

---

# 모델2 Architecture MVC를 이용한 모바일 게임 앱

김현재\* · 이재빈\* · 홍종의\* · 김종호\* · 이욱현\*

\*한북대학교

## The Mobile Game App using Model2 Architecture MVC

Hyun-Je Kim\* · Jae-bin Lee\* · Jong-ui Hong\* · Jong-Ho Kim\* · Uk-hyeon Lee\*

\*Hanbuk University

E-mail : hongjongui@naver.com

### 요 약

요즘 초중고 학교 교육 수준이 높아짐과 더불어 국사의 관한 관심도도 높아지고 있다. 국가 문화재에 대한 정보를 알리기 위해 관심과 흥미를 가질 수 있게 하는 스마트 앱을 개발하고자 한다. 대상자들의 관심도를 높이기 위해 문화재 관련 앱을 게임 형식으로 만들고자 한다. 게임은 현재 스마트폰 사용자의 증가와 더불어 가장 쉽게 접할 수 있으며 게임 속도를 동반한 흥미를 증대시킬 수 있어 문화재 정보 검색을 언제 어디서나 가능하도록 모바일 게임 앱으로 개발하고자 한다. 이 앱은 모델2 Architecture MVC 기법을 사용한다.

### ABSTRACT

Nowadays, with increasing in difficulty level of elementary, junior high and high school education, we are growing more and more interested in the national history. We intend to develop the mobile App to make the people have the concern and interest in the national cultural assets to introduce the information about them. The game can be learned easily and we are increased the interest in the national cultural assets by game speed. We intent to develop the mobile App to be able to search the information about the national cultural assets at anytime and anywhere. This App uses the Model2 Architecture MVC method.

### Keywords

*national cultural assets, mobile game App, Model2 architecture MVC*

### I. 서 론

시중에 많은 마켓에 다양한 종류의 앱이 존재하지만 우리문화재에 대한 앱은 거의 없다고 할 수 있다. 또한 문화재 게임용 앱은 거의 없다고 본다. 이것이 우리가 이 프로젝트를 하게 된 동기이다.

이 앱은 전 연령층이 사용가능하며 지식 축적뿐만 아니라 외국인 등에게 문화재에 관한 홍보가

가능하다. 국가 문화재를 홍보 하게 된다면 지역 관광명소로도 소개가 가능하고 대한민국 수도권 지리 정보 제공에도 도움이 된다. 시험문제 출제를 위해 범위(호선)를 제시할 수 있으며 문화재 탐방하고자 할 때 사전답사와 장소 확인이 가능하다. 또한 교육용 게임 앱을 만들고자 하는 사람들에게 하나의 본보기가 될 수 있고 설계 방법으로도 도움을 줄 수 있어 개발하게 되었다. 마지막으로 알고 있는 친구들이나 지인들과 경쟁을

하면서 지식수준이 향상된다. 앱의 설계를 위해 MVC모델을 사용하게 되었다.

## II. 본 론(제안기법)

다음은 논문 이해를 돕기 위한 MVC에 대한 간략한 설명이다. MVC의 약어는 Model, View, Controller로 나뉘어져 있다. Model은 비즈니스 영역의 상태 정보를 요청 받아서 작업을 하는 것을 뜻한다. View는 말 그대로 보여주는 역할인데 JSP파일의 형식을 보면 HTML과 비슷하다.

모델2는 모델1에 비해 개발 초기의 부하가 높고, 아키텍처에 대한 높은 이해를 요구하지만 재사용성이 높고 확장이 모델1에 비해 용이하다는 장점이 있다. 특히 MVC 각 영역의 역할이 구분되도록 보여준다. 단점이 있다면 개발이 더 어려워지며 비용이 증가한다는 것이다. 이 장점들 때문에 프로젝트 개발에 적합하다고 생각하여 어려움에도 불구하고 모델2 구조를 선택했다.

MVC 모델을 사용해 받은 정보를 SDCard에 저장하여 하나의 Activity에서 구현하는 기법을 제안한다. Mobile Phone에서 Touch Event를 통해 받은 정보 즉, 호선정보와 난이도 정보를 Class에 담아 Controller에게 Request를 한다. Request 받은 Controller는 Business Logic을 이용하여 Controller가 가지고 있는 데이터(호선, 난이도)를 가공하고 그 정보를 얻은 후 Query문을 작성하여 Oracle DB에 접속한다. 접속 후 Image와 문제를 받아 Controller에게 넘기고 Controller는 다시 View에게 넘겨 Request한 곳으로 Response해 데이터를 넘기는 동작을 취하고 주요기능으로는[게임전]과 [게임중]인 2개로 나누어진다.

### 1. Before The Game

상세한 내용이기 전에 간략히 설정하면 기본적으로 [게임전]으로는 Main Activity에 출력된 <게임 시작>인 Button을 클릭하면 호선 선택으로 넘어가게 되며 하고자 하는 호선에 있는 문화재의 정보를 알기 위한 전단계이다 호선은 1호선부터 특수 호선 까지 Setting 되어 있으며, 자유로이 선택이 가능하다. 호선 선택 후 난이도에 관한 문제들은 [초급], [중급], [고급]을 선택 할 수 있다. 문제를 선택 시 나오는 문화재 Image는 난이도의 별도의 상관없이 공통적으로 Image가 부여되고 문제 형식만 달라진다.

중요한 것은 처음 시작 시 초급 난이도만 사용

가능하다. 초급에서 문제를 풀어 일정량의 경험치를 축적(축적)해 요구되는 Level에 도달하면 한 단계 높은 중급을 풀 수 있게 된다. 고급 또한 조건에 해당된다면 풀 수 있고 고급까지 오게 되면 자신이 원하는 난이도를 직접 선택하는 技能(기능)이 추가된다.

### 가. 게임 전

Game을 시작할 때 호선과 난이도를 선택 한다 이때 호선 선택과 난이도 선택에 대한 정보는 Android의 Activity의 Stack 구조를 이용해 MainActivity에서 같은 패키지 안에 있는 Information Class를 Instance화 시켜 생성한 후 활성화 된 Activity인 난이도 선택에서 Information Activity에 있는 Get, Set Method를 이용해 사용자의 게임 정보를 저장한다

이때 Information Class에 있는 변수로는 호선 선택 정보를 저장할 수 있는 LineNumber, 난이도 선택 정보를 저장할 수 있는 LevelNumber, 맞춘 개수를 저장 할 수 있는 ONumber, 총 개수의 맞은 문제 수를 저장 할 수 있는 TotalNumber가 있다.

이때 TotalNumber는 사용자가 프로그램 종료 전까지 몇 개의 문제 풀었는지 알려주는 정보다. 만약, 사용자가 다음 레벨로 넘어가는 상황이라면 ONumber를 TotalNumber에 더하는 형식인 TotalNumber += ONumber가 된다. 해당 레벨에 문제를 풀만큼 더하여 끝나는 시점까지 가지고 있다가 게임 종료와 함께 텍스트 문서로 저장한다.

이렇게 굳이 하나의 Class에 정보를 담지 않아도 Intend를 이용해 키 값을 전달하던가 아니면 Access Modifier를 이용해 변수를 공유하는 등등 여러 방법이 있지만 이렇게 하는 이유는 사용자의 성향을 알아보기 위해서다 사용자의 성향을 통해 이용한 문제를 분배하는 업데이트를 해주기 위해서 따로 Information Class를 만든 것이다. 즉, 성향에 따른 사용자 정보를 수집하기 위해서 이다.

Information Class의 GetMethod를 사용하여 각 정보를 얻은 후 ServerMethod를 이용해 Server로 보낸다. Server는 Request 받은 정보를 이용해 Query문을 처리하여 처리된 정보를 Response를 통해 SDCard로 보내진다. 이때 Query문에는 사용자의 선택 정보를 파일 이름으로 검색한다

파일 이름은 사용자의 호선 선택정보(1~9) + 사용자의 난이도 선택정보(1~3) + 문제번호(01~20), 문제에 대한 이미지(01~20), 문제에 대한 정답 TEXT(1~20)이다.

다음과 같이 표시 할 수 있다.  
 1101Problem(1호선 초급 1번문제),  
 1101Image(1호선 초급 1번 이미지),  
 1101RightAnswer(1호선 초급 1번 정답)  
 여기서 문제번호, 이미지번호, 정답 TEXT는 Server쪽에서 반복문을 이용해 1~20까지 Response해 준다. Response 받은 정보는 동시에 SDCard에 저장되고 사용자의 활성화 된 Activity에서 이 처리한 정보를 사용하게 된다

이때 활용되는 Problem Activity는 단 한 개만 존재한다. 그 이유는 여러 Activity를 이용해서 게임을 할 경우 구현은 쉬우나 메모리에 부하가 걸려 앱의 속도가 저하되고 유지보수의 어려움이 있어 다음과 같이 Application을 구현했다.

Activity의 재귀호출을 이용해 활성화된 Activity의 Intent를 자기 자신으로 하여 호출하는 방법을 선택 했다. 이를 통해 Activity 하나를 사용하기 때문에 이전의 다중 Activity를 이용한 문제점을 보완하고 전체적인 프로그램의 효율이나 Intent로 인한 메모리 사용을 절감할 수 있다

Activity가 하나이므로 고려해야할 사항인 문제 이미지, 정답을 어떻게 처리할 것인가에 관한 문제가 생겼다. 이 문제에 대한 해결 방안은 SDCard에 저장된 정보의 파일명으로 해결했다 Activity에 static 멤버변수 i로 정의하고 Intent 될 때 마다 i++한다. Activity가 호출 될 때마다 i값은 증가하고 i가 20이 되면 자동적으로 다음 레벨로 넘어간다. 이때 Server에 Request를 보내어 그 호선에 다음 레벨에 관한 문제를 받는 형식이다

게임 도중 Back Button을 누를 경우 Instance 된 Information Class정보를 SetMethod를 이용해 0으로 초기화 시킨다. 이유는 Activity의 Stack구조 때문에 게임을 재시작할 경우 초기화 되지 않은 정보가 사용될 수 있기 때문이다. 이후 게임종료를 선택하면 Instance된 Information Class의 정보가 앱에 있는 사용자 정보에 Text로 저장되어 그 동안의 누적정보를 확인할 수 있다

## 2. After The Game

### 가. Playing Game

[게임중]에서는 <힌트추가>, <Pass>가 생성(생성)되어있다. 힌트추가 Button을 누르면 자기 자신을 호출한 것이 아니라 자기 Hint가 적용된 Activity를 호출한다. 따라서 난이도 별로 2개씩 6개의 Activity가 있다. Hint Button을 Click하면 hint가 적용된 Activity를 호출하고 다음 문제는 원래 Activity를 호출하는 재귀 호출 형식으로 구현하는 구조이다.

Pass Button을 누르면 static 변수 i만 증가하고 Instance된 Information Class에 정보들을 변하지 않고 다음 문제로 넘어가는 구조이다 [초급]의 문제형식은 O,X형태로 문제의 내용을 보고 정답 버튼 2개중 한 개를 클릭하면 된다. [중급]은 4지선답형으로 객관식으로 출제(출제)된다. 중급부터는 난이도를 고려해 Hint가 생성되며 보기가 어려울시 Click하여 4개의 보기 중 2개를 제거해 좀더 쉽게 답을 선택 가능하게 한다. [고급]은 주관식으로 출제되며 문제가 더 어려워지고 자세한 내용이 들어간다. 힌트는 초성을 보여주며 자신이 생각한 답과 비교하여 답을 정답에 가깝게 類推(유추)가 가능하게 만들어준다. 모르는 문제는 풀지 않고 <Pass>를 선택하여 다음 문제로 넘어갈 수 있다. 넘어간 문제는 나중에 풀 수 있도록 미루어 준다. 단, <Pass>로 문제를 넘어가게 되면 해당된 문제의 경험치는 상승하지 않는다.

### 나. Finished Game

<오답체크>는 게임이 끝난 후 본인이 풀었던 문제 現況(현황)을 보여준다. 틀린 상태는 표시되어 보여준다. 부가적으로 Main Activity의 설정과 내 정보가 있는데 설정에서는 Game Setting인 <볼륨 설정>, <시간설정>, <게임정보>, <개발사 정보>를 확인할 수 있다. 볼륨설정에서는 효과음과 배경음악을 調節(조절)할 수 있고, Time은 주어진 설정된 상태로 본인에 맞게 증가 및 감소시킬 수 있다. <게임정보>에서는 게임에 관한 소개를 보여주며 개발사 정보에서 만든이, 만든날짜, 출처 등이 기재 된다. [내정보]는 <레벨>, <경험치>, <맞춘수>, <틀린수>를 확인할 수 있다. 레벨은 문제를 풀어 일정량의 경험치를 획득해 도달할 경우 向上(향상)된다. 사용자가 현재까지 푼 문제의 정답과 오답 수를 확인할 수 있다.

## 3. Mobile Phone Request 방식 및 Response 방식

### 가. Mobile Phone Request 방식

Mobile Phone에서 Request 시 호선 선택 및 난이도 선택 정보를 Http Protocol의 Service Method를 이용하여 정보를 전달한다. 이때 Service Method를 사용하는 이유는 doGet, doPost 방식을 모두 호환하기 위해 사용한다. 사용자가 난이도 정보를 선택하는 순간 Thread.sleep을 이용한 Mobile Phone Request Time 확보한다. 이때 요청된 정보는 User가 선택한 정보로써 Service Method에 저장되어 요청된다.

### 나. Mobile Phone Response 방식

Request방식과 동일하게 Http Protocol을 사용하며 Web Server에서 Servlet에 있는 Service

Method를 이용하여 Mobile Phone SDcard에 저장하는 방식이다.

다. SDcard 용량 초기화

Finished Game시 해당된 Activity는 Delete Folder를 호출해 SDcard에 있는 File을 모두 삭제한다. 이 때의 Logic은 TargetFolder.listFiles()를 이용한다. .length를 이용한 크기를 알아낸 후 Delete Method를 이용해 Size가 0이 될 때까지 삭제하는 Logic이 SDcard 초기화 하는 방식이다.

4 Application 실험 및 평가

가. Application 실험

본 Application은 Intent를 쓰지 않아 하나의 Activity를 만들어 연동하여 화면에 출력하게끔 하지 않기 때문에 보다 적은 Activity를 구성하게 된다. 그러므로 수정할 부분이 발생하면 범위가 한정되어 있어 손쉽게 삽입, 삭제 등등 수정할 수 있는 점이 장점이며 Activity가 차지 할 수 있는 용량이 적어진다. 그리고 Application은 초등 및 중학교에서 잘 활용할 수 있다. 시험문제 출제를 위해 범위(호선)를 提示(제시)하여 보다 중요한 문화제를 공부하여 역사 공부에 관한 문제를 出題(출제)가능하다. 또한 문화제 장소를 探訪(탐방)하고자 할 때 이 Application에서 장소가 확인 가능하다.

나. Application 평가

알고 있는 知人(지인)들과 競爭(경쟁)을 하면서 지식수준이 향상된다. 전 연령층이 사용가능하며 지식 축적뿐만 아니라 외국인 등에게 문화제에 관한 弘報(홍보)가 가능하다. 국가 문화제를 홍보하게 된다면 지역 관광명소로도 紹介(소개)가 가능하고 대한민국 수도권 지리 정보 提供(제공)에도 도움이 된다.

또한, 교육용 게임 Application을 만들고자 하는 사람들에게 하나의 본보기가 될 수 있고 설계 방면으로도 도움을 줄 수 있다. 마지막으로 이 Application은 교육용으로 만들었고 문화제청의 도움을 받아 商業的(상업적) 目的(목적)을 취할 수 없다.

III. 결 론

SDcard 용량 초기화, Application 실험, Application 평가 3가지의 결과에서 본 논문에서 제안하는 재귀호출 Activity가 적용된 경우 다중 Activity보다 Mobile Phone의 용량을 최소화 및 유지보수의 편리함을 확인할 수 있다. 재귀 Activity를 사용함으로써 추가적 용량 사용 없이

사용자 체험성능을 개선시킬 수 있고 이를 통해서 궁극적으로 시스템 성능개선 효과를 가져 온다고 할 수 있다.

참고문헌

[1] Do it! 안드로이드 앱 프로그래밍  
 [저자명] : 정재곤  
 [출판사] : 이지스퍼블리싱  
 [2] 단계별 예제로 배우는 안드로이드 프로그래밍  
 [저자명] : 한동호  
 [출판사] : 제이펍  
 [3] 안드로이드 프로그래밍 정복1  
 [저자명] : 김상형  
 [출판사] : 한빛미디어  
 [4] 안드로이드 프로그래밍 정복2  
 [저자명] : 김상형  
 [출판사] : 한빛미디어  
 [5] Learning PHP,MySQL & JavaScript  
 [저자명] : 로빈 닉슨  
 [출판사] : 에이콘  
 [6] Oracle JDBC JSP Programming  
 [저자명] : 이건영  
 [출판사] : 네오솔루션  
 [7] JSP 2.2 Servlet 3.0(모델2로 구현하는 자바 웹 프로그래밍  
 [저자명] : 오정원  
 [출판사] : 헤지원