

모바일게임의 분석과 설계에 대한 연구

(A Study on the Design and Analysis of a Mobile Game)

우 원 택*

(Won-Taek Woo)

요 약 본 연구는 모바일게임 개발에 있어 가장 편리한 도구로 알려져 오고 있는 J2ME를 이용하여 하나의 모바일게임을 개발해 봄으로써 자바 모바일게임의 개발과정을 쉽게 이해하고 J2ME의 취약점을 보완할 수 있는 개발 방법을 제시해 보고자 함에 그 목적이 있다. 아울러 자바 모바일게임의 개발과정에서 사용하는 기존의 도스(DOS)모드에서의 Batch 파일 이용방법과 J2ME Wireless Toolkit 이용방법 대신에 새로운 통합 개발환경인 이클립스(Eclipse)를 이용한 모바일게임 개발방법을 소개함으로써 복잡한 명령어나 설치과정을 생각하지 않고 보다 쉽게 모바일게임을 개발할 수 있는 방법을 제시하였다. 이밖에 모바일게임의 환경, J2ME플랫폼의 구조, 기존연구자의 이론적 연구에 대한 문헌적 고찰을 병행함으로써 J2ME의 어떤 부분이 모바일게임의 특성에 중요하고 모바일게임 개발과정에서 이 중요한 부분을 어떻게 구현하는지를 알아보았다.

핵심주제어 : 자바모바일게임, 자바응용프로그램, J2ME

Abstract The purpose of this study is to design and develop a mobile game using J2ME, which is currently the most popular choice for game development, to understand the process of mobile game development and to develop methods that address weaknesses of J2ME. The development was done with the Eclipse IDE(Integrated Development Environment) tool to simplify mobile game development by removing the need for complex commands and installation processes which are intrinsic to the DOS command Batch file and J2ME Wireless Toolkit when used for the implementation of java mobile games. Literature was also reviewed on mobile game environments, the structures of J2ME platform, and past research on mobile games, to investigate which parts of J2ME are important to the characteristics of mobile game development and how to implement these important parts when developing.

Key Words : Java Mobile Game, Java Application Program, J2ME

1. 서 론

모바일콘텐츠시장에서 가장 주목을 받고 있는 콘텐츠는 모바일게임시장이다. IGDA의 2005년 모바일게임백서에[27] 따르면 전 세계 모바일게임시장의 매출은 2003년 11억 달러였고 2008년에는 ARC연구그룹에[1] 따르면 84억 달러로, Screendigest 연구그룹에[2] 따르면 41억 달러가 될 것으로 각각 다르게 추정하고 있다. 전 세계

시장점유율도 2003년 매출기준으로 일본이 52%, 한국이 26%, 유럽이 10%였으나 2008년에는 유럽이 31%, 미국이 26%, 일본이 15%, 한국이 9%로 시장점유율이 변경될 것으로 예측하고 있고 앞으로 미국과 유럽시장의 급성장이 예상되고 있다. 국내 모바일게임시장에 대한 한국게임산업진흥원의[3] 보고에 따르면 2002년도에 1,004억 원 수준이던 규모가 매년 50%의 성장률을 기록하여 2005년에는 2,187억 원 2006년에는 4,200억 원 수준으로 예측되고 있다. 이런 성장세를 지속한다면 국내 전체게임시장에서 모바일

* 대구한의대학교 모바일컨텐츠학부

일게임이 차지하는 비중도 현재 4% 수준에서 2006년 7% 수준까지 높아지게 된다. 현재 SK텔레콤, KTF, LG텔레콤 등 이동통신 3사가 서비스하고 있는 모바일게임은 1300개가 넘는 것으로 집계되고 있으며 현재 국내에서 모바일 게임을 개발하고 있는 업체는 400-700개 정도로 추정되고 있다. 최근에는 엔씨소프트, 넥슨 등 기존 대형 게임사들도 모바일게임시장에 직간접적으로 진출해 상당한 경쟁이 예측되고 있으며 국외의 경우에도 모바일게임시장의 성장 잠재력으로 게임 산업에는 참여하지 않을 것만 같았던 전통적인 기업들까지도 이에 동참하고 있다. 선 마이크로시스템즈(Sun Microsystems)의 게임사업부가 그 예이다. 그리고 노키아(Nokia)의 엔게이지(N-Gage)와 퀄컴(Qualcomm)의 브루우(Brew)와 같은 모바일장치생산자와 보다폰(Vodafone)의 콘텐츠 서비스와 같은 네트워크운영자도 산업에 진입하고 있다. 모바일게임은 C++, Java 정확히 말해서 J2ME(Java 2 Microedition)와 퀄컴의 브루우 플랫폼으로 개발할 수 있다. C++은 시스템자원에 바로 접근하여 네이티브코드(Native Code)로 컴파일 할 수 있는 장점이 있고 브루우는 C++, Java, XML, Flash를 포함하는 어떠한 언어와도 작업이 가능하고 모바일게임 개발자들에게 전 영역에 걸친 해결책(End-to-end solution)을 제공하지만 자바가 게임개발에서 가장 지지받는 모바일 플랫폼으로 지지 받고 있다.[4]-[8] 1999년 팜 OS용으로 KVM(Kilobyte Virtual Machine)이 소개된 이후 소형기기를 위한 자바 플랫폼이 정의되기 시작하였고 J2ME가 만들어 졌다. J2ME는 자바가 제공하는 특징인 일반성을 최대한 활용할 수 있게 노력하였고 대부분의 소형 기기에서는 통일된 GUI를 제공하지 않는데 J2ME에서는 가능한 일반적인 GUI를 제공하려고 노력하고 자바의 기본 특징인 플랫폼 독립성을 유지하려고 하였다. 무선 인터넷 서비스 제공에서 WAP 애플리케이션 방식이나 I-Mode 프로그램 방식과는 달리 MIDlet을 실행하는 자바 폰은 IP/HTTP 방식을 사용하기 때문에 별도의 무선 프로토콜 없이 바로 서버로 연결할 수 있고 대부분의 컴퓨팅이 휴대폰 자체에서 실행될 수 있

다는 장점이 있다. [9]-[11] 본 연구는 모바일게임 개발에 있어 가장 편리한 도구로 알려져 있는 J2ME를 이용하여 하나의 모바일 게임을 개발해봄으로써 자바모바일게임의 개발과정을 쉽게 이해하고 J2ME의 취약점을 보완할 수 있는 개발 방법을 제시해보고자 함에 그 목적이 있다. 이를 위해 본 연구에서는 첫째, 모바일게임의 정의, 분류, 가치사슬과 모바일게임에 대한 기존 연구를 고찰해보고 둘째, 모바일플랫폼의 구조와 MIDP(Mobile Information Device Profile)의 역할, 취약성 및 프로그래밍적 접근법등을 알아봄으로써 J2ME 프레임워크의 새로운 활용방법과 모바일게임의 특성에서의 J2ME의 중요부분을 살펴보고 마지막으로 통합개발 툴인 이클립스를 이용하여 하나의 모바일 게임을 개발하는 과정을 소개하여 봄으로써 종합적 차원에서의 모바일게임의 개발과 관련 이슈들에 대해 연구해보고자 한다.

2. 모바일 게임의 환경

좋은 게임을 만들기 위해서는 모바일게임의 특성을 잘 파악하고 모바일게임이 각광 받는 이유와 그 환경에 대해 잘 이해하고 있어야 성공적인 모바일게임 개발의 기회를 잡을 수 있다. 이하에서 모바일게임의 정의, 분류 및 그 가치사슬에 대한 모바일게임의 환경과 선행연구자들의 연구에 대해서 알아보려고 한다.

2.1. 모바일 게임의 정의

철학자 루트비그 비트겐슈타인(Ludwig Wittgenstein)이 1953년에 그의 저서 철학적 탐색(Philosophical Investigations)에서 게임에 대한 정의를 발표한 이후 불란서의 사회학자 로저 크와라와(Roger Coillois)와 게임설계자인 크리스 크로포드(Chris Crawford), 에릭 짐머만(Eric Zimmerman)등 여러 학자들이 게임에 대한 정의를 내리고 있으며[12]-[15] 위키피디아(Wikipedia), 시티젠다인(Citizendium), 컨서버피디아(Conservapedia)등의 유명 온라인백과사전

에서도 게임을 정의하고 있다. 이 중에서도 가장 최근인 2007년에 어니스트 애덤스(Ernest Adams)와 앤드류 롤링스(Andrew Rollings)가 내린 정의를 보면 '게임은 가공의 현실 속에서 행해지는 놀이의 일종으로 게임의 참여자는 여러 규칙에 따라 적어도 하나 이상의 임의적이고 어려운 목표를 달성하려고 하는 행위' 라고 규정하고 있다.[16] 여기에서 게임이 되는 필수조건으로는 놀이(Play), 목표(goal), 규칙(Rules), 가공(Pretending)의 네 가지 요소가 있으며 장난감은 규칙과 목표가 없으므로, 퍼즐은 목표를 가지고 있지 않으므로 게임이 될 수가 없다고 하였다. 여기에서 목표는 재미, 점수 따기, 결승점도착, 공주구하기, 악한이나 보스 죽이기, 세상 구하기 등이 될 수 있다. 한편 위키피디아에서[17] 정의한 모바일게임의 내용을 보면 '모바일게임은 모바일 폰이나, 스마트폰, PDA 또는 핸드헬드(Handheld)컴퓨터 등에서 실행되는 비디오 게임이다'라고 하였다. 모바일게임은 모바일장치 그 자체에서 제공하는 단순문자메세지(SMS), 멀티미디어메세지서비스(MMS)나 GPRS 위치식별 커뮤니케이션 기술을 이용하여 게임을 만들 수도 있으나 통상적으로는 모바일 폰에 다운로드하여 플레이하는 게임을 말하며 모바일 회사(Operator)의 무선네트워크(Radio network)를 통하여 다운로드하거나 모바일장치 구입 시에 탑재된 게임을 이용하거나 블루투스(Bluetooth)나 메모리카드의 적외선 연결을 통해 다운로드하여 이용한다.

2.2. 모바일 게임의 분류

모바일게임은 윈도우모바일, 팜OS, 심비안 OS(Symbian OS), 매크로미디어(Macromedia)의 플래시라이트(Flash Lite), 도코모(DoCoMo)의 도자(Doja), 선의 JavaME, 퀄컴의 BREW(Binary Runtime for Wireless), WIPI(Wireless Internet Platform for Interoperability)나 인퓨시오(Infusio)의 ExEn(Execution Environment)와 같은 기술과 플랫폼을 사용하여 개발한다. 이 중에서도 자바가 당초에는 가장 일반적인 플랫폼

이었다. 그러나 자바의 성능 제한과 더 복잡한 게임의 개발이 요구되어 여러 가지 네이티브인 형식의 플랫폼들이 채택되고 있다. 모바일게임의 장르별 분류를 보면 <표 1>과 같다.

<표 1> 모바일게임의 장르

장르별(모바일게임)			
플레이 형태별	게임의 내용별	이용방식별	프로토콜
-내장형 -1인용 (지역/네트워크) -멀티 플레이어 (지역/네트워크)	-보드 -대전 -슈팅 -스포츠 -어드벤처 -아케이드 -시뮬레이션 RPG	-단말기내장 -스트리밍 방식 -다운로드 방식	-WAP방식 -VM방식 KVM GVM

모바일게임은 플레이 형태에 따라 임베디드, 싱글, 멀티플레이 게임의 3가지로 분류할 수 있고 게임의 내용별로는 보드, 대전, 슈팅, 스포츠, 어드벤처, 아케이드, 시뮬레이션 RPG로 나눌 수 있으며 이용방식에 따라 단말기 내장, 스트리밍 방식, 다운로드 방식으로 분류된다. 이용방식에 따른 모바일게임의 분류를 보면 <표 2>와 같다.

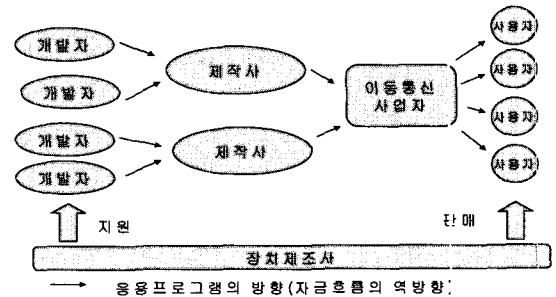
<표 2> 모바일게임의 분류

방식	특징
임베디드 게임	+휴대폰에 내장된 게임 +선택폭 좁고 대부분 No 그래픽
싱글게임	+로컬 또는 네트워크 연결 +사용자 혼자 하는 단순 게임
멀티 플레이어	+로컬 또는 네트워크 연결 +여러 사용자가 서로 대전 +일대일 대전 보드게임이 주류
스트리밍 방식	+WAP이용이 대표적/온라인접속 +대부분 텍스트위주, 일부 그래픽은 흑백 또는 4Gray, 약간의 칼라 이미지 사용, 단순애니메이션 가능
다운로드 방식	+오프라인 또는 온라인 플레이 가능 +게임라이브러리내장된 단말기에서 다이내믹한 표현게임

2.3. 모바일 가치사슬

투자적 관점에서의 모바일게임의 매력은 널리 사용될 수 있다는 점이다. 십대들을 대상으로 하는 콘솔게임과는 달리 모바일게임은 어느 때(Anytime), 어느 곳(Anywhere), 어느 누구(Anyone)나 접근할 수 있는 특성이 있으므로 게임개발자에게는 상대적으로 저렴하고 여러 소비자 계층에 따른 특화된 게임을 구현함으로써 더욱 폭넓고 수익성 있는 게임의 개발 가능성을 제공할 것으로 보인다.[18] 그러나 모바일화는 삶의 질을 높이고 개인의 유연성과, 접근성, 업무 효율 및 생산성, 안전성 등을 높일 수 있는 장점이 있는 반면 비용, 과도한 사용, 프라이버시, 보안, 환경과 보건 등의 문제를 야기 시키고 있다.[19] 모바일게임의 성공 요건으로는 첫째 배우기가 쉬워야 한다. 모바일게임은 전문가들에 비해 학습곡선이 완만하고 게임을 배우는데 많은 시간을 보내려하지 않는 고객의 특성을 가지기 때문에 게임은 다운로드 즉시 플레이 할 수 있도록 단순하여야 한다. 둘째 좋은 모바일게임은 짧은 시간 동안 오락가치를 제공하고 게임과 직무 작업 간을 부드럽게 이전할 수 있도록 일시정지성이 제공 되어야 한다. 셋째 지속된 수입을 얻기 위해서는 가입기반(Subscription-based)의 게임이 개발되어야 한다. 넷째 풍부한 상호작용성이 있는 게임을 만들어야 한다. 다른 플레이어를 끌어들이며 게임플레이어의 지능과 무작위성을 높이는 풍부한 사회적 상호작용이 있는 게임이 오늘날 대형 다수자게임에서 성공을 거두고 있기 때문이다. 다섯째로 모바일 혁신의 장점을 이용해야 한다. 거대한 연구비가 투자되어온 기술 예를 들면 글로벌위치시스템(GPS: Global Positioning System), 확장바코드스캐너(Extensions bar code scanner), 단순문자메세지서비스, 멀티미디어문자서비스 등의 혁신적 장치의 특성과 네트워크 인프라스트럭처들을 이용할 수 있어야 한다. 마지막으로 모든 연령과 성별계층이 모바일게임을 하기 때문에 폭력적이거나 노골적 성적 콘텐츠는 피해야 한다. 모바일게임의 시장성과 아울러 모바일시장에 관여하는 가치사슬을 보면 모바일게임의 가치사

슬은 기존 게임이나 기업 응용프로그램과는 상당히 다른 가치사슬을 가지고 있다.[20] <그림 1>은 모바일가치사슬의 구성원과 자금의 흐름을 보여주고 있다.



<그림 1> 모바일가치사슬

<그림1>에서 소프트웨어개발자들은 응용프로그램을 만듦으로써 가치사슬을 유도한다. 그러나 대부분의 개발자들은 대중에 대한 매쓰마케팅을 할 능력이 없고 고객이 다가갈 수 있는 유통채널도 없다. 게임 제작사(Publishers)는 개발자로부터 타이틀들을 수합하여 브랜드 이름을 붙이고 기술적 정확성을 증명하고 게임을 판촉하며 이동통신사업자(Wireless carriers)들과 사업관계를 구축한다. 제작사는 개발자들을 대신하여 대금을 청구하고 개발자와 더불어 수익을 공유한다. 이동통신사업자는 모바일게임의 유통채널의 중심에 있다. 이동통신사업자는 네트워크를 관리하고 고객정보를 가지고 있으며 어떤 장치와 서비스를 제공할 것인지를 결정한다. 이동통신사업자는 모바일 전체수익의 큰 부분을 차지한다. 그리고 기존의 서비스협정에 따라 고객에게 대금을 청구한다. 장치제조사(Wireless Device Manufactures)는 기술혁신을 유도하며 새로운 하드웨어와 플랫폼을 지원한다. 사실 J2ME의 성공은 주로 주요장치제조사의 채택 때문이다. 장치제조사는 이동통신 사업자와 고객 간에 독립적인 힘을 나타낸다. 마지막으로 고객은 가치사슬의 구성원들에게 비용과 이윤을 지불한다. 고객은 제작사들의 웹사이트로부터 또는 이동통신 사업자들이 제공하는 콘텐츠제공시스템을 통하여 게임을 구입한다.

2.4. 모바일 게임의 선행연구

유비쿼터스 시대를 지향하는 오늘날 정보기술을 고려할 때 모바일의 생활화는 시대적 대세로 잡아가고 있고 향후에도 지속적으로 모바일게임의 수요는 증대할 것으로 예상되지만 모바일게임에 관한 연구와 개발사례에 관한 연구는 그다지 많지가 않은 것 같다. 국내의 연구자에 의한 연구와 연구주제를 보면 <표 3>과 같다.[21]-[25]

<표 3> 국내연구자의선행적연구

연구자/게제지	연구주제	기여도/시사점
최재영,황병곤 (2006대구대정보통신연구)	위퍼플랫폼모바일 게임개발	마우스치즈찾기 게임개발/지뢰찾기 게임의 확장성 연구
홍석형,이영철 (2005세명대세명논총)	PDA온라인 고스톱 게임	실시간무선고스톱 게임개발/모바일네트워크게임의 확충
성재경,김용국 (2003한국정보과학회가을발표논문집)	PDA네트워크 바둑게임	네트워크바둑게임의 UI설계/휴대용바둑 게임의 변형개발
전준곤, 김현수 (2003한국정보과학회가을발표논문집)	자바API 구현테스팅	J2ME, KVM, MIDP의 연동구현/통합연동 게임개발의 확대
최학현 (2002게임학회지)	무선인터넷게임 기술동향	무선게임의 기술성, 시장성 분석/유무선연동게임개발의 확대

국외연구의 경우는 <표 4>와 같으며[26]-[30] 연구논문보다는 교과서로 모바일지식을 전달하거나 모바일업체의 연구포럼이나 게임협회의 연구백서 등에 의한 발표나 인터넷 사이트에 연구발표가 더 많은 것으로 보인다.

3. 모바일플랫폼과 구조

J2ME플랫폼은 개발자, 장치제조사, 이동통신사업자, 고객 모두에 의해 가장 널리 채택되어 왔고 가장 탁월한 모바일게임 플랫폼으로 부상해 오고 있다. 이는 모든 모바일 당사자들에게

<표 4> 국외연구자의 선행적 연구

연구자/게제지	연구주제	기여도/시사점
Nokia (2007Forum.Nokia.Com)	모바일게임그래픽	모바일그래픽기능 향상/스크린방향조절기능,고정랜드스케이프화면등 소형스크린의 한계극복
IGDA online Game SIG(2005IGDA)	모바일게임백서	최신의 모바일기술동향및 시장분석자료/차세대기술동향 및 시장성과악
Liljedal(2005Upsala Uni. 석사학위논문)	컨택트지카 모바일게임	유령잡기게임개발/모바일특성과 게임플레이와의 혼합 증대
Bjork(2001HCI동경학술대회)	모바일게임용2인용해적게임	멀티플레이해적모바일게임구축/사회적상호작용성을 고려한 게임개발의 확대
Thomas(2000Wearable Computer 심포지움)	위치기반모바일 게임	ARquake모바일 FPS 게임개발/게임개발시 GPS, 시각추적시스템 등의 모바일특성 융합 확대

이익이 되는 중요한 특성들을 가지고 있기 때문이다. 이하에서 모바일게임의 개발언어, J2ME 플랫폼의 구조, MIDP의 역할과 J2ME의 취약성과 프로그래밍적 접근법에 대해 살펴보면 다음과 같다.

3.1. 모바일게임의 개발언어

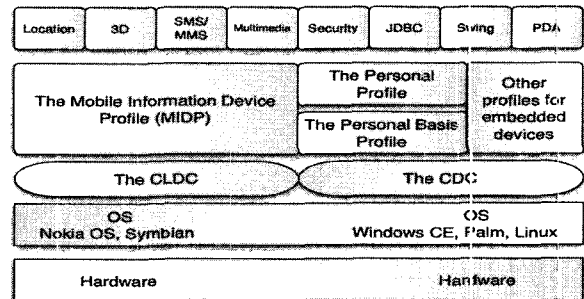
앞 절의 모바일게임의 분류에서 살펴 본바와 같이 모바일게임은 C++, J2ME와 쉘컴의 브루우 플랫폼 등으로 개발할 수 있다. 브루우는 C++, Java, XML, Flash를 포함하는 어떠한 언어와도 작업이 가능하지만 자바가 게임개발에서 가장 인기가 있는 선택이 되고 있다. J2ME는 모든 핸드셋 시장에서 지원받는 산업표준으로 오늘날 모바일 폰의 대부분은 자바가 가능하다. J2ME는 무료이고 오픈플랫폼으로서 개발비용이 저렴하고 이를 사용하는 개발자들에게 무료로 사용가능하다. 그러나 상업용 애플리케이션을

개발할 경우에는 선사에 로열티를 지불하여야 한다.[24] 이러한 이유와 다양한 언어 사용의 문제를 피하기 위해 한국통신기술협회에서는 WIPI를 제안하였다. WIPI의 경우 C와 Java 두 가지 언어를 지원하고 있다. 그러나 WIPI C의 경우 C 언어의 문법을 사용하지만 순수하게 새로 정의된 WIPI만의 API를 정의하고 있다. 반면 WIPI Java의 경우는 Java의 문법뿐만 아니라 기본 API 역시 호환이 된다.[31] J2ME장치는 오프라인모드에서 게임을 플레이할 수 있고 모바일 장치의 여러 프로토콜을 사용하여 후방게임서버(Backend game server), 다른 무선클라이언트(Wireless peer)와 다른 인터넷클라이언트(Internet peer)들을 하나로 통합시킬 수 있으며 시각적 호소력이 강한 게임에 필수적인 풍부한 사용자 인터페이스를 제공한다. 그리고 J2ME의 스마트클라이언트는 모바일네트워크의 SMS나 MMS서비스에도 접근할 수 있고 GPS와 같은 확장장치에도 접속할 수가 있으며 이러한 기능은 모바일 라이프스타일의 이점을 충분히 활용할 수 있는 게임의 개발을 가능하게 한다.[32] 그리고 고급 암호화 기술과 디지털 서명알고리즘을 제공함으로써 프라이버시에 민감한 자료들을 인터넷을 통해 안심하고 보낼 수 있게도 한다.[33] 코드 작업의 높은 휴대성("Write once run anywhere")은 한 브랜드 내지 한 종류의 핸드셋을 위해 작성된 게임 응용프로그램이라도 자바가 가능한 핸드셋의 어떤 다른 브랜드나 핸드셋 유형에서도 동작이 가능하다. 최근의 MIDP 버전 2.0의 경우에는 그 자신의 모든 API가 게임개발에 전용토록 되어있어 게임개발을 더욱 간단하게 그리고 빠르게 하고 있다.

3.2. J2ME플랫폼의 구조

1999년 Sun사에 의해 발표된 J2ME의 핵심적인 요소는 CLDC(Connected Limited Devices Configuration)와 MIDP로 구성되어 있다. CLDC는 셀폰(Cell Phone)이나 PDA와 같은 작고 제한된 모바일장치들에 맞는 하드웨어적 구성을 잡아주는 Java언어의 핵심라이브러리로 구성되어 있고 MIDP프로파일은 CLDC상에서 구현

되는 GUI, 네트워킹, 저장장치 API 등을 가지고 있는 특정 장치 중심의 고급화된 API라이브러리이다. 특정 프로파일을 위해 작성된 자바프로그램은 이 프로파일에 의해 지원되는 하드웨어나 OS플랫폼 간에 포팅(Porting)이 가능하며 MIDP와 PDA프로파일은 CLDC의 가장 중요한 프로파일들이다. 그리고 J2ME는 <그림 2>에서 보는바와 같이 컨피규레이션(Configuration), 프로파일(Profile)과 옵션패키지로 구성된 다층구조를 채택함으로써 이전의 데스크톱 자바언어에서 나타난 최저공용지배자(Lowest common dominator)문제를 해결하였다. [34]-[37]



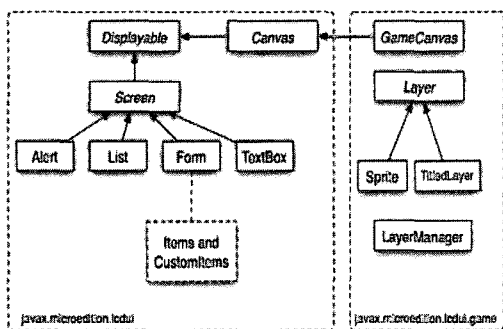
<그림 2> J2ME의 구조

상기 J2ME의 구조에서 보는바와 같이 MIDP 위에는 GPS, 3D, SMS/MMS, 멀티미디어, 보안, JDBC, 스윙(Swing), PDA 등 다양한 패키지들을 탑재할 수 있다. CLDC는 자바의 Core API와 KVM을 정의하고 있으며 Core API는 java.lang, java.util, java.io가 해당된다. MIDP는 CLDC를 기본 컨피규레이션으로 정하고 있다. MIDP에서는 J2SE(Java 2 Standard Edition)에서 사용할 수 있는 API 중에서 일부만을 사용할 수 있지만 그 외에 추가적으로 정의하고 있는 API들도 있다. MIDP에 해당하는 API는 javax.microedition.midlet으로 MIDlet의 기본적인 프레임워크를 잡아준다. javax.microedition.lcdui는 MIDlet 화면인 LCD 창에 디스플레이할 수 있도록 하여주고 javax.microedition.rms는 정보를 MIDlet이 끝나도 남아 있도록 저장한다. javax.microedition.io는 네트워크 기능과 휴대폰의 적외선 및 직렬포트를 지원한다. J2ME 라이브러리와 구성요소들은 J2SE와 유사한 특징을

제공하기 때문에 J2SE 개발자들은 쉽게 모바일 세상으로 이전할 수 있게 되어 있다.[38][39]

3.3. MIDP 2.0의 역할

MIDP 2.0 API는 모바일장치에서 풍부한 내용의 멀티미디어 응용프로그램과 게임을 개발하기 위한 API들의 집합이다. MIDP 2.0은 MIDP 1.0에서 보다 많은 기능과 특징을 가지고 개발되었는데 MIDP 2.0에 추가된 새로운 중요 특징은 게임 API를 추가하였다는 점이다. 정확하게 말하면 javax.microedition.lcdui.game이라는 게임API패키지이다. MIDP 2.0은 게임개발자에게 MIDP 1.0에서 일반연습장의 스크래치 작업으로부터 개발해야만 했던 많은 코딩블럭을 사용하기 쉬운 구축블럭을 만들어 제공한다. 이러한 구축블럭들은 게임캔버스(Game canvas)를 이용한 고속이동이 가능하고 도형의 패턴을 화면에 표시하는 쪽화면(Sprites), 레이어(Layers)들과 같은 다양한 게임엘리먼트들을 생성하고 조정하기 위한 클래스들로 되어있다.



<그림 3> MIDP UI의 구조

<그림3>은 MIDP의 사용자 인터페이스의 구조를 보여주는 것으로 그림의 좌측은 javax.microedition.lcdui의 API들이고 우측은 javax.microedition.lcdui.game의 API들이다. javax.microedition.lcdui API패키지는 모든 유형의 모바일응용프로그램에서 사용자 인터페이스(UI: User Interface)를 개발하는데 필요하다. 이 API는 스크린, 폼, 텍스트박스, 라디오버튼과 같은 UI컴포넌트들을 만들고 조정하기 위한 클래스들

을 제공하며 게임을 포함한 모바일 응용프로그램의 입력처리를 위한 클래스를 제공한다. AWT와 스윙을 사용하여 GUI를 개발한 경험이 있는 개발자들은 javax.microedition.lcdui 패키지가 이들 API들과 유사하다는 것을 발견하게 될 것이다. MIDP 2.0 API의 다른 패키지로는 javax.microedition.midlet이 있다. 이 API 패키지는 모든 모바일 응용프로그램을 개발하는데 필요한 기본 사항들을 제공하며 javax.microedition.midlet.MIDlet 클래스들을 가지고 있다. 이는 MIDlet으로 알려진 J2ME 기반의 모바일 응용프로그램 개발의 기본이 되는 클래스이며 모든 모바일 응용 프로그램의 메인클래스에 의해 상속하여야 한다. MIDlet 클래스는 MIDlet을 생성하는데 필요한 자원들을 제공한다. MIDP 2.0을 사용하면 게임개발의 많은 시간을 단축시킬 수 있다.

3.4. J2 ME의 취약성과 프로그래밍

J2ME 프로그래밍은 간단하고, 구축 중립적, 객체 지향적, 휴대성, 분산체제, 높은 성능, 멀티쓰레드, 견고성, 다이내믹, 안정성 등의 자바언어자체의 이점을 모두 활용할 수 있을 뿐만 아니라 전문개발자의 경우 디자인패턴도 이용할 수 있다는 장점도 있다. 하나의 간단한 Hello프로그램도 쉽게 여러 개의 클래스로 나눌 수가 있으며 이렇게 함으로써 유지관리와 확장성을 용이하게 할 수 있다. 그러나 J2ME는 사용메모리의 제한성 때문에 여러 개의 객체를 사용하는 것이 좋다. 모든 작업을 다 수행하는 하나의 큰 슈퍼클래스를 만드는 것보다 여러 개의 간단한 클래스로 나누는 것이 좋다. 일반적으로 개발자가 고려해야할 메모리로는 작업메모리, 스토리지메모리, 응용메모리의 세 가지 유형이 있으며 게임 출시를 위해서는 특정 모바일장치의 사양을 장치제조업체와 상담하여 알아 두는 것이 좋다. 실행시의 최대 게임크기와 가능한 메모리도 체크하는 것이 좋다. 이밖에 스크린의 제한도 장치제조업체마다 다르고 모델마다 다르기 때문에 스크린크기(Screen size), 프레임레이트(Frame rate), 컬러 등에 대한 사양을 장치제조

업체와 상담하여 알아 두는 것이 좋다. 어떤 폰과 모델을 사용할 것인가를 결정한 후에는 게임을 최적화하기 위한 코딩 작업을 하게 된다.[40] 코딩을 위한 프로그래밍 방법으로는 첫째, 최소한의 클래스를 사용하고 둘째, 짧은 변수이름, 메서드이름, 클래스이름을 사용하고 셋째, SOAP과 같은 불필요한 프로토콜의 사용을 피하고 일반적인 프로토콜을 사용하여 단순화하는 것이 좋다. 왜냐하면 이를 처리하는데 필요한 추가적인 레이어 나 추가메모리를 차지하는 라이브러리 때문에 비용이 증가하기 때문이다. 넷째, 만약 네트워크게임이면 시작 시 한 번에 모든 자원을 내리받는 것이 좋고 그래픽, 맵, 레벨과 같은 자원을 얻기 위한 서버호출을 줄이는 것이 좋다. 다섯째, 그래픽을 최적화하고 여러개의 별개 그래픽으로 분산하는 것보다 하나의 시트에 통합하는 것이 좋다. 여섯째, 게임을 완전히 개발한 후에는 Obfuscator 도구를 실행시켜 개발된 코드의 역공학(Reengineering)의 위험을 제거하고 jar 파일의 크기도 현격히 줄이도록 한다.

4. 모바일게임의 분석과 설계

본 장에서는 앞서 설명한 MIDP의 API들과 이들 각각의 클래스들이 게임개발의 과정에서 어떻게 사용되고 연결되는지를 살펴보기 위해 하나의 장애물 모바일게임을 분석, 설계하여 실행해 봄으로써 이들의 이용방법을 알아보고자 한다. 본 연구의 게임은 자바모바일게임으로 단독 플레이어 오프라인 게임이다. 이 게임은 한대의 자동차가 장애물을 피하여 이동하는 것으로 플레이어는 장애물과의 충돌을 피하기 위해 자동차의 운전대를 왼쪽과 오른쪽의 핸드 셋 키를 사용하여 왼쪽으로 또는 오른쪽으로 이동한다. 충돌이 일어나면 게임은 종료하고 점수가 디스플레이 되는 단순 게임이다.

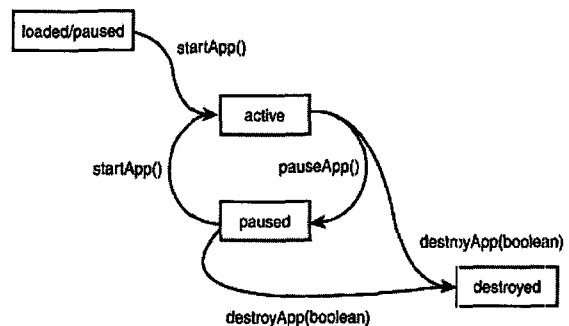
4.1. 장애물게임의 설계

장애물 게임을 설계하기 위해 앞 절에서 설명

한 J2ME의 취약성과 프로그래밍의 효율성 제고를 위해 제시된 여러 가지 접근법을 염두에 두어 우선 프로그램의 클래스를 FirstDriveMIDlet,, SecondDriveCanvas, ThirdObstacle, FourthStopCanvas의 네 개의 자바 파일로 구성하여 크기가 작은 프로그램이 되도록 하여 인스톨시 시간을 절약하게 하였다. 변수의 이름과, 클래스이름, 메서드이름도 짧게 정하고 기본적으로 사용되는 기능 이외에 부가적으로 불필요한 기능은 넣지 않도록 하여 프로그램의 용량을 줄이도록 하여 구성된 파일의 순서대로 구동되게 설계되었다. 그리고 2장의 모바일가치사슬에서 설명한 모바일 게임 사용자의 특성과 성공요인을 고려하여 배우기 쉽고, 다운로드 즉시 플레이 할 수 있게 단순하고, 짧은 시간동안의 오락가치가 제공되게 하였고, 직무와 게임간의 이전이 가능하게 일시적인 정지가 가능하게 하나의 단순 게임으로 설계하였다.

4.1.1. MIDlet의 구동

본 장애물게임이 실행되기 위해서는 먼저 MIDlet이 구동되어야 한다. MIDlet은 J2SE의 애플릿과 비교될 수 있는 것으로 MIDlet의 상태들은 디스플레이 상태들 보다는 좀 더 독립적이라는 점을 제외하면 거의 유사하다. MIDlet은 적재, 실행, 정지, 종료의 네 가지 상태를 가진다.



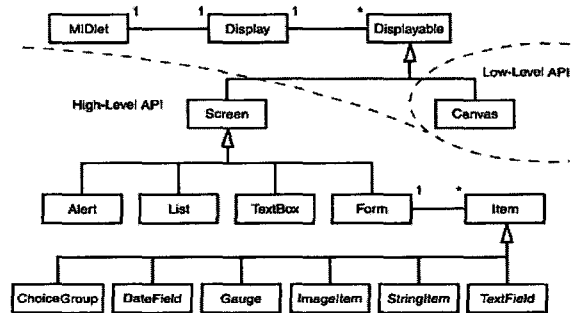
<그림 4> MIDlet의 생명주기

<그림 4>는 MIDlet의 생명주기를 보여 주는 것으로 MIDlet이 장치에 적재되면 생성자(Constructor)가 호출된다. 이 적재상태는 프로그램관리자가 startApp()메소드를 호출하여 응

용프로그램을 시작하기 전의 상태이며 startApp()메소드가 호출되면 MIDlet은 실행상태가 되며 프로그램관리자가 pauseApp()나 destroyApp()를 호출하기 전에는 실행상태에 계속 있게 된다. pauseApp()는 MIDlet을 일시 중지시키며 destroyApp()는 MIDlet을 종료시킨다.[41] FirstDriveMIDlet 자바파일은 MIDlet의 기본클래스들을 javax.microedition.midlet.MIDlet에서 상속받고 javax.microedition.lcdui.CommandListener에 인터페이스 하여 응용프로그램이 실행되고 처리될 동안 생성된 명령이벤트들을 받아들인다. 명령이벤트는 소프트웨어튼 등을 사용하여 Exit, Cancel, Back, Ok, Stop과 같은 명령을 발생시키면 생기는 것으로 commandAction()메소드에 의해 처리된다. FirstDriveMIDlet은 모든 캔버스를 포함하는 컨테이너 역할을 한다. 캔버스는 모바일장치의 스크린에 그림을 그리는데 필요한 표면을 나타내는 객체들이다. MIDlet은 javax.microedition.lcdui.game.GameCanvas 클래스에서 생성되는 SecondDriveCanvas를 가지고 있다. 게임캔버스는 효율적인 애니메이션 그래픽을 그리는 특수 캔버스로 애니메이션을 매끄럽게 하기 위해 버퍼로부터 스크린 그래픽을 대기시켜 순서를 기다리게 한다. FirstDriveMIDlet의 또 다른 캔버스로는 FourthStopCanvas가 있으며 이를 생성하기 위해 javax.microedition.lcdui.Canvas 클래스에서 상속받는다. 이 캔버스는 텍스트, 선, 단순 웨이프(shape)등을 그리는데 사용되며 스크린에 복잡한 그래픽을 그리는 대신에 단순한 그래픽을 그릴 때 사용한다. 예를 들면 게임종료 스크린이나 게임명령어와 같은 것을 디스플레이 할 때 사용된다. 게임 응용프로그램의 미들릿은 무수한 캔버스를 포함할 수 있으나 javax.microedition.lcdui.Display 클래스의 setCurrent() 메소드를 사용하여 한 번에 단지 한 캔버스만 디스플레이 시킬 수 있다. FirstDriveMIDlet의 startApp()메소드에서 SecondDriveCanvas가 초기화되고 SecondDriveCanvas의 addCommand() 메소드를 사용하여 Exit명령이 SecondDriveCanvas에 추가된다.

4.1.2. 캔버스의 실행

고급 API에 비해 저급 API는 MIDP 디스플레이가 픽셀 단위로 나타나게 하는 API들이다. 이를 위해 lcdui패키지는 캔버스라는 특수 유형의 스크린을 제공한다. 캔버스는 자체적으로는 어떤 드로잉메소드를 제공하지는 않고 paint() 콜백메소드를 제공한다. AWT 컴포넌트에 있는 paint()메소드와 유사한 프로그램매너저가 스크린에 그림을 그릴 필요가 있다고 결정하면 캔버스의 paint() 콜백메소드가 호출된다. paint() 메소드의 유일한 파라미터로 Graphics객체가 있다. Graphics객체는 drawline(), fillRect(), drawstring()과 같은 실제로 스크린에 내용을 그리는 데 필요한 모든 객체를 제공한다.[42] AWT와는 대조적으로 lcdui는 고급과 저급의 그래픽을 혼용하여 사용하지는 않는다. 고급 컴포넌트와 저급 컴포넌트를 동시에 스크린에 나타나게 할 수는 없다. 폼, 리스트, 버튼 등의 고급 컴포넌트를 디스플레이하기 위해서는 display클래스를 사용하거나 displayable클래스의 isshown() 메소드를 호출하여 스크린에 내용을 나타낸다. <그림 5>는 MIDP의 GUI를 저급과 고급으로 분류한 그림이다.



<그림 5> 저급과 고급의 MIDPGUI 클래스들

본 연구의 SecondDriveCanvas는 게임루프를 독립적으로 실행시키는데 필요한 쓰레드에서 그 자신을 실행시키기 위해 java.lang.Runnable 인터페이스를 실행시킨다. 게임루프는 게임을 종료하는데 필요한 조건이 될 때까지 게임을 계속적으로 실행시킨다. 본 연구에서 게임의 종료 조건은 자동차가 장애물과 충돌하거나 플레이어가 Exit버튼을 사용하여 게임을 종료하는 경우

이다. SecondDriveCanvas.java의 게임루프는 tick(), input(), render()와 같은 메소드들로 구성되어 있다. tick()메소드는 게임을 종료하는데 필요한 조건이 성립하는지를 체크하여 게임의 상태를 변경한다. input()메소드는 각각의 키 눌림에 대해 필요한 행위를 수행하며 render() 메소드는 게임의 상태에 따라 게임을 렌더링한다. SecondDriveCanvas는 캔버스의 여러 레이어 들을 추가하고 관리하는 javax.microedition.lcdui.game.LayerManager의 인스턴스를 사용한다. 이때 각 레이어는 스프라이트라는 한 쪽의 화면에 해당한다. 스프라이트는 게임의 한 시각적 기본요소로 게임캔버스를 돌아다니는 캐릭터나 다른 스프라이트와 상호 작용하는 것들을 말하는 것으로 여기서는 자동차나 장애물들이 스프라이트이다. 각각의 스프라이트는 완전히 또는 부분적으로 투명한 하나의 시각적 레이어를 형성하고 각각의 레이어들은 서로 겹개어져 있다. 스프라이트는 javax.microedition.lcdui.game.Sprite 클래스의 한 객체이며 스프라이트를 다른 스프라이트와의 충돌감지에 따라 스프라이트를 표시하거나 변형하거나 회전시키는 기능을 제공한다. tick()메소드는 내부적으로 충돌을 체크하기 위해 스프라이트클래스의 collidesWith() 메소드를 사용한다. SecondDriveCanvas는 게임캔버스에서 장애물 스프라이트를 이동시키고 렌더링 하는 Third Obstacle 클래스의 객체를 실행시키고 다른 자동차 스프라이트와 충돌하는지 않는지를 체크한다.

4.1.3. 장애물관리자

세 번째로 실행되는 장애물(ThirdObstacle)은 자동차의 경로 상에 무작위로 나타나는 장애물들을 만들고 관리하는 것으로 간단하고 편리한 사용을 위해 ThirdObstacle.java는 자동차의 경로에서 한 번에 나타낼 수 있는 장애물의 최대 수를 상수 값으로 가지고 있다. 장애물을 스크린의 각 위치에 무작위로 나타내기 위해 장애물 관리자는 아주 단순한 전략을 사용한다. 초기배치에서 10개의 장애물 스프라이트를 생성하여 이 스프라이트들을 게임캔버스에 나타내기 위해 레이어관리자(LayerManager)를 사용한다. 그리고 무작위로 생성된 X, Y 좌표 값으로 이들의

위치를 정하고 게임캔버스의 아래쪽으로 이동시킨다. 각각의 장애물들이 자동차와 충돌하지 않고 캔버스의 맨 아래 끝에 도착하면 이들의 위치는 다시 리셋(Reset)된다. 이때에 게임스코어의 값은 1이 증가된다.

4.1.4. 게임종료캔버스

FourthStopCanvas는 FirstDriveMIDlet으로부터 게임이 지속된 시간과 게임의 점수를 보여주는 캔버스로 실행루프에서 게임 종료조건이 나타나면 실행된다. 정상적인 게임을 하였다면 마지막 네 번째에 실행되는 자바 프로그램이다.

4.2. J2ME의 실행환경

J2ME프로그램을 작성하여 배포하는 과정을 보면 첫째 MIDlet 클래스를 작성하고 둘째 작성한 클래스파일을 컴파일(javac)하고 셋째 컴파일 한 클래스파일이 MIDP기준에 맞는지를 사전검증(Preverify)한다. 넷째, MIDP를 실행하여 MIDlet Suite 를 만든다. MIDlet Suite는 클래스와 자원(resource)을 포함하는 JAR파일과 JAR파일의 구성을 설명하는 JAD파일을 포함한다. 다섯째, 서버에 MIDlet Suite를 올린다. 여섯째, 모바일장치에서 JAM(Java Application Manager)을 이용하여 해당 MIDlet Suite를 다운로드하여 실행한다. 이러한 과정을 거치기 위해서는 먼저 J2SE SDK를 설치하고 J2ME의 MIDP를 설치하여야 한다. 실제로 도스(DOS)명령프롬프트환경 하에서 이들을 설치하고 Java 컴파일러와 MIDP의 실행파일의 구등을 위한 path 설정과 classpath 설정 및 실행과정은 매우 복잡하다. 따라서 두 번째의 실행 옵션으로 선마이크로시스템즈에서 제공하는 J2ME Wireless Development Toolkit(WTK)을 이용하는 방법이 있다. J2ME WTK은 MIDP-fcs를 내부에 포함하고 있기 때문에 별도의 MIDP-fcs를 설치하지 않아도 MIDlet을 실행시킬 수 있다. 그러나 WTK는 소스를 만들거나 수정하는 편집기능은 제공하지 않으며 소스파일을 토대로 컴파일, 사전검증, 그리고 미들릿슈트의 생성과 에뮬레이터 기능을 제공한다. 세 번째의 실행 옵션으로

이클립스, JBuilder, Forte for Java CE, 등의 IDE통합프로그램을 이용하는 방법이 있다. IDE는 편집기능과 더불어 실행환경을 모두 제공하므로 WTK나 도스환경 하에서 배치파일을 이용하여 실행하는 것처럼 소스파일을 만들고 나서 이를 다시 옮겨 JAR파일을 만드는 이중 작업을 하지 않아도 되기 때문에 상당히 편하다. 단점이라면 일단 통합개발환경이기 때문에 고사양의 시스템 상에서 사용하여야 한다는 점이다. 본 연구에서는 이클립스를 이용하여 개발하고자 한다. 왜냐하면 JBuilder는 상용프로그램이므로 구입하여야 하며 Forte for Java CE는 JDK와 MIDP의 설치순서에 민감한 통합개발환경인 반면 이클립스는 무료이며 이클립스미(EclipseME) 플러그인이 설치된 후, 이클립스는 모바일 프로그램에 가장 적합한 IDE중 하나가 되었으며 WTK와 함께 최적의 모바일 개발 환경을 갖추게 해주기 때문이다.

4.3. 장애물게임의 실행

장애물게임을 실행하기 위해서는 J2SE SDK와 WTK를 윈도우XP플랫폼에 설치하고 앞 절에서 설명한 통합개발환경인 이클립스를 이용하여 실행하였다. 이하에서 각각의 절차를 살펴보면 다음과 같다.

4.3.1. J2SE의 설치

<http://java.sun.com/javase/downloads/index.jsp>의 자바선의 J2SE다운로드 사이트에서 JDK 5.0 Update 9 with NetBeans 5.5의 다운로드 항목에서 윈도우용 JDK 5.0의 실행파일인 `jdk-1_5_0_09-nb-5_5-win.exe`를 다운로드하여 J2SE를 설치한다.

4.3.2. J2ME의 설치와 연동

<http://java.sun.com/javame/downloads/index.jsp>의 자바 선의 J2ME사이트에서 Sun Java WTK 2.2항목을 선택하여 회원가입 후에 `j2me_wireless_toolkit-2_2-windows.exe` 실행파일을 다운로드하여 J2ME를 설치한다. J2ME WTK를 설치하면 자동으로 JDK 5.0의 설치 디렉터리를 탐색

하여 경로를 연결하므로 J2SE와 J2ME는 서로 연동된다.

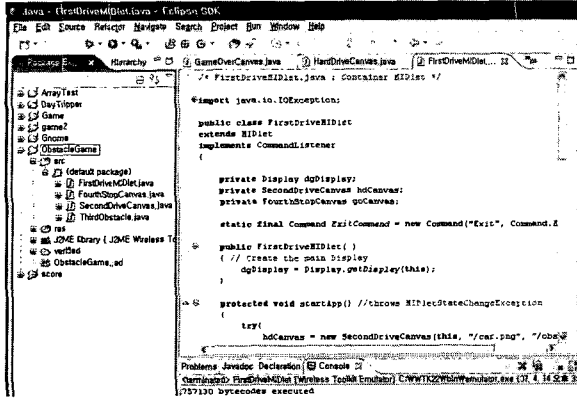
4.3.3 이클립스의 설치

자바프로그래밍을 위해서는 자바프로그램을 작성하고, 컴파일하고, 실행해야 된다. 컴파일러와 자바 VM은 J2SE라는 개발환경을 이용하게 되는데 프로그래머가 편리하게 사용하기 위해서는 프로그램을 작성하기 위한 에디터와 연결된 개발환경이 필요하다. 이를 지원하기 위한 도구가 이클립스이다. 이를 설치하기 위해서는 이클립스사이트 <http://www.eclipse.org/downloads/>에서 `eclipse-SDK-3.2-win32.zip`을 다운로드하여 설치하면 J2SE와 자동 연동되어 쉽게 프로그램을 개발할 수 있다. 이클립스로 자바모바일게임을 개발하기 위해서는 이클립스미를 플러그인으로 설치하여야 한다. 이클립스를 구동한 후 Help메뉴에서 Software updates를 선택하고 Find and install을 선택하여 <http://www.eclipseme.org/updates/>를 원격리모트Update사이트로 지정하여 이클립스미를 설치한다. 설치가 성공적으로 이루어지면 Window메뉴의 Preference 다이아로그에 J2ME항목이 추가된다. 다음으로 모바일 폰의 에플레이트를 이클립스에 설치하여야 하는데 이전에 설치했던 WTK에 있는 모바일 폰의 에플레이트를 Import해오기 위해 이클립스의 Window메뉴의 Preferences를 선택한 후 J2ME항목의 Devise Manager를 선택하여 Import 창을 띄운 후 J2ME WTK가 설치된 디렉터리(예를들면 `C:\WTK22`)를 입력한 후 Refresh버튼을 클릭하면 J2ME WTK에 있는 모바일 폰의 에플레이트가 이클립스로 Import된다. 이렇게 하면 J2SE, 이클립스, 이클립스미, WTK가 종합적으로 연동되어 이클립스 환경 하에서 쉽게 모바일프로그램을 개발할 수 있다.

4.3.4. 모바일게임의 실행

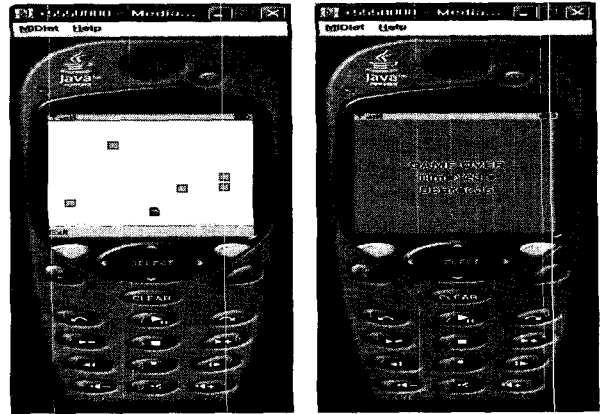
이클립스를 구동하여 File->New->Project를 선택하여 새 프로젝트 선택 창을 띄운다. 여기서 J2ME의 J2ME MIDlet Suite프로젝트를 선택하면 프로젝트 이름을 입력하는 새로운 창이 뜬다. 새 프로젝트의 이름과 저장될 위치를 입

력하면 장치선택 창이 나타난다. 장치그룹 중 J2ME WTK 2.2와 MediaControlSkin장치를 선택하고 Finish를 클릭하면 새 프로젝트가 패키지 탐색 창에 <그림 6>과 같이 나타난다.



<그림 6> 이클립스의 새 프로젝트 생성

왼쪽의 패키지 탐색 창에 있는 프로젝트 이름을 확장하면 여러 디렉터리가 나타나는데 src 디렉터리를 선택하여 필요한 자바파일을 작성한다. FirstDriveMIDlet.java, SecondDriveCanvas.java, ThirdObstacle.java, FourthStopCanvas.java의 네 가지 자바 파일을 삽입하였다. 그리고 res 디렉터리에 자동차와 장애물 그림파일인 Car.png와 Obstacle.png 파일을 삽입한다. 이렇게 게임개발에 필요한 프로그램파일과 그림파일을 생성한 후 게임을 실행하면 된다. Run메뉴에서 Run을 선택하면 MIDlet 선택 창이 나타난다. 여기서 프로젝트 이름과 실행할 MIDlet을 선택하고 Run을 클릭하면 <그림 7>과 같은 게임의 실행화면이 나타난다. 그림의 왼쪽은 게임의 실행화면이고 게임을 플레이 한 후 게임 종료 조건이 성립되면 오른쪽과 같은 종료화면이 나타난다. 실행 시 주의할 사항의 하나로 실행될 컴퓨터 이름이 한글로 되어 있으면 IP Address 접근 에러가 발생하여 실행에러가 발생할 수 있으므로 이때는 컴퓨터 이름을 영문으로 바꾸어서 재실행하여야 한다.



<그림 7> 모바일게임의 실행 사례

5. 결 언

본 연구에서는 모바일 플랫폼의 선두 다름에서 켈컴의 브루우를 제치고 확고한 세계 포준의 자리를 차지하고 있는[43]-[45] J2ME를 이용하여 모바일게임을 개발해 봄으로써 자바모바일게임의 개발과정을 쉽게 이해하고자 함에 그 목적을 두었다. 모바일게임은 주로 J2ME, 브루우, 심비안 등 3개의 주류적 언어로 개발되어 왔으며 이중에서도 자바가 지배적 언어로 2004년 6월 현재 전 세계에 공급된 자바용 핸드셋의 수는 350백만 정도이며 브루우는 이의 5분의 1수준인 54백만 개 정도이다. 심비안은 3.7백만 정도이다.[27] 자바는 사용자 중심언어이며 반면 브루우는 개발자와 제작자 중심언어이다. 따라서 자바는 브루우에 비해 개발비회수가 낮고 네이티브코드가 아닌 바이트코드로 인터프리트 되기 때문에 실행속도도 느리고 3D그래픽과 같은 고속처리가 요구되고 네트워크사업자와의 요구충구시스템 통합 측면에서 브루우보다 다소 가치가 떨어져 보인다. 미국의 주 모바일게임 판매자인 베리존(Verizon)의 경우 모두 브루우를 사용하고 있으며 2004년 기준으로 54백만 대의 판매 중 26.7백만 대를 미국에서 판매하였다. 그리고 기존의 자바 MIDP 1.0은 게임 개발이 제한적이고 단순게임개발만 가능하였으며 서로 다른 핸드셋 간의 이식문제, 사운드, 진동과 같은 기능의 부재 등으로 장치제조사들은 제조사

마다 별도의 추가 라이브러리를 필요로 하였다. 본 연구에서는 이와 같은 J2ME의 취약성을 보완하고 메모리문제, 스크린크기, 포인터사용, 분배API, 프레임레이트, 컬러 등의 문제를 의식하지 않고 게임디자인에만 전념할 수 있도록 제시된 MIDP 2.0을 사용하고 J2ME의 취약점인 속도문제를 고려하여 최소한의 클래스를 사용하였고 변수, 메소드, 클래스의 이름을 가능한 짧게 하여 약간 느리다는 J2ME의 취약부분을 보완하였다. 그리고 MIDP 2.0에서 모바일게임의 개발 특성에 맞게 새롭게 제시된 미들릿, 캔버스, 레이아웃관리자, 스프라이트, 명령리스너(Command Listener), 명령이벤트 등의 새로운 J2ME프레임워크인 게임API와 게임엘리먼트들을 활용하여 게임을 개발함으로써 보다 효율적이고 신속한 모바일게임의 개발방법을 제시하였다. 그리고 본 연구에서는 기존의 모바일게임 개발 방법과 같이 도스(DOS)환경에서 Batch파일을 이용하거나 J2ME의 WTK를 이용하여 모바일 게임을 개발하지 않고 통합개발환경 툴인 이클립스를 사용하여 게임을 개발함으로써 여타의 통합개발 툴보다 쉽고 간단하게 게임을 개발하는 과정을 제시하였다. 아울러 본 연구에서는 모바일게임 사용자의 특성과 사용자의 매력도를 향상시키는 요인과 기술, J2ME의 어떤 부분이 모바일게임의 특성에서 중요하고 중요한 부분을 어떻게 구현하는지에 대해 알아보기 위해 모바일게임의 환경과 현재와 미래의 모바일게임의 시장성, 모바일게임에 대한 정의, 모바일게임의 분류와 가치사슬, 모바일플랫폼의 구조와 모바일게임의 개발에 대한 국내외의 선행적 연구와 이들 연구의 연구 기여도와 향후 연구의 시사점에 대해서도 고찰하였다. 연구결과에 따르면 유비쿼터스 시대를 지향하는 오늘날의 정보기술을 고려할 때 모바일의 생활화는 시대적 대세로 자리 잡고 가고 있으며 향후에도 지속적으로 모바일게임의 수요는 증대할 것으로 예상되나 이에 대한 연구는 미진한 편이었다. 따라서 학문적 관점에서의 모바일게임에 관한 이론적 연구와 개발사태에 관한 연구와 나날이 진보되는 새로운 API를 이용한 모바일게임의 개발과 연구가 계속되어야 할 것이다.

참 고 문 헌

- [1] ARC Research, <http://arcresearch.com>, 2007
- [2] Screendigest, <http://www.screendigest.com/>, 2007
- [3] 한국게임산업진흥원, <http://www.kogia.or.kr/>, 2007
- [4] Roger Riggs, Antero Taivalsaari, Jim Van Peurse, Jyri Huopaniemi, Mark Patel and Aleksu Uotila, Programming Wireless Devices With the Java 2 Platform Microedition, Second Edition, Addison-Wesley, 2003
- [5] Robert Laberge and Srdjan Vujosevic, Building PDA Databases for Wireless and Mobile Development, Wiley Publishing, Inc, 2003
- [6] Geoffrey Elliott and Nigel Phillips, Mobile Commerce and Wireless Computing Systems, Pearson, 2004
- [7] Nick Grattan, and Marshall Brain, Windows CE 3.0: Application Programming, Prentice Hall PTR, 2001
- [8] John Kanalakos, Developing .NET Enterprise Applications, Apress, 2003
- [9] 김성환, 양석호지음, 휴대폰을 위한 모바일 자바프로그래밍: 이클립스기반의 J2ME. WIPI. 게임프로그래밍, 피어슨, 2005
- [10] 최학현, 무선인터넷게임의 기술 동향, 한국 게임학회논문지, 2002년제2권1호,pp.6-13
- [11] Pekka Niskanen, Inside WAP: Programming Applications with WML and WML Script, Addison-Wesley, 2001
- [12] Ludwig Wittgenstein, Philosophical Investigations, Blackwell Publishing, 1953/2002
- [13] Roger Coillois, Les jeux et les hommes (Games and Men), Ghimard, 1957
- [14] Chris Crawford, Chris Crawford on Game-Design, New Riders, 2003
- [15] Katie Salon and Eric Zimmerman, Rules of Play: Game Design Fundamentals, MIT Press, 2003

- [16] Ernest Adams and Andrew Rollings, *Game Design and Development*, PrenticeHall, 2007, pp.5-pp17
- [17] 위키피디아온라인, <http://www.wikipedia.org/>
- [18] 유소란, 모바일게임 시장 및 개발동향, 정보처리 제9권 제3호, 2002년 5월, pp.42-48
- [19] Valentino Lee, Heather Schneider and Robbie Schell, *Mobile Applications: Architecture, Design and Development*, Pearson Education, 2004
- [20] 모바일단말상용화센터, 차세대 모바일기술 국제컨퍼런스 2006, 대구EXCO, 2006년9월 14일-15일
- [21] 최재영, 황병관, 위피플랫폼에 의한 모바일 게임콘텐츠의 개발, 정보통신연구(대구대), 2006년7월제4권1호, pp.97-102
- [22] 홍석형, 이영철, 오선진, PDA기반 온라인 게임의 설계 및 구현, 세명논총(세명대), 2005년제12집, pp.131-157
- [23] 성재경, 김용국, PDA기반 네트워크바둑게임의 유저인터페이스연구, 한국정보과학회 학술발표논문집, 2003년가을호 Vol.30 No.2(1), pp.496-498
- [24] 전준근, 김현수, 모바일 환경용 Java 컨피규레이션 API 구현 및 테스트, 한국정보과학회 학술발표논문집, 2003년가을호 Vol.30 No.2(1), pp.394-396
- [25] 최학현, 무선인터넷게임의 기술동향, 한국 게임학회지, 2002년제2권제1호, pp.6-13
- [26] Nokia, *Mobile Game Graphics: Overcoming the Small Screen Challenge*, Forum.Nokia.Com, Version 10, January 16, 2007
- [27] IGDA Online Game SIG, 2005 Mobile Games White Paper, Game Developer Conference 2005
- [28] Anders Liljedal, *Design Implications for Context Aware Mobile Games*, Thesis for Degree of Master of Science, Upsala University (Sweden), 2002
- [29] S. Bjork, J. Falk, R. Hansson and P. Ljungstrand, *Pirates!: Using the Physical World as a Game Based Paper at Interact*, IFIP TC.13 Conference on Human Computer Interaction, July 2001, Tokyo, Japan, pp.9-13
- [30] B. Thomas, B. Close, J. Donoghue, J. Squires, P. DeBonde, M. Morris and W. Piekarski, *ARQuake: an outdoor/indoor augmented reality first person application*, Wearable Computer, The Fourth Annual Symposium on 2000, pp.139-146
- [31] 김인교, 권강, 유태영 공저, 위피 모바일게임 프로그래밍, 대림출판사, 2005
- [32] 김대회, 위치기반 모바일게임 콘텐츠의 활성화 방안, 한국게임학회논문지, 2004년9월 제4권3호, pp.3-9
- [33] 원유재, 무선인터넷보안기술, 이동네트워크 정보보호기술연구센터, 2001년2월22일, pp.51-70
- [34] Mark Beaulieu, *Wireless Internet: Applications and Architecture*, Addison-Wesley, 2002
- [35] George Lawton, *Moving Java into mobile phones*, IEEE Computer, Vol. 35, No. 6, June 2002, pp.17-20
- [36] Dmitry S. Kochnev and Andrey A. Terekhov, *Surviving Java for Mobiles*, IEEE Pervasive Computing, Vol. 2, No. 2, April 2003, pp. 90-95
- [37] David A. Umphress and James H. Cross, II, Jhilmil Jain, Nischita Meda and Larry A. Baroaski, *Bringing J2ME industry practice into the classroom*, ACM SIGCSE Bulletin, Vol. 36, No. 1, March 2004, pp. 301-305
- [38] 최유희, 신규상, 윤석진, 박창순, 모바일상거래시스템을 위한 아키텍처설계 방법, 정보처리학회지, 2002년5월제9권3호, pp.418-420
- [39] Wassim Itami and Ayman Kayssi, *J2ME Application-layer end-to-end Security for m-Commerce*, Journal of Network and Computer Applications, Vol. 27, No. 1, January 2004, pp.13-32
- [40] Jason Lam, *Considerations for Mobile Game Development*, Wireless News, January 3, 2007
- [41] 오용석, 김명호, 유제정공저, *Mobile Java Programming*, 가남사, 2001

- [42] 강성윤, 이경범, 홍성민공저, 자바모바일프로그래밍:자바 기초에서 모바일까지, 도서출판 대림, 2004
- [43] 임경혜, 모바일시장에서의 JAVA의 역할, 썬마이크로시스템즈 Student Guide, 2000
- [44] 김경한, Java in Wireless 개발자를 위한 지침서: End to End Java 애플리케이션구축하기, Sun Microsystems Monthly Newsletter, 2002년10월, pp.22-25
- [45] 김경한, Java in Wireless 개발자를 위한 지침서: 무선소프트웨어 디자인 테크닉, Sun Microsystems Monthly Newsletter, 2002년7월, pp.28-31

우 원 택 (Won-Taek Woo)



- 1979년2월성균관대학교통계학학사
- 1981년8월연세대학교경영대학원석사
- 1991년2월영남대학교상경대학박사
- 2007년6월~현재대구한의대학교모바일컨텐츠학부부교수
- 관심분야: MIS, 전자상거래, 모바일게임