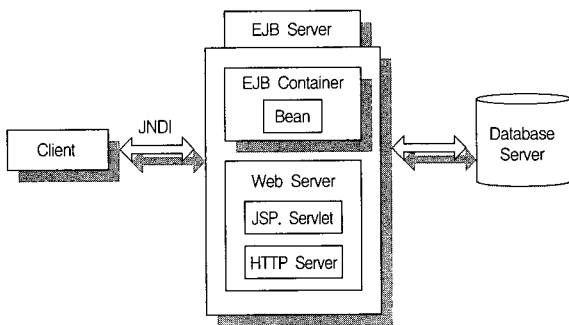
무선 단말기를 통한 인터넷 접근은, 시간과 장소에 구애받지 않는 정보 접근의 가능성을 제시하지만 단말기 자체의 제한, 즉 작은 스크린, 저용량의 프로세서와 메모리등과 무선망의 성능상의 제한으로 인해 기존 유선 인터넷에서와 같이 멀티미디어를 포함한 다양한 형태의 정보 제공이 어렵다. 따라서 무선 통신의 기본적 성격인 이동성(mobility)과 휴대성(portability)에 적합한 서비스를 생각할 수 있다. 무선 인터넷에서 제공되는 서비스는 크게 다음과 같은 3가지의 종류로 구분할 수 있다[6]. 첫째는 전자우편, 팩스 등을 포함하는 일반적인 개인정보 관리 및 통신 서비스로서 이 분야의 서비스는 유럽, 미국, 일본에서는 대단히 중요한 서비스로 부각되어있다. 이동통신 사업자는 중앙에서 사용자의 개인정보 데이터베이스를 관리하여 데이터의 무결성과 동기성의 보장을 한다. 즉, 업무상 필요한 사람들의 주소록과 개인정보를 이동통신 사업자가 관리하는 것이며, 개

인정보 데이터베이스를 중앙에서 집중 관리하게 됨에 따라 가입자의 이탈방지 및 매출증대를 유발하게 된다. 둘째로 여러 가지 부가 정보를 문자로 제공하는 문자정보 서비스가 있다. 간단한 예로는 교통, 여행, 뉴스, 기상, 연예 등의 정보제공 서비스와 단말기 상호간의 문자통신 서비스를 들 수 있다. 음성통신이 아닌 무선 데이터를 가지고 새로운 수익의 원천을 창출하고자 하는 통신사업자에게 있어서 이 부분은 중요하며, 대부분의 이동통신 사업자는 이 부분을 강화하고 있는 현실이다. 현재 웹에서 이용 가능한 다양한 종류의 콘텐츠를 무선에서도 이용할 수 있게 되며, 특히 단말기 상호간 인사말교환, 동호인 그룹간 그룹 단말 전송 등이 젊은 사용자 계층으로 넓게 퍼지고 있다. 셋째는 무선 인터넷의 핵심인 모바일 전자 상거래부분이다. 무선이동 통신서비스의 이동성을 이용하여 언제, 어디서나, 누구에게나 전자상거래를 할 수 있는 수단을 제공한다. 이 분야의 서비스는 주가 정보 제공, 주식거래, 은행 잔고 조회 및 예금 이체, 전자지갑 등을 포함한다. 전 세계적으로 무선을 이용한 무선 뱅킹과 무선 상거래가 확대되고 있는 추세이며, 이는 모바일 전자상거래가 공간상 제약을 극복하고 현대사회의 급변하는 생활습관에 잘 부합되기 때문으로 분석되고 있다.

모바일 전자상거래의 구축에는 무선 인터넷 서비스에 이용될 수 있는 여러 가지 기법들이 이용 가능하다. 기존 인터넷과 이동단말기 사이의 단말의 이동, 하드웨어의 제약, 무선 매체가 가지는 고유한 특성들을 증대하는 프록시를 이용하는 기법들을 이용할 수 있으며[7-11] 프록시 구조에 유연성을 주고 활용성을 높이는 에이전트 형태를 구축하는 방법도 이용할 수 있다[10]. 무선 인터넷 서비스를 위한 정보의 표현 및 전달에 있어서 무선 인터넷 고유의 방법론을 채용하려는 시도와[11], 기존의 인터넷에서 사용하는 방법론을 단순히 확장하여 적용하려는 시도로 진행되고 있다[7-10]. 현재 전 세계적으로 무선 인터넷 고유의 기법인 WAP(Wireless Application Protocol)을 이용하여 모바일 전자상거래를 포함하는 각종 무선 인터넷 서비스가 제공되고 있다[11].

선(Sun)사에 의해 제안된 EJB(Enterprise Java Beans)는 대규모 프레임워크를 위한 자바기술로써 분산 환경하에서 어플리케이션을 개발, 배포, 실행하기 위한 아키텍처이다[12-15]. 분산 환경하의 어플리케이션들은 트랜잭션 관리, 보안, 데이터베이스 제어 등과 같은 시스템 레벨의 서비스를 필요로 하며, J2EE(Java 2 Enterprise Edition) 플랫폼은 이러한 서비스 등을 제공하여 사용자가 비즈니스 로직에만 전념할 수 있도록 다양한 기능을 제공하고 있으며 전 세계 여러 곳의 전자상거래 사이트에 적용되어 운영되고 있으며 모바일 전자상거래 시스템의 구축에도 효과적으로 적용이 가능하다[12-14]. EJB에서는 빈(Bean)을 기본적인 컴포넌트 단위로 정의하고 있다. 빈에는 두 종류가 있는데 세션 빈(Session Bean)과 엔티티 빈(Entity Bean)이다. 엔티티 빈

이란 영속성(Persistence)을 가지는 데이터를 엔터프라이즈 빈으로 표현한 형태이다. 즉, 영속적인 기억장소에 저장된 데이터를 표현하는 엔터프라이즈 빈이다. 엔티티 빈이 영속적인 기억장소에 저장된 데이터를 비즈니스 객체 즉, 엔터프라이즈 빈 컴포넌트로 추상화한 것이라면, 세션(Session) 빈은 이러한 데이터를 사용하는 비즈니스 로직(비즈니스 프로세스)을 정의하는 엔터프라이즈 빈이다. 즉, 엔티티 빈은 다중 클라이언트에 의해 공유되는 데이터를 표현하는 반면, 세션 빈은 이러한 공유데이터를 사용하는 워크플로우를 표현하는 빈이다. EJB 시스템은 (그림 1)과 같이 구성되며 각 구성요소에 대한 설명은 아래와 같다[12].



(그림 1) EJB시스템의 구조

• EJB Server

EJB Server는 EJB Container를 하나 혹은 그 이상을 포함하는 형태로, 클라이언트의 요청을 처리하는 역할을 담당하고 있다. 3-tier 시스템에서 애플리케이션 서버의 역할을 담당하고 있으며, 클라이언트가 서버에 어떤 요청을 하게 되면, EJB Server는 이 요청을 분석하여, 해당요청에 적당한 처리를 하게 된다. 클라이언트가 html을 요청하게 되면, EJB Server는 사용자의 요청을 웹서버에 전달하여 적절한 웹 서비스를 실행하게 되고, EJB가 필요한 요청을 하면 이 요청을 다시 EJB Container에 전달하게 된다. EJB서버는 이러한 서비스를 위해서 JNDI를 통하여 프로그램을 서비스와 EJB 컴포넌트에 연결해주며, 사용자 인증관련기능도 제공한다.

• EJB Container

Enterprise Beans는 EJB Container에서 인스턴스화된 후 서비스를 제공하게 되는데, 이를 위해 EJB Container는 트랜잭션 모니터(Transaction Monitor), 보안(Security), 원격 클라이언트접속(Remote Client Connectivity), 빈의 라이프 사이클 관리(Life Cycle Management), 데이터베이스 Connection Pooling 등을 담당한다.

• Java Naming and Directory Interface(JNDI)

Enterprise Bean는 자동적으로 ENC(Environment Naming Context)라고 하는 명명 시스템을 가지는데, ENC는 EJB Container에 의해 관리되며, JNDI는 ENC를 통해 네트워크

상 혹은 다른 Container 서비스에서 EJB 컴포넌트들을 찾아서 연결해 주는 역할을 수행한다.

• EJBHome

빈의 생명주기와 관련된 부분으로 빈을 생성하거나 소멸시키는 역할을 수행하며, 클라이언트는 EJB를 얻기 위해서 먼저 EJBHome객체를 얻어야 한다. 클라이언트는 JNDI lookup을 통해서 EJBHome객체를 얻으며, Home interface를 통해서 EJBHome과 인터페이스한다. 이것은 빈 개발자에 의해서 정의된다. 또한 클라이언트는 EJBHome 객체를 통해서 새로운 빈을 생성할 수 있다.

• EJBObject

EJBHome객체에 의해서 생성된다. 클라이언트에서 빈에 정의된 메소드를 호출하면 이것은 EJBObject로 전달된다. EJBObject는 컨테이너와 상호작용을 통해서 실제 빈 구현 클래스에 존재하는 메소드를 호출한다. Remote interface라는 것을 통해서 EJBObject와 인터페이스한다. Remote interface는 개발자가 정의하며 빈의 메소드가 정의되어 있다.

• Java Transaction API/Java Transaction Service (JTA/JTS)

Enterprise Java Beans에서 JTA/JTS를 트랜잭션 API로 사용하고 있다. Enterprise Bean Provider는 JTA를 사용하여 저수준의 JTS를 컨트롤 할 수 있으며, Deployer 역시 배포시에 EJB Component의 트랜잭션 속성을 정의 할 수 있다.

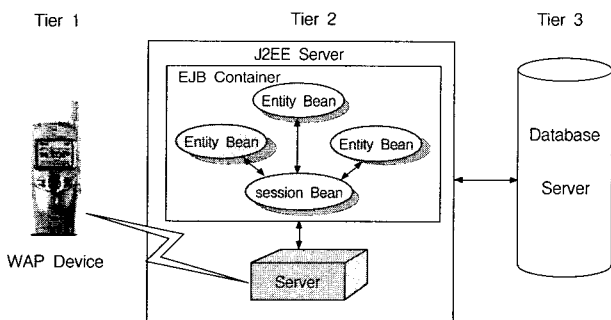
EJB를 사용함으로써, 복잡한 분산 객체 프레임워크에 대한 작성 없이 확장성과 신뢰성이 높고 안전한 어플리케이션의 작성이 가능하게 된다. EJB는 어떤 회사의 엔터프라이즈 미들웨어에서도 이동성 있고 재사용이 가능한 어플리케이션을 지원하도록 설계되었다. EJB의 목적에는 여러 가지가 있다. 시스템 수준의 세부적인 기능을 숨김으로써, 어플리케이션의 적성을 용이하게 하며, 한번 작성된 EJB 어플리케이션은 다른 환경에서도 바로 실행되게 한다. 따라서 구현된 엔터프라이즈 EJB 컴포넌트는 복수의 플랫폼에서 설치되어 사용될 수 있다. 또한 개발과 전개, 그리고 전사적인 어플리케이션의 생성 주기에 관한 실행 특성들을 제시한다. 한편 EJB는 자바 API와의 호환을 지원 할 뿐만 아니라, 엔터프라이즈 자바 빈과 자바 외의 언어로 구현된 어플리케이션과의 상호연동 또한 지원한다. 또한 CORBA 프로토콜과의 호환도 지원한다. EJB의 컴포넌트 아키텍처는 컴포넌트의 표준적인 구축에서부터 관리 및 유지 보수까지의 전반적인 컴포넌트 개발을 용이하게 지원하기 위하여 다음의 3가지 요소들로 구성된다. 먼저 컴포넌트를 개발하기 위한 틀과 배치된 컴포넌트를 관리하기 위한 컨테이너, 그리고 컴포넌트를 배치시키고 유지 관리하는 톨로서, 이들

각 요소들은 컴포넌트를 제작하는 과정에서 필수적으로 요구되는 부분이다[12-14, 23, 24].

3. M-Commerce를 위한 컴포넌트 설계

현재의 중/대형 규모 소프트웨어 개발의 여러 특징 중의 하나는 아키텍처 기반이라는 점이다. 매우 다양한 네트워크, 다양한 컴퓨터 자원들을 포함한 광범위한 분산 컴퓨팅 환경에서, 필요한 모든 소프트웨어를 처음부터 개발하는 것은 어려운 작업이 된다. 기존의 라이브러리, 프레임워크, 컴포넌트를 포함한 다양한 리소스들뿐만 아니라, 여러 프로젝트를 통한 경험/전략까지도 재사용을 하는 것이 필수적이다[14].

무선 인터넷에서도 예외가 아니다. M-Commerce 시스템을 EJB를 사용해서 구현함으로써 EJB의 장점을 그대로 상속받는다. 본 논문에서 설계한 M-Commerce 시스템은 (그림 2)와 같은 구조를 가진다. 클라이언트(휴대용 단말기)로 서버에 접속을 하면, 클라이언트는 서블릿을 통하여 서버와 통신을 하고, 디스플레이될 기본 화면을 제공받게 된다. 사용자의 요구에 따라 메뉴 중에서 원하는 항목의 세션빈에 연결된다. 사용자는 세션빈과의 대화를 통해서 원하는 정보를 제공 받는다. 연결된 세션빈은 각각의 엔티티 빈들과의 통신을 DB를 제어 하고 이를 통해 사용자 정보의 저장과 삭제를 가능케 한다. 또한 세션 빈에서 실제적인 소비자와의 대화가 이루어진다. 대화를 통해 소비자가 원하는 일련의 기능들을 수행한 후 엔티티 빈과의 연결해서 DB와 연동된다.



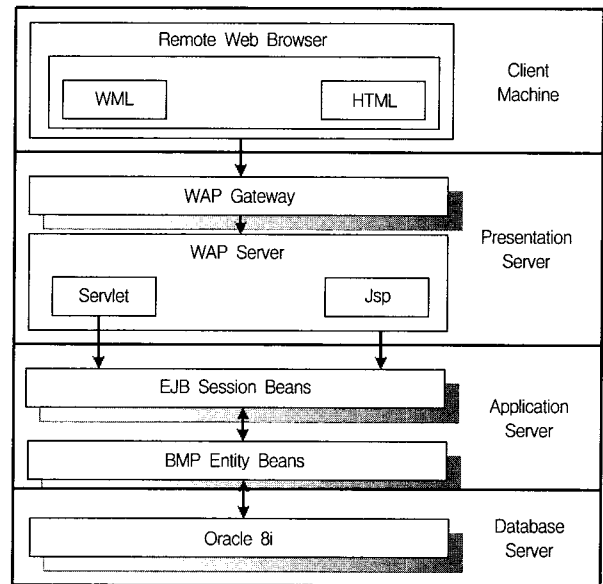
(그림 2) M-Commerce 시스템

본 논문에서 EJB로 설계 구현하는 M-Commerce 시스템의 기본적인 구조는 (그림 3)과 같다. 각 구성 요소의 기능을 살펴보면 다음과 같다.

- **클라이언트(Client)** : 클라이언트는 웹 브라우저나 모바일 폰(Mobile Phone)과 같은 무선 장비상의 다양한 Markup Language 브라우저를 통해 주로 UI(User Interface)를 담당하며, Applet 또는 JAVA application 형태일 수도 있다.
- **프리젠테이션(Presentation)** : JSP, Servlet 응용 프로그램으로 클라이언트의 요구를 받아 애플리케이션 층에 전달하고 수행결과를 받아 다양한 클라이언트 측 응용 프

로그래밍의 요구에 맞게 결과를 전달한다.

- **애플리케이션(Application)** : 비즈니스 로직을 포함하고 있는 엔터프라이즈 자바 빈즈(Enterprise Java Beans, EJB) 컴포넌트들의 집합이다.
- **데이터베이스(Database)** : 애플리케이션 서버와 연결을 통해 자료를 저장하는 역할을 한다.



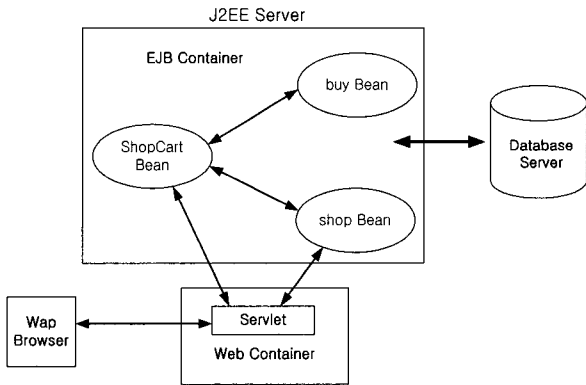
(그림 3) M-Commerce 시스템의 기본 구조

3.1 시스템의 기본 구조

전자상거래상에서 쇼핑물은 가장 기본적인 전자상거래 사업 모델로서 기업이나 점포의 웹사이트를 이용한 마케팅 모델이다. 처음에는 기업홍보, 제품홍보를 위해 사용하다가 점차 고객으로부터 주문도 받고 대금도 지불할 수 있는 기능을 첨가하게 되었다[16-18]. 판매 기업 입장에서 기존의 마케팅 채널과 다른 또 하나의 마케팅 채널로서 저렴한 비용으로 전 세계로 시장을 확대함으로써 수요를 확충하여 매출 증대를 가능케 하고, 고객입장에서선 전통적 채널보다 저렴한 가격, 폭 넓은 선택기회, 보다 많은 정보, 시간과 공간을 초월한 구매에서 대금지불절차까지의 편리함을 제공함으로써 전자상거래에 참여하는 동기를 부여한다. 특히 한번 구매했던 고객의 경우 고객정보를 구축할 수 있어 이를 이용해 재구매시 1대1 개인화 된 마케팅을 통한 편익을 제공하고 있으며, 소위 데이터베이스 마케팅(DB 마케팅)이나 고객관리(Customer Relationship Management)의 가장 기본적인 기능을 제공한다. 또한 게시판은 유/무선 인터넷상에서 의사를 교환하는 가장 중요한 기능을 담당하고 있으며, 여론을 수집하거나 문의 사항을 전달하는 것으로 이미 유선 인터넷상에서는 널리 사용되고 있다. 그리고 쇼핑물과 동시에 구축되어 사용자의 편의를 돕는 방법으로 널리 사용되고 있다[17, 18].

이러한 점을 감안하여 본 논문에서는 ‘쇼핑몰’과 ‘게시판’을 구현하였다. 전자상거래 시스템을 구축하거나 실제로 사용되고 있는 시스템에 가장 유용하게 사용할 수 있으며 쇼핑몰과 게시판을 컴포넌트로 구현함으로써 재사용성을 확대시켰다.

3.1.1 쇼핑몰 시스템

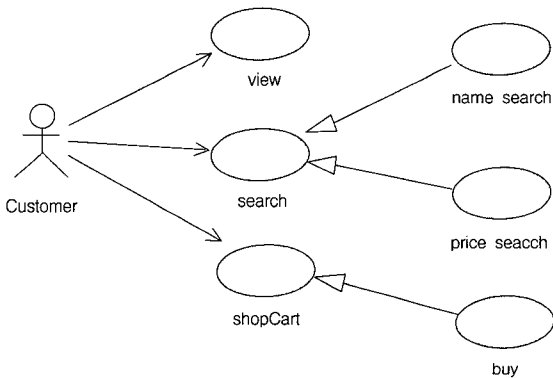


(그림 4) EJB 기반 쇼핑몰 시스템 구성도

인터넷 쇼핑몰이란 인터넷의 홈페이지를 이용하여 웹상에서 상품 카탈로그를 제공하거나 고객이 원하는 물건의 구매를 쉽고 빠르게 주문에서 배달까지 일괄적으로 처리할 수 있게 만들어 놓은 홈페이지를 말한다[17, 18]. 본 논문에서 구현한 쇼핑몰 시스템의 구조는 (그림 4)와 같다. UML (Unified Modeling Language)의 Diagram들을 이용하여 구성과 동작원리를 표현된다[19, 20].

가. 쇼핑몰의 기능

쇼핑몰의 기능은 Use Case Diagram을 통해서 알아본다. (그림 5)와 같이 쇼핑몰의 기본 기능은 상품을 보고 구매한다. 이 중 상품을 검색해서 원하는 물건을 찾을 수 있고, 이러한 검색은 이름을 통한 검색과 원하는 가격별로 검색하는 기능으로 구분한다. 구매는 물건들을 살펴 본 후 우선 자신의 shopCart에 저장하고, 저장된 물건 중에서 구입한다.

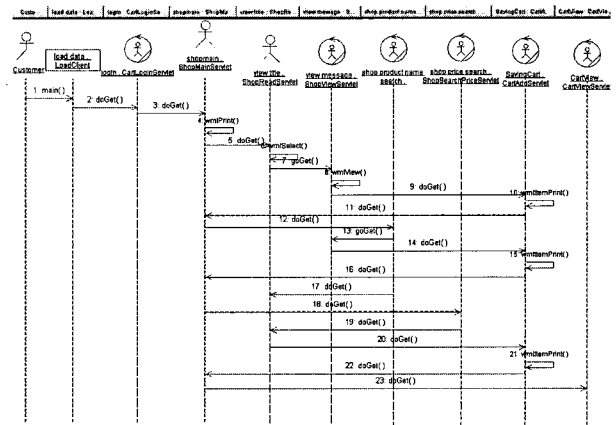


(그림 5) 쇼핑몰 UseCase Diagram

나. 쇼핑몰 시나리오

쇼핑몰의 흐름은 일어나는 이벤트들을 시간의 순서대로 나열해서 보여주는 Sequence Diagram을 통해서 쉽게 살펴본다. 쇼핑몰의 Sequence Diagram은 (그림 6)과 같다.

먼저 LoadClient 클래스가 할당된 load data객체를 통해서 상품들을 서버에 저장한다. 사용자는 login객체를 통해서 사용자별로 접속을 하고, ‘shopmain : ShopMainServlet’ 객체에 접속한다.



(그림 6) 쇼핑몰 Sequence Diagram

shopmain에서 상품리스트를 볼 것인지 아니면 원하는 상품을 검색 할 것인지를 선택한다. ‘view title : ShopReadServlet’ 메뉴를 선택하면 상품 리스트를 볼 수 있고, 사용자는 상품명을 보고 원하는 상품을 선택하면 ‘viewmessage : ShopViewSelect’를 통해서 상품에 관한 상세한 정보를 살펴본다. 또한 ‘shopProductNameSearch’나 ‘shopPriceSearch’를 통해서 이름이나 가격으로 물건을 검색할 수 있으며, 원하는 물건은 장바구니에 추가한 후 구매한다.

다. 쇼핑몰의 구성

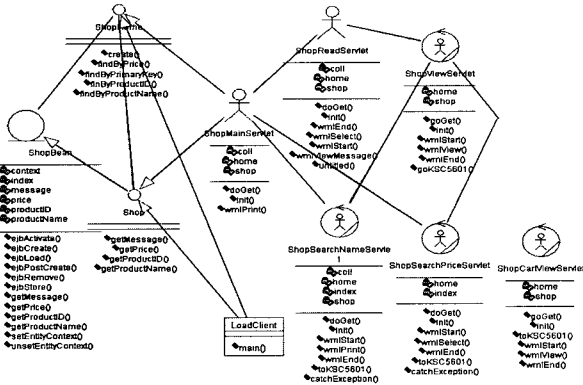
쇼핑몰은 ShopHome, Shop의 두개의 인터페이스로 이루어진다. ShopBean에서는 Shop 클래스의 일반화가 이루어진다. ClassDiagram을 통하여 각각의 클래스에서 사용되는 메소드와 속성 그리고, 각 클래스간 연관관계를 살펴본다.

쇼핑몰의 리모트 인터페이스인 Shop 클래스는 getProductname(), getMessage(), getProductID, getPrice()의 4개의 메소드로 이루어진다. 각각은 물건의 이름, 물건의 설명, 물건의 제조회사, 그리고 물건의 가격을 반환 받는다.

홈 인터페이스인 ShopHome 클래스는 5개의 메소드로 구성된다. 먼저 create에서는 빈의 생성을 위한 메소드이며 5개의 인자 값 즉, 물건 번호, 제조회사, 물건 이름, 상세 내용, 가격을 가진다. findByPrimaryKey는 쇼핑몰의 primary key로 지정되어 있는 물건 번호를 통해서 원하는 객체를 찾는 메소드이고, findByProductname는 물건이름으로 찾아내며, findByProductID는 제조회사로 찾아내며, findByPrice

는 두 개의 입력 값으로 원하는 가격대의 물건을 찾아내는 메소드이다.

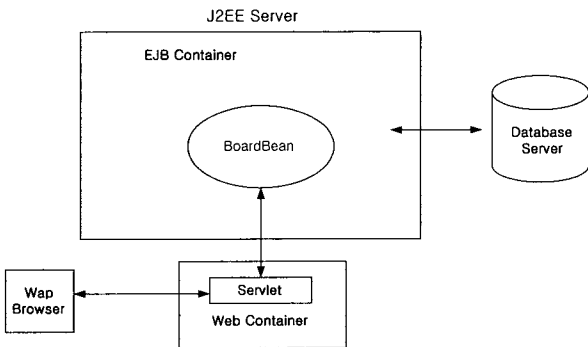
ShopBean 클래스의 ejbCreate는 빈의 생성을 담당하며, 5개의 인자값을 현재 값으로 바꾸는 역할을 한다. ejbActivate()에서는 활성화 상태에서, ejbPassivate()에서는 비 활성화 시, ejbRemove()는 제거 시, ejbPostCreate()에서는 생성 후의 기능에 대한 메소드이다.



(그림 7) 쇼핑물의 Class Diagram

3.1.2 게시판 시스템

게시판은 불특정 다수의 가입자를 대상으로 메시지를 작성할 수 있으며 게시판의 글을 읽을 수 있는 기능이다. EJB 기반 게시판 구조도는 (그림 8)과 같다.



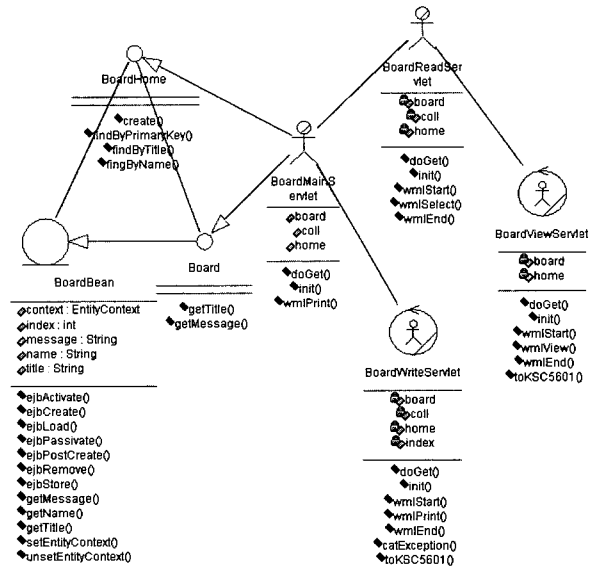
(그림 8) EJB 기반 게시판 구조도

가. 게시판의 구성

(그림 9)의 Class Diagram을 통해서 클래스의 구성을 살펴본다. 게시판의 리모트 인터페이스는 Board 클래스이다. getTitle()은 게시판의 제목을 반환한다. 사용자가 볼 수 있도록 제목을 보여줄 수 있는 기능이다. getMessage(), getTitle(), getName()은 각각 게시판의 내용, 선택된 제목별 내용, 게시판에 글을 쓴 사용자를 보여 준다.

BoardHome 클래스는 홈 인터페이스이며 findByPrimaryKey()에서 primary key로 검색하는 기능을 제공한다. 여기서 primary key는 int형 index이다. findByTitle과 findByName은 각각 제목으로 검색하는 기능과 이름으로 검색

하는 기능을 제공한다.



(그림 9) 게시판 Class Diagram

인터페이스들을 실제로 구현하는 것은 Bean 클래스이다. ejbCreate()는 실제로 Bean이 생성되는 부분이며 반환 값은 null이다. ejbActivate()에서 빈의 활성화 되었을 경우에 상황을 설정하고, ejbPassivate()에서 빈의 passivate 상태, 즉 빈이 사용되지 않을 경우에 시스템의 불필요한 자원의 낭비를 위해 설정을 최소화 하는 상태이다. ejbPostCreate()에서는 생성 후에 일어나는 일을 설정해 준다.

4. M-Commerce 컴포넌트 구현 및 실행결과

4.1 M-Commerce 컴포넌트 구현

4.1.1 쇼핑물의 구현

가. 비즈니스 메소드

(그림 10)과 같이 쇼핑물의 비즈니스 메소드는 5개의 메소드로 이루어진다. 여기서는 빈 클래스 내에 정의된 Persistent 필드의 값을 변경하거나 얻어오는 메소드를 정의한다. 각각을 살펴보면 getProductName()은 물건의 이름을 반환해 주고, getMessage는 물건에 따른 자세한 설명을 주고, getProductID()는 물건의 제조회사를 반환하며, getPrice()는 물건의 가격을 반환한다.

```
public interface Shop extends EJBObject
{
    public String getProductName() throws RemoteException ;
    public String getMessage() throws RemoteException ;
    public String getProductID() throws RemoteException ;
    public int getPrice() throws RemoteException ;
}
```

(그림 10) 쇼핑물 리모트 인터페이스

나. 쇼핑물의 라이프 사이클

쇼핑물의 라이프 사이클은 홈 인터페이스에서 정의한다. 쇼핑물의 홈 인터페이스는 (그림 11)와 같이 5개의 메소드로 구성된다. 메소드는 크게 create()와 finder메소드 두가지로 분류되며, create 메소드는 엔터프라이즈 빈의 리모트 인터페이스 타입을 반환한다. 홈 인터페이스에 정의된 메소드와 파라미터 타입과 개수가 일치하는 ejbCreate메소드를 엔터프라이즈 빈 클래스에서 구현한다. create()에서는 빈의 생성을 위한 메소드이며, 5개의 인자 값을 가진다. 각각은 물건 번호, 제조회사, 물건 이름, 상세 내용, 가격을 나타낸다.

findByPrimaryKey에서는 쇼핑물의 primary key로 지정되어 있는 물건 번호를 나타내며, findByProductName은 물건이름을, findByProductID에서는 제조회사를, findByPrice에서는 가격을 나타낸다.

```
public interface ShopHome extends EJBHome
{
    public Shop create(int index, String productId, String
productName, String message, int price) throws RemoteException,
CreateException ;
    public Shop findByPrimaryKey(int index)
throws FinderException, RemoteException ;
    public Shop findByProductName(String productName)
throws FinderException, RemoteException ;
    public Collection findByProductID(String productID)
throws FinderException, RemoteException ;
    public Collection findByPrice(int lowPrice, int highPrice)
throws FinderException, RemoteException ;
}
```

(그림 11) 쇼핑물의 홈 인터페이스

다. 쇼핑물의 수행

```
public String ejbCreate(int index, String productID, String productName,
String message, int price) throws CreateException
{
    if (productID == null)
        { throw new CreateException("You must specify a productID."); }
    this.index = index ;
    this.productID = productID ;
    this.productName = productName ;
    this.message = message ;
    this.price = price ;
    return null ;
}
public void setEntityContext(EntityContext context){ this.context =
context ; }
public void ejbActivate() { productID = String/context.getPrimaryKey() ; }
public void ejbPassivate()
{
    message=null ;
    productID = null ;
    productName = null ;
}
public void ejbRemove() {}
public void ejbLoad() {}
public void ejbStore() {}
public void unsetEntityContext() {}
public void ejbPostCreate(int index, String productID, String productName,
String message, int price) {}
```

(그림 12) 쇼핑물 빈 클래스

(그림 12)에서는 쇼핑물의 실질적인 수행을 보여준다. ejbCreate()부분은 홈 인터페이스에서 정의한 create 메소드의 파라미터 타입과 개수가 일치한다. EJB 클라이언트가 홈 인터페이스의 create()메소드를 호출하면 호출된 create()메소드에 상응하는 ejbCreate 메소드를 EJB 컨테이너가 호출하게 되고, 이 결과로 새로운 빈이 생성되게 된다. ejbActivate()에서는 활성화 상황에서, ejbPassivate()에서는 비 활성화시, ejbRemove()는 제거시, ejbPostCreate()에서는 생성 후의 기능에 대한 기능이다.

라. 쇼핑물의 장바구니

장바구니의 비즈니스 메소드는 getName(), setName(String name)이며 각각 목록을 저장하거나 가져오는 역할을 한다.

```
public String getName() {return name ; }
public void setName(String name) {this.name = name ; }
public String ejbCreate(int index, String name) throws createException
{
    if (name == null)
        {throw new CreateException("You must specify a
productID."); }
    this.index = index ;
    this.name = name ;
    return null ;
}
```

(그림 13) 장바구니 빈 클래스

(그림 13)에서는 장바구니에 대한 비즈니스 메소드들의 실질적인 수행을 보여준다. getName()은 호출된 곳에 이름을 반환하며, setName()은 이름을 저장한다.

4.1.2 게시판의 구현

```
public String ejbCreate(int index, String name, String title, String
message)
throws CreateException
{
    if (name == null)
        {throw new CreateException("You must specify a
productID."); }
    this.index = index ;
    this.name = name ;
    this.title = title ;
    this.message = message ;
    return null ;
}
public void setEntityContext(EntityContext context)
{this.context = context ; }
public void ejbActivate(){name = (String)context.getPrimaryKey() ; }
public void ejbPassivate()
{ message = name = title = null ; }
public void ejbPostCreate(int index, String name, String title,
String message) {}
```

(그림 14) 빈 클래스

(그림 14)에서와 같이 ejbCreate()에서는 생성을 하고 각

각의 변수 값에 현재 입력된 내용으로 내용을 변경한다. 반환 값은 null이다.

각각을 살펴보면, ejbActivate()에서 빈의 활성화되었을 경우에 상황을 설정한다. ejbPassivate()에서 빈이 passivate 상태, 즉 빈이 사용되지 않을 경우에 시스템의 불필요한 자원의 낭비를 위해 설정을 최소화하는 상태를 보여 준다. ejbPost Create()에서는 생성 후에 일어나는 일을 설정해 준다.

4.2 M-Commerce 컴포넌트의 실행

본 논문에서 구현한 컴포넌트는 프로그램의 이식성 및 이동성의 효과적인 제공을 위해 java언어로 작성되었다. EJB를 위한 환경으로는 SUN사에서 제공하는 J2EE패키지에서 기본적으로 포함되어 있는 서버와 DB와 Deploytool을 사용하였고, DB는 Cloudscape를 사용하였다.

모바일 환경을 전제로 구현을 하였기 때문에 모바일 환경에서 디스플레이 될 수 있도록 WML(Wireless Markup Langage)을 이용하여 제작하였다[11, 21, 22].

4.2.1 쇼핑물 실행 화면

가. 초기화면

무선 쇼핑물 환영합니다.	이름: 홍길동
시작 option	Menu alpha
1▶입력 2 취소/처음으로이동	홍길동님 환영합니다.
OK	확인 다시쓰기

- 무선 쇼핑물의 초기화면이다.
- 이름을 입력한다.

나. 상품 전체 보기

1▶상품전체 보기 2 상품검색 3 장바구니 보기	1 목장판 2 베개 3 보온컵 4▶목도리 5 시계
OK	상세보기 취소
제조사: 소왕 상품명: 목도리 특징: 울로 짠 목도리 가격: 38000원	장바구니에 목도리가 추가되었습니다.
확인 장바구니에추가	확인

- 상품 전체 보기에서 여러 가지 상품을 볼 수 있고, 상세보기를 누르면 그의 대한 가격과 리스트들이 나열된다.

- 장바구니에 추가 메뉴를 선택하면 장바구니에 추가된다.

다. 장바구니 보기

1 상품전체 보기 2 상품검색 3▶장바구니 보기	1▶목도리
OK	자세히보기 메뉴로이동
제조사: 소왕 상품명: 목도리 특징: 울로 짠 목도리 가격: 38000원	목도리(가) 구매 신청되었습니다.
확인 구매하기	확인

라. 검색기능

1 상품전체 보기 2▶상품검색 3 장바구니 보기	원하는 검색방법을 선택하세요
OK	1▶이름으로 찾기 2 가격으로 찾기
	OK

이름: 목장판	제조사: 세종 상품명: 목장판 특징: 천연목 100개 가격: 100000원
입력 alpha	확인 장바구니에추가

- 이름으로 찾기, 가격으로 찾기라는 두개 메뉴 중에 이름으로 찾기를 선택한다. 그럼 이름을 입력한다.
- 장바구니에 추가할 수 있다.

원하는 검색방법을 선택하세요	최소가격: 1000
1▶이름으로 찾기 2▶가격으로 찾기	
OK	OK alpha

최대가격: 30000	시작가: 1000 최고가: 30000
	1▶베개 2 보온컵 3 시계
입력 alpha	자세히보기 취소

시작가: 1000 최고가: 30000	제조사: 대왕 상품명: 보온컵 특징: 온도보존 가격: 10000원
1▶베개 2▶보온컵 3 시계	
자세히보기 취소	확인 장바구니에추가

- 가격으로 찾기를 선택한다.
- 최소가격과 최대가격을 입력한다.

- 그에 맞는 상품들이 나온다. 거기서 맘에 드는 상품들을 선택하면 그에 대한 상세한 설명이 나온다.
- 여기서도 장바구니에 추가할 수 있다.

4.2.2 게시판 실행 결과

가. 게시판 읽기

무선 게시판 환영합니다.	1) 게시판 읽기 2) 게시판 쓰기
시작 option	OK

나. 게시판 쓰기

1) 게시판 읽기 2) 게시판 쓰기	제목: EJB시작하기
OK	OK +ALPHA
이름: 홍길동	내용: 이제부터 EJB강의를 시작하겠습니다.
OK alpha	Menu alpha

- 2번 게시판 쓰기를 선택하고 순서에 따라 제목과 이름과 내용을 입력한다.

5. 결 론

큰 폭의 성장을 거듭하고 있는 무선 인터넷 시장은 다양한 콘텐츠를 요구하고 있으나, 유사한 시스템의 중복적인 개발로 인해 자원과 인력의 낭비를 초래하고 있다. 콘텐츠의 효과적인 개발을 위해선 재사용성이 뛰어난 컴포넌트 기법을 이용하여 효율성을 증가시킬 필요성이 있다.

본 논문에서는 모바일 전자상거래를 위한 무선 콘텐츠 제작 환경에 대한 기술적인 분석을 통해 EJB 기반의 모바일 전자상거래 시스템을 제작에 적합한 제작 환경을 고찰하고, 모바일 전자상거래에서 기본적으로 요구되는 구성요소를 컴포넌트로 구현함으로써 모바일 전자상거래를 위한 EJB 컴포넌트 기술에 대하여 설계 및 구현 방안을 제시하였다.

효과적인 모바일 전자상거래 정보의 사용을 위해서는 서비스에 대한 요구분석과 이에 수반되는 필요 컴포넌트의 표준화 기술의 연구와 개발이 이루어져야 할 것이다.

참 고 문 헌

[1] D. J. Goodman, "The Wireless Internet : Promises and Challenges," IEEE Computer, Vol.33, pp.36-41, July, 2000.

[2] G. Flammia, "The Wireless Internet Today and Tomorrow," IEEE Intelligent Systems, Vol.15, pp.82-83, Sept.-Oct., 2000.

[3] Richard Monson-Haefel, "Enterprise Java Beans," 2nd ED., O'Reilly, 2000.

[4] J. Davison, "Mobile E-Commerce," Ovum, 2000.

[5] 양영중, 김철홍, 박장순, "소프트웨어 컴포넌트 산업 현황 및 산업육성 정책 방향", 정보과학회지, Vol.19, Feb., 2001.

[6] "무선 인터넷 백서 2001", 소프트뱅크 미디어.

[7] A. Fox, S. Gribble, T. Chawathe and E. Brewer, "Adapting to Network and Client Variation using Active Proxies : Lesson and Perspectives," IEEE Communications, pp.10-19, Aug., 1998.

[8] R. Gloyd and B. Housel, "Mobile Web Access using eNetwork Web Express," IEEE Personal Communications, pp. 47-52, Oct., 1998.

[9] B. Housel and D. Lindquist, "Web Express : A System for Optimizing Web Browsing in a Wirelsss Environment," MOBICOM 96, pp.108-116, Nov., 1996.

[10] S. H. Phatek, et. al., "Web& : An Architecture for Non-Interactive Web," <http://www.cs.rutgers.edu/~phatak/www9/www9.html>.

[11] Charles Arehart, et. al., "Professional WAP," WROX Press, 2000.

[12] Subrahmanyam, Ronald, Chad, Robert, Tracie, Mark, Alex, Alex, Jim James, Thor, Bryan, Don, Dr P G, Dave, "Professional Java E-Commerce," Wrox. Press Ltd.

[13] Rahim Adatia et. al., "Professional EJB," Vol.2, WROX Press, 2001.

[14] 박지훈, 이용원, 김종윤, "EJB 엔터프라이즈 자바빈즈", 대청출판사, 2001.

[15] Sandra L. Emerson, Michael Girdley, Rob, "J2EE Applications and BEA WebLogic Server," Prentice Hall, 2001.

[16] Subrahmanyam, Ronald, Chad, Robert, Tracie, Mark, Alex, Alex, Jim James, Thor, Bryan, Don, Dr P G, Dave, "Professional Java E-Commerce," Wrox. Press Ltd. pp.815-895.

[17] James T. Perry, Gary P. Schneider, "Electronic Commerce 2E," COURSE TECHNOLOGY, 2001.

[18] Joanne Neidorf, Robin Neidorf, "e-Merchant : Retail Strategies for e-Commerce," Addison-Wesley, 2001.

[19] 조완수, "UML 객체지향 분석 설계", 홍릉 출판사, 2000.

[20] Booch, Rumbaugh, Jacobson, "The Unified Modeling Language User Guide," Addison Wesly, 1998.

[21] [WML] "Wireless Markup Language," WAP Forum, April, URL : <http://www.wapforum.org/>, 1998.

[22] Foo Soo Mee, Karli Watson, Ted Wugofski, "Beginning WAP-WML and WMLScript," WROX Press, 2000.

[23] 차정은, 양영중, 신석규, "EJB 컴포넌트의 코드 자동 생성 도구의 개발", 정보처리학회 2001년도 추계학술발표논문집, 제 8권 제2호, 2001.

[24] 김동관, 김철홍, 정효택, "서블릿에서 EJB컴포넌트 추출을 위한 기법", 정보과학회 2001가을 학술발표논문집, 제28권 제2호, 2001.



신 동 규

e-mail : shindk@sejong.ac.kr
1986년 서울대학교 계산통계학과(이학사)
1992년 Illinois Institute of Technology
전산학과(공학석사)
1997년 Texas A&M University 전산학과
(공학박사)

1986년~1991년 한국국방연구원 연구원
1997년~1998년 현대전자 멀티미디어연구소 책임연구원
1998년~현재 세종대학교 컴퓨터공학과 조교수
관심분야 : 웹기반 멀티미디어, WAP, XML응용, 영상압축



신 동 일

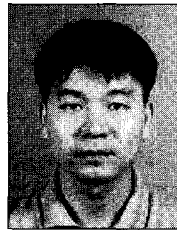
e-mail : dshin@sejong.ac.kr
1988년 연세대학교 전산학과(이학사)
1993년 M.S. in Computer Science,
Washington State University
1997년 Ph.D in Computer Science,
University of North Texas

1997년~1998년 시스템공학연구소 선임연구원
1998년~현재 세종대학교 컴퓨터공학과 조교수
관심분야 : 이동통신, WAP, XML응용, CSCW, 지능형 에이전트



차 석 일

e-mail : kiry@gce.sejong.ac.kr
2000년 세종대학교 전산·정보·응용통계
학부 졸업(이학사)
2001년~현재 세종대학교 공과대학 컴퓨터
공학과 재학
관심분야 : 전자상거래 보안, 어플리케이션
서버, 웹 기술 등



장 철 수

e-mail : jangcs@etri.re.kr
1995년 인하대학교 공과대학 전자계산공
학과 졸업(공학사)
1997년 광주과학기술원 정보통신공학과
1997년~현재 한국전자통신연구원 연구원
관심분야 : 모바일응용서버, 웹 기술, 데이터
베이스 등



이 경 호

e-mail : khleesun@etri.re.kr
1987년 고려대학교 산업공학과 학사
1990년 고려대학교 산업공학과 석사
1990년~현재 ETRI 책임연구원
관심분야 : e-Business, 모바일 어플리케
이션 서버



김 중 배

e-mail : jjkim@etri.re.kr
1986년 고려대학교 공과대학 산업공학과
졸업(공학사)
1988년 한국과학기술원 산업공학과 졸업
(공학석사)
1998년~현재 한국과학기술원 산업공학과
박사과정

1988년~1991년 대한항공(주) 시스템부
1991년~현재 한국전자통신연구원 컴퓨터소프트웨어기술연구소
모바일응용서버연구팀장
관심분야 : 미들웨어, 시스템 소프트웨어, 인터넷 정보검색 등