

# 캠퍼스 안내 앱 개발을 위한 MDA 적용 방법론

김민직, 김행곤  
대구가톨릭대학교 IT 공학부  
e-mail : jsvshb@cu.ac.kr  
hangkon@cu.ac.kr

## MDA Applying Method for Campus Guidance Application Development

Min-Jic Kim, Haeng-Kon Kim  
School of IT Engineering, Catholic University of Daegu

### 요 약

모델기반 아키텍처 MDA (Model Driven Architecture)는 소프트웨어 개발의 추상화 수준을 높이기 위한 최근의 개발 아키텍처이며 완벽한 이해성을 확보하기 위해서 구체적인 구현기술은 반드시 개발자에게 추상적으로 (encapsulation) 숨겨져야 한다. 즉, 개발자는 자동적으로 실행 가능한 시스템으로 변환하는 플랫폼에 구매 받지 않는 모델들 PIM(Platform Independent Model)만을 고려하게된다. 본 논문에서는 MDA 를 캠퍼스 안내 앱에 적용하여 개발함으로써 모바일 디바이스 및 OS 에 독립된 어플리케이션을 개발 방법론을 제시한다.

### 1. 서론

최근 컴퓨터 중심 작업들은 휴대폰으로 급속도로 이동하고 있으며 기존의 휴대폰과 달리 스마트폰은 사용자가 필요한 어플리케이션을 설치함으로써 쉽게 휴대폰에 기능을 추가할 수 있는 커스터마이징 (Customize) 휴대폰을 만들 수 있다. 다양한 기능을 탑재한 휴대폰의 발전 흐름에 발맞추어 네트워크의 기술도 발전하였다. 네트워크 기술은 휴대폰을 항상 인터넷에 연결하여 다양한 정보를 검색하고 찾을 수 있고, 많은 서비스를 사용자가 제공 받을 수 있다. 아울러 모바일 산업은 음성통신 위주에서 데이터 통신 위주로 산업의 중심이 옮겨오고 있고 산업 패러다임의 변화에 따라 음성 중심의 휴대폰산업에서 인터넷 통신을 위한 기기로 탈바꿈하여 스마트폰 산업으로 발전되고 있다.[1]

꾸준한 기술의 발전과 사용자의 편의를 위한 요구 사항을 충족시킴에 따라 여러 멀티미디어 기기들의 대부분의 기능이 휴대폰에 흡수되었고 휴대가 용이하도록 소형화되기를 요구하여 오늘날의 스마트폰으로 발전하게 되었다.

현재 컴퓨터의 어플리케이션은 단순히 컴퓨터 자체의 기능만을 이용하는 것이 아니라 서비스 기반으로 변화되어 항상 연결되어 있는 인터넷을 활용하고 있다. 이러한 기술들은 모바일 디바이스에도 적용될 수 있는데 모바일 디바이스의 특성에 맞게 적용되어야 할 것이다.[2]

모바일 디바이스의 특징에는 ‘제한된 리소스’, ‘물리적 위험’, ‘풍부한 네트워크 연결성’, ‘한정된 배터리 용량’이 있는데 여기서 모바일 어플리케이션을 개발

함에 있어서 가장 많이 고려되는 것은 ‘제한된 리소스’이다. 컴퓨터의 기능을 모바일 디바이스가 일부 가지고 있다고는 하지만 컴퓨터의 자원에 비해 지극히 소형화된 모바일 디바이스는 그 만큼 여러 자원들에 제한을 갖는다. 이러한 특성 때문에 컴퓨터 어플리케이션을 모바일 디바이스에 똑같이 적용을 할 수 없어 모바일 디바이스에 맞게 다양한 도메인에 따라 다양한 방법론을 도입하여 개발을 해야 한다.[1]

본 논문에서는 모델기반 아키텍처 MDA (Model Driven Architecture)는 소프트웨어 개발의 추상화 수준을 높이기 위한 최근의 개발 아키텍처이며 완벽한 이해성을 확보하기 위해서 구체적인 구현기술은 반드시 개발자에게 추상적으로 (encapsulation) 숨겨져야 한다. 즉 개발자는 자동적으로 실행 가능한 시스템으로 변환하는 플랫폼에 구매 받지 않는 모델들 PIM(Platform Independent Model)만을 고려하게된다. 따라서 본 논문에서는 MDA 를 캠퍼스 안내 앱에 적용하여 개발함으로써 모바일 디바이스 및 OS 에 독립된 어플리케이션을 개발 방법론을 제시한다.

### 2. 관련연구

#### 2.1 모바일 앱

어플리케이션 (application)은 컴퓨터 운영체제 위에서 실행되는 모든 소프트웨어를 뜻하는 것으로 줄여서 앱 (app)이라고도 부른다. 모바일앱 (mobile app)은 스마트폰 등과 같은 모바일 단말 위에서 실행되는 소프트웨어를 의미하며 사용자가 아이폰과 안드로이드의 앱스토어 등에서 직접 다운로드 한 후 사용할 수 있다.[3]

앱스토어 (App store)는 각종 모바일앱을 자유롭게 사고 팔 수 있는 온라인 마켓플레이스이다. 애플사의 앱스토어가 상업적으로 성공하면서 범용 OS 사업자, 단말기 제조사, 이동통신사 등 다양한 사업자들이 이 시장에 뛰어들기 시작하여 앱스토어와 유사한 모바일 앱 마켓플레이스를 출시하거나 사전부터 진출 계획을 준비하여 왔다. 주요 모바일 앱 마켓플레이스를 보면 애플사의 앱스토어, 구글의 안드로이드 마켓, 노키아의 오비 스토어, RIM사의 블랙베리 앱월드, 마이크로소프트사의 윈도우 마켓플레이스, SK 텔레콤의 T 스토어, 삼성전자의 삼성앱스 등이 있다.[3]

**2.2 MDA (Model Driven Architecture) 기법**

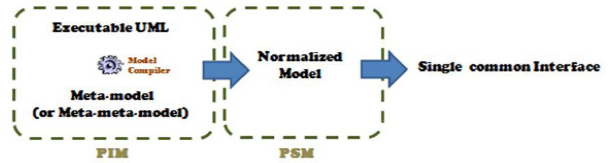
OMG (Object Management Group)에서 정의한 소프트웨어 개발 프레임워크로 시스템 기능 사양의 명세와 특정 기술 플랫폼에 대한 구현사항을 분리하여 프로그래밍 언어, 특정 벤더 플랫폼, 미들웨어 등에 중립적 아키텍처 모델을 가진다. 이기종 플랫폼 간의 언어 독립성을 UML (Unified Modeling Language)의 시장 주도적 힘을 바탕으로 받아들이기 쉬운 방법을 제시하며 탄생한 것으로 2001년 3월 OMG가 처음 발표하였고, 2003년 6월 ‘MDA Guide 버전 1.0’을 공개하였다. 이러한 MDA의 구성은 <표 1>과 같다.[4]

PIM은 플랫폼에 독립된 모델로써 요구사항 수집 및 사용자 정의에 대한 기술을 하며 PSM은 플랫폼에 종속된 모델로써 상세사항 즉, 소스코드와 구현에 관한 작업이 이루어진다. 메타언어는 모델을 작성하기 위한 언어로써 MOF (Model Object Facility)와 호환성을 갖는다. 변환정의는 PIM에서 PSM으로의 전환을 위한 변환 규칙을 정의한 것이고 변환도구는 변환정의에 의해 PIM을 PSM으로 전환하는 것을 말하며 대표적인 도구로 Star UML이 있다.

이러한 MDA의 개발 과정은 (그림 1)에서와 같이 사용자의 UML을 사용하여 PIM을 설계한다.

<표 1> MDA 구성

구성요소	내용
PIM	Platform Independent Model, 특정 기술 플랫폼에 종속되지 않은 모델
PSM	Platform Specific Model, 특정 기술 플랫폼에 종속된 모델
메타언어	모델을 작성하기 위한 메타 (Meta)언어, MOF 호환성
변환정의	PSM 모델로 전환을 위한 변환 규칙을 정의
변환도구	변환 정의에 의해 PIM 모델을 PSM으로 전환



(그림 1) PIM과 PSM 변환 모델

소프트웨어 애플리케이션은 금융에서 국방에 이르기까지 광범위한 영역의 문제들을 대상으로 만들어진다. 게다가 이들 애플리케이션의 복잡성 또한 더욱 커지고 있다. 또한, 소프트웨어 기술 자체의 복잡성이 폭발적으로 증가하고 있으며 개발자들은 HTML, XML, WML, 다계층 아키텍처, J2SE, J2EE, J2ME, .NET, 컴포넌트와 오브젝트 모델, 커넥터, 메시지 브로커 등과 같은 여러 가지 기술을 익혀야만 한다. 소프트웨어 개발의 주요 과제는 생산성을 증대하면서 동시에 유연성과 표준화를 조화시키는 것이 필요하다. MDA가 소개되기 전까지 기존 언어는 이런 요구사항 중 한 두 가지는 맞추었지만 모두를 만족시키지는 못하였다. MDA의 출현으로, 완벽한 유연성을 유지하면서, 표준 환경 내에서 높은 생산성이 가지적으로 현실화되었다. 또한, 수동적으로 작성되는 코드가 적어지고 최상의 사례를 적용할 수 있는 패턴을 사용함으로써 애플리케이션 품질도 개선된다.

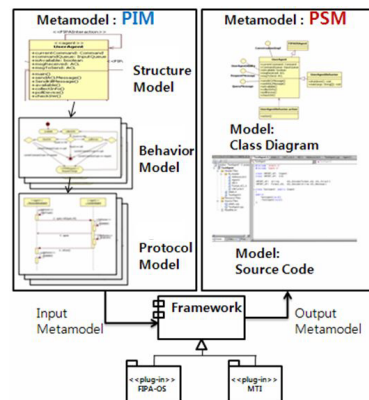
MDA (Model Driven Architecture)는 OMG (Object Management Group)에서 공식 발표된 이후로 소프트웨어 업계에서는 매우 유망한 패러다임으로 받아들여지고 있다. MDA가 소프트웨어의 개발과 유지보수 업무 방식에 대변혁을 가져오는 잠재성을 가졌다고 말할 수 있으며 MDA가 빠른 속도로 대중화되어가고 있다. MDA는 새로운 소프트웨어를 개발하는데 있어 더 큰 생산성을 얻기 위해서 필요하다[5]

**3. MDA 적용 캠퍼스 안내 모바일 앱 개발**

**3.1 개요**

MDA 적용은 개발자가 하위 인프라구조를 크게 고려하지 않고 추상적인 PIM만 고려하여 생산성 및 품질을 향상시킬 수 있다.

본 논문에서는 (그림 2)와 같은 프레임워크를 적용하여 캠퍼스 안내 시스템을 설계 구축하고 평가한다.

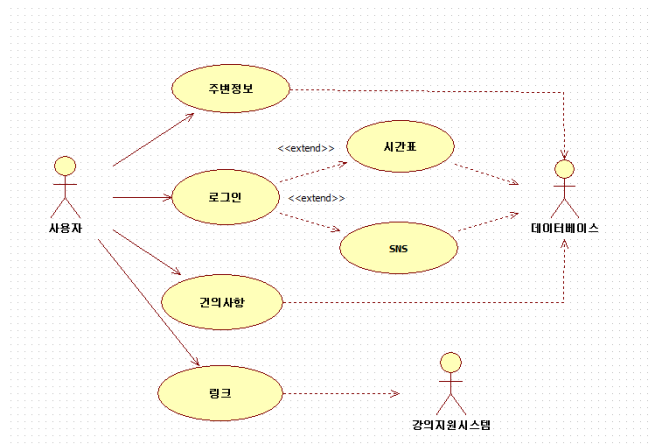


(그림 2) 캠퍼스 안내 앱 개발을 위한 MDA 프레임워크

### 3.2 PIM 적용

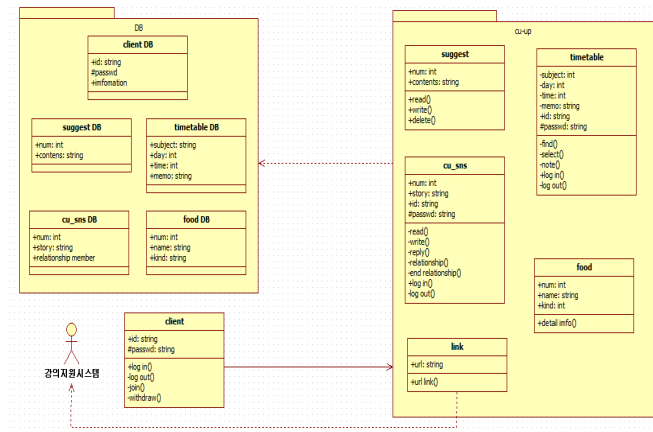
MDA 를 통한 캠퍼스 안내 모바일 앱 개발을 위해 플랫폼 독립 PIM 구조적 행위적 프로토콜 모델을 작성한다.

(그림 3)은 캠퍼스 안내 모바일 앱의 전체적인 UML 로써 유스케이스 다이어그램에서는 사용자가 데이터 베이스와 강의지원시스템을 사용한다.



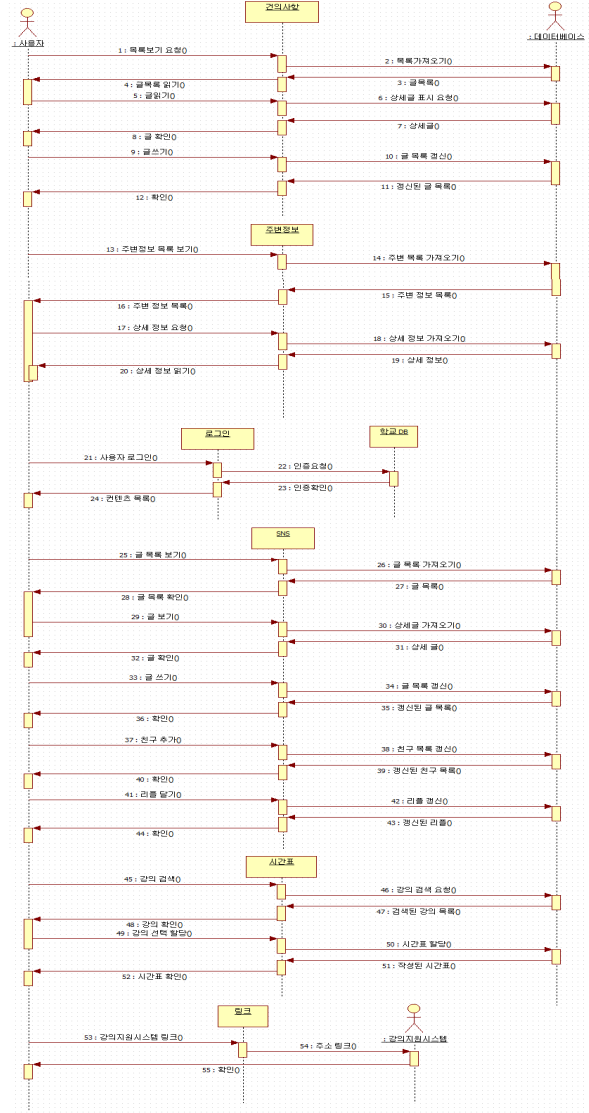
(그림 3) PIM 에 의한 유스케이스 다이어그램

(그림 4)클래스 다이어그램에서는 각종 정보를 제공하기 위한 각 기능별 속성과 동작을 나타낸다.



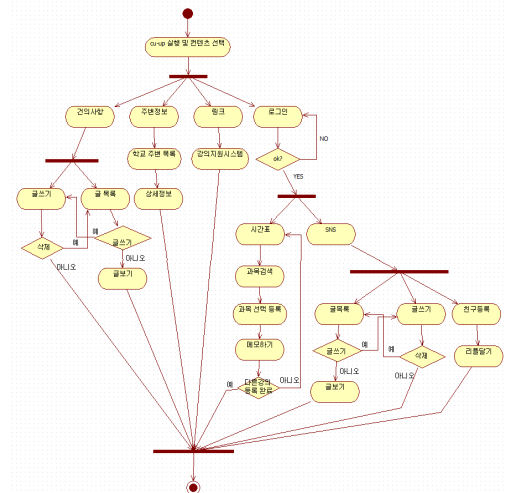
(그림 4) PIM 클래스 다이어그램

(그림 5)의 시퀀스 다이어그램에서는 사용자가 어플리케이션을 사용하게되는 순서를 나타낸다.



(그림 5) PIM 에 의한 시퀀스 다이어그램

(그림 6)의 액티비티 다이어그램에서는 어플리케이션의 실행보투 종료까지의 수행과정을 볼 수 있다.



(그림 6) PIM 액티비티 다이어그램

### 3.3 PSM 적용

(그림 7)은 안드로이드에 종속한 코드로 소스 내용은 캠퍼스 안내 모바일 앱을 구현하기 위한 데이터베이스를 생성하고 어플리케이션 실행 시 수행할 수 있는 기능을 구성한 각 버튼 클릭 시 처리하는 메소드를 나타낸다.

```

* cuUPActivity.java
package acad;
import android.app.Activity;
import android.os.Bundle;
import android.text.Html;
import android.text.method.LinkMovementMethod;
import android.widget.TextView;
public class CuUPActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        TextView btn = (TextView)findViewById(R.id.textView01);

        btn.setText(Html.fromHtml("<a href = 'http://www.cu.ac.kr/board/notice.htm?left_menu=1'>공지사항"));
        btn.setMovementMethod(LinkMovementMethod.getInstance());

        TextView btn2 = (TextView)findViewById(R.id.textView02);

        btn2.setText(Html.fromHtml("<a href = 'http://www.cu.ac.kr/board/free.htm?left_menu=5&menu_sub=1'>열린광장"));
        btn2.setMovementMethod(LinkMovementMethod.getInstance());

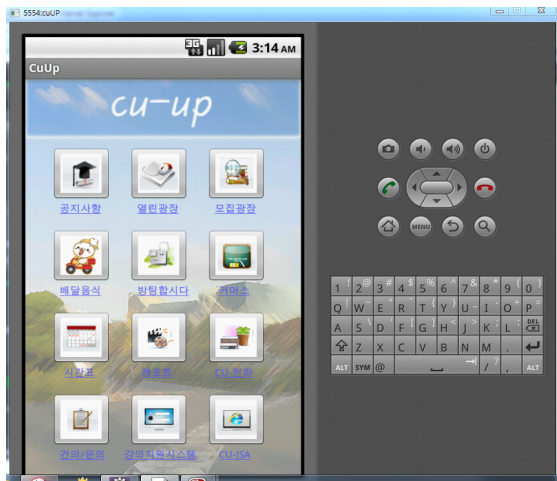
        TextView btn3 = (TextView)findViewById(R.id.textView03);

        btn3.setText(Html.fromHtml("<a href = 'http://www.cu.ac.kr/board/study.htm?left_menu=6&menu_sub=1'>모집광장"));
        btn3.setMovementMethod(LinkMovementMethod.getInstance());
    }
}
    
```

(그림 7) PSM cuUPActivity 코드

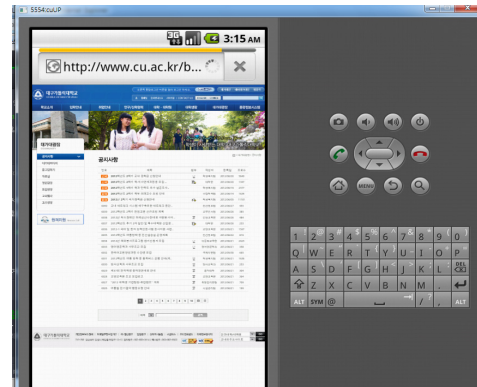
### 3.4 실행 예

안드로이드 에뮬레이터 화면으로 성적어플리케이션을 실행한 결과는 먼저 (그림 8)은 안드로이드 에뮬레이터를 실행하여 캠퍼스 안내 모바일 앱의 실행 초기 화면이다.



(그림 8) PSM 에뮬레이터 실행 화면

(그림 9)는 캠퍼스 안내 모바일 앱을 실행하여 공지사항 화면을 링크한 것으로 앱으로 간단하게 학교의 공지사항을 확인할 수 있다.



(그림 9) PSM 모바일 앱의 공지사항 화면

### 4. 결론

기존의 모바일 어플리케이션은 각 모바일의 플랫폼 별로 플랫폼에 종속된 OS 기반에 플랫폼에 종속된 사용가능한 언어로 개발이 되어야 한다. 즉, 아이폰의 어플리케이션을 안드로이드에도 사용하기 위해서는 안드로이드에 맞게 개발을 해야 하며 그에 따른 시간과 인력이 요구된다. 아이폰 개발에 능숙한 사람이 안드로이드에 능숙하게 될 정의 시간도 요구되고 안드로이드 개발자가 아이폰에 맞는 어플리케이션을 개발하려고 할 때에도 마찬가지로 서로의 플랫폼에 맞는 언어에 대한 이해가 요구되며 이러한 작업은 번거롭다.

본 논문에서는 PIM 방법론 적용과 PSM 을 매핑을 통해 다양한 기술 도메인 적용이 필요하다.

MDA 를 통한 모바일 디바이스 및 OS 에 독립된 어플리케이션 개발이 가능함을 보여준다. 향후에는 MDA 를 통한 PIM PSM 자동화 변환을 개발을 보여주는 결과이다.

### 참고문헌

- [1]이호중, 라현정, 김수동, “서비스 기반 모바일 어플리케이션의 MVC 아키텍처”, 한국컴퓨터종합 학술대회, 제 37 권, 제 1(A)호, pp.74-79, 2010
- [2]전중홍, 이승윤, “차세대 모바일 웹 어플리케이션 표준화 동향”, 전자통신동향분석, 제 25 권, 제 1 호, pp.100-113, 2010
- [3]오정민, 김형환, 문남미, “스마트폰 어플리케이션의 소비 성향 분석을 통한 국가별 선호 연구”, 한국방송공학회 학술발표대회논문집, pp.328-329, 2010.
- [4]김우식, 권오천, 신규상, “Model Driven Architecture 기술 소개”, 전자통신동향분석, 제 17 권, 제 6 호, pp.11-19, 2002.
- [5]이풍석, 장수현, 이은석, “MDA 를 이용한 다중 에이전트 기반 시스템 개발단계에서 재사용성 향상을 위한 프레임워크”, 한국정보과학회 추계학술발표논문집, 제 34 권, 제 2(B)호, pp.78-83, 2007.