

# 멀티 플랫폼 어플리케이션을 개발하기 위한 모바일 스튜디오 구조 연구

주경노\*, 박용성\*\*, 윤찬현\*  
\*한국과학기술원 전기및전자공학과  
\*\*한국과학기술원 그리드미들웨어연구센터  
e-mail : {eu8198, miracle0318, chyoun}@kaist.ac.kr

## A Study on Structure of Mobile Studio for Developing Multi-Platform Applications

Kyung-No Joo\*, Yong-Sung Park\*\*, Chan-Hyun Youn\*  
\*Dept. of Electrical Engineering, KAIST  
\*\*GRID Middleware Research Center, KAIST

### 요 약

여러 모바일 플랫폼들이 등장함에 따라 플랫폼 별로 어플리케이션을 따로 제작하고 따로 관리해야 하는 불편함과 대부분의 개발 환경이 PC 환경이기 때문에 이동 중에는 어플리케이션을 개발할 수 없는 문제점을 해결하기 위해 한번의 소스 코딩만으로 여러 플랫폼에서 적용이 가능하도록 해주며 이동하면서 어플리케이션 개발을 가능케 해주는 모바일 스튜디오의 구조를 제안하고자 한다.

### 1. 서론

최근에 급격하게 스마트폰 사용자가 늘어나면서 iOS, Android 등 다양한 모바일 플랫폼들이 대두되기 시작하였다. 이에 발맞춰 개발자들은 새로운 플랫폼의 개발에 집중하게 되었다. 그러나 어플리케이션을 개발할 때 각 플랫폼마다 환경과 특징이 다르므로 플랫폼별로 따로 어플리케이션을 제작해야 한다는 문제점이 생기게 되었다. 예를 들어 많은 사람들이 이용하는 카카오톡 어플리케이션의 경우 iOS 버전과 Android 버전의 두 가지 버전이 존재하며 개발자 입장에서 같은 프로그램을 두 번 만들어야 하는 불편함이 있다. 또한 어플리케이션 별도 제작에 따르는 추가 비용 역시 뒤따른다. 게다가 어플리케이션을 업데이트 할 때에도 플랫폼 별로 업데이트를 해야 한다는 불편함도 발생한다.

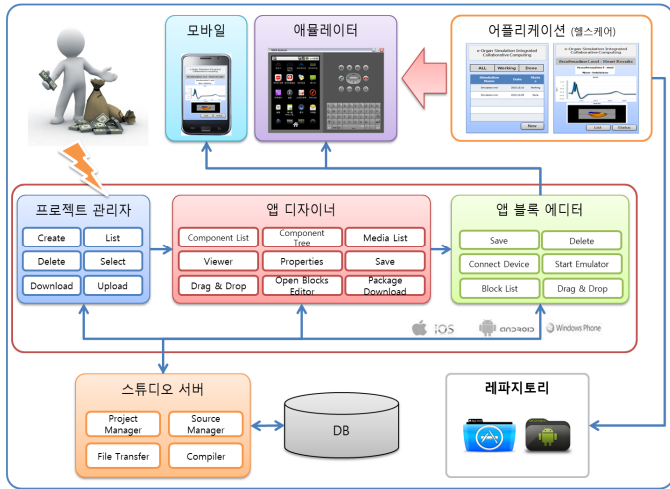
이러한 문제점을 해결하기 위해 모바일 플랫폼 종류에 상관없이 앱을 자유롭게 개발, 배포, 관리할 수 있는 스튜디오가 개발되어야 한다. 모바일 클라우드 기술을 이용하면 클라우드 서버에서 대규모 데이터의 효율적인 분석이 가능하며 또한 어플리케이션 빌드와 같은 복잡한 처리도 맡길 수 있다. 이는 단일소스 이기종 OS 를 지원하며 언제 어디서나 모바일 단말로도 어플리케이션 개발에 도움이 되는 어플리케이션을 개발할 수 있다. 모바일에서 어플리케이션을 개발할 수 있게 되면 이동하면서도 이가 가능하게 된다. 즉 공간에 구애받지 않은 어플리케이션 개발이 가능하게 되는 것이다. 이에 본 논문에서는 모바일 클라우드를 이용하여 멀티 플랫폼 어플리케이션을 개발하기 위한 모바일 스튜디오의 구조를 제안하고자 한다.

### 2. 모바일 스튜디오 요구사항

모바일 스튜디오는 이동 개발 환경과 협업 개발 환경을 구성해 제작자가 스마트하게 어플리케이션을 개발할 수 있도록 스마트워크 환경을 제공하는 데 목적이 있다. 이를 위해 모바일 스튜디오가 만족하고자 하는 세부 요구사항들은 다음과 같다. [1,2,3,4]

- 자유로운 개발환경: 언제 어디서나 모바일 단말기만 있으면 플랫폼 종류에 상관없이 자유롭게 어플리케이션을 개발하고 배포할 수 있도록 해야 한다.
- 클라우드 연동: 모바일의 상대적으로 낮은 성능과 제한된 리소스를 가지고 어플리케이션을 빌드하고 실행시키는 데에는 한계가 있기 때문에 모바일 스튜디오는 대부분의 복잡한 계산을 클라우드에 넘겨 리소스를 최대한 절약하도록 한다.
- 완벽한 클라우드 동기화: PC 와 모바일에서 작성한 소스나 개발환경이 클라우드 내에서 seamless 하게 동기화되도록 설계해야 한다.
- 안전한 멀티카피 관리: 모바일 클라우드 환경에서 Reliable 하고 Secure 하게 소스 및 컨테이너들의 멀티카피를 관리하고 최적의 멀티카피를 선택할 수 있어야 한다. 이는 레퍼지토리(Repository) 기술을 통해 해결할 수 있다.
- QoS 관리: 이동 개발 환경을 실현시키기 위해서는 SLA 를 만족하는 서비스가 제공되어야 한다.

### 3. 모바일 스튜디오 구조 설계



(그림 1) 모바일 스튜디오 구조

그림 1에 모바일 스튜디오의 구조를 도식화하였다. 프로젝트 관리자는 앱 개발을 위한 소스를 프로젝트 단위로 관리해주는 역할을 하며, 프로젝트를 조회, 추가, 삭제하는 기능과 사전에 개발된 소스를 업로드하고 다운로드하는 기능을 갖는다.

앱 디자이너는 어플리케이션 디자인을 도와주는 부분으로 알맞은 컴포넌트를 제공해 사용자로 하여금 드래그&드롭으로 쉽게 어플리케이션을 디자인할 수 있도록 도와준다. 또한 이렇게 디자인된 앱을 서버에 저장하고 PC에 download하도록 하는 기능을 갖는다.

앱 블록 에디터는 디자이너에서 추가한 버튼 등의 컴포넌트에 대한 이벤트를 담고 있다. 또한 개발된 앱을 테스트하기 위한 에뮬레이터와 연결하는데도 필요하다.

스튜디오 서버는 프로젝트 정보들과 소스를 관리해주며 파일을 전송하고 개발된 앱을 다양한 OS에서 사용 가능하도록 컴파일하는 역할을 한다.

에뮬레이터는 실제 폰처럼 작용하는 에뮬레이터를 이용하여 개발된 앱을 테스트할 수 있도록 해준다.

레파지토리는 App Repository, Code Repository, Add-on Library로 구성되어 있어서 앱을 스토어에 보관하고 배포하고, 코드와 라이브러리들을 보관하는 역할을 한다.

그림 2의 Repository Connect 부분을 통해 App을 레파지토리로 배포하고 파일을 전송하고 수신하게 된다.

따라서 이러한 구조를 가진 그림 1의 모바일 스튜디오를 통해 사용자는 쉽게 모바일 환경에서 어플리케이션을 개발할 수 있으며 모바일 스튜디오로 개발된 부분은 클라우드를 통해 그림 2의 OMC Dev Center로 넘어가게 된다. 이곳은 모바일 단말과 PC 등으로 개발된 코드를 관리하고 빌드하는 장소로서 다른 OS나 개발 환경간에 사용될 수 있도록 해주는 Cross-Platform과 어플리케이션을 빌드해주는 App Builder, 어플리케이션을 관리해주는 App Management, 모바일에서 개발을 용이하게 할 수 있도록 자주 쓰는 함수나 명령어들을 보관해 두는 Code Stream, 코드를 관리해주는 Code Management, 모바일 단말이나 PC로부터 수정된 코드를 받는 Mobile Connect, 레파지토리 및 어플리케이션이나 소스코드 등을 주고받기 위한 Repository Connect, FTP를 통해 데이터를 주고받는 FTP Connect와 데이터베이스로 구성되어 있다. 이곳에서 빌드한 어플리케이션은 한 개의 소스코드만으로 여러 플랫폼을 지원할 수 있게 되며 이를 Hybrid App이라고 부른다.

### 4. 결론 및 제언

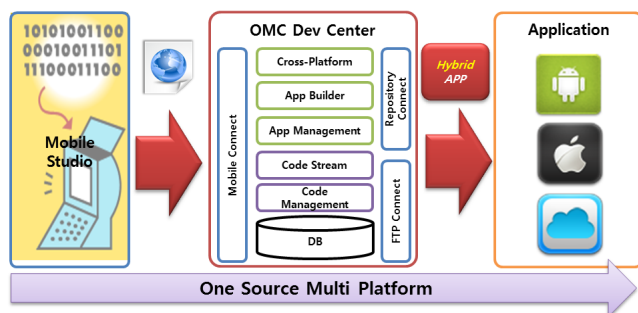
본 논문에서는 기존의 앱을 개발하는데 있어 문제가 되었던 이동 개발 환경과 협업 개발 환경 강화를 위해 클라우드 기술을 이용해서 모바일 클라우드의 구조를 설계하였다. 어플리케이션 빌딩 등 리소스가 많이 사용하는 과정은 클라우드에 위임하면서 모바일 단말의 제한된 리소스 문제를 해결하였으며, PC와 모바일 단말간의 동기화를 통해 길을 걸으면서도 프로그램을 개발할 수 있는 이동 개발 환경과 협업 환경을 구성해 스마트워크 환경을 구축할 수 있음을 확인하였다. 논문에서 제안한 설계의 적합성을 실험하기 위한 테스트베드 구축과 검증, 실험하는 과정이 향후 이루어져야 할 것이다.

### Acknowledgement

본 논문은 2012년도 지식경제부의 재원으로 ‘개인 및 기업 맞춤형 서비스를 위한 개방형 모바일 클라우드용 통합개발환경 및 이기종 단말-서버 간 협업 기술 개발’과 BK21사업의 지원을 받아 수행된 연구임.

### 참고문헌

- [1] 한국정보과학회, 자바기반 분산시스템을 위한 통합개발환경의 구현
- [2] KT 종합기술원, Cross-Platform 모바일 앱 개발 프레임워크에 대한 고찰
- [3] 한국 IT 서비스학회, 오픈소스 기반 통합개발 도구
- [4] 정보통신연구진흥원, 웹서비스 통합 개발 환경 설계 및 구현



(그림 2) 멀티 플랫폼을 지원하는 모바일 스튜디오