Final Lab

Mobile Programming

유시환 교수님

PB TUNES

32112186 고영일

32112139 김태형

32112200 송시호

32112219 황형언

목차

1. 개발 환경
2. 개발 목적
3. SCRUM
4. 코드 소개
5. 진행 과정
6. 최종 완성본
7. 추후 추가 예정 기능
8. 어려웠던 점, 보완점
9. 후기

10. 참조

1. 개발환경

윈도우 8.1 pro K

RAM : 4GB

프로세서 인텔 코어 i5-5200 CPU 2.20GHz

개발 Tool : 안드로이드 스튜디오 1.3.0

개발 기종 : galaxy nexus 7

1. 개발 목적

흥겨운 야구장에서 응원가를 몰라 즐기지 못하는 사람들을 위해, 각 선수들의 응원가를 쉽게 찾을 수 있게 해준다.

팀 별로 선수 사진을 제공하고, 그 사진과 응원가를 연결함으로

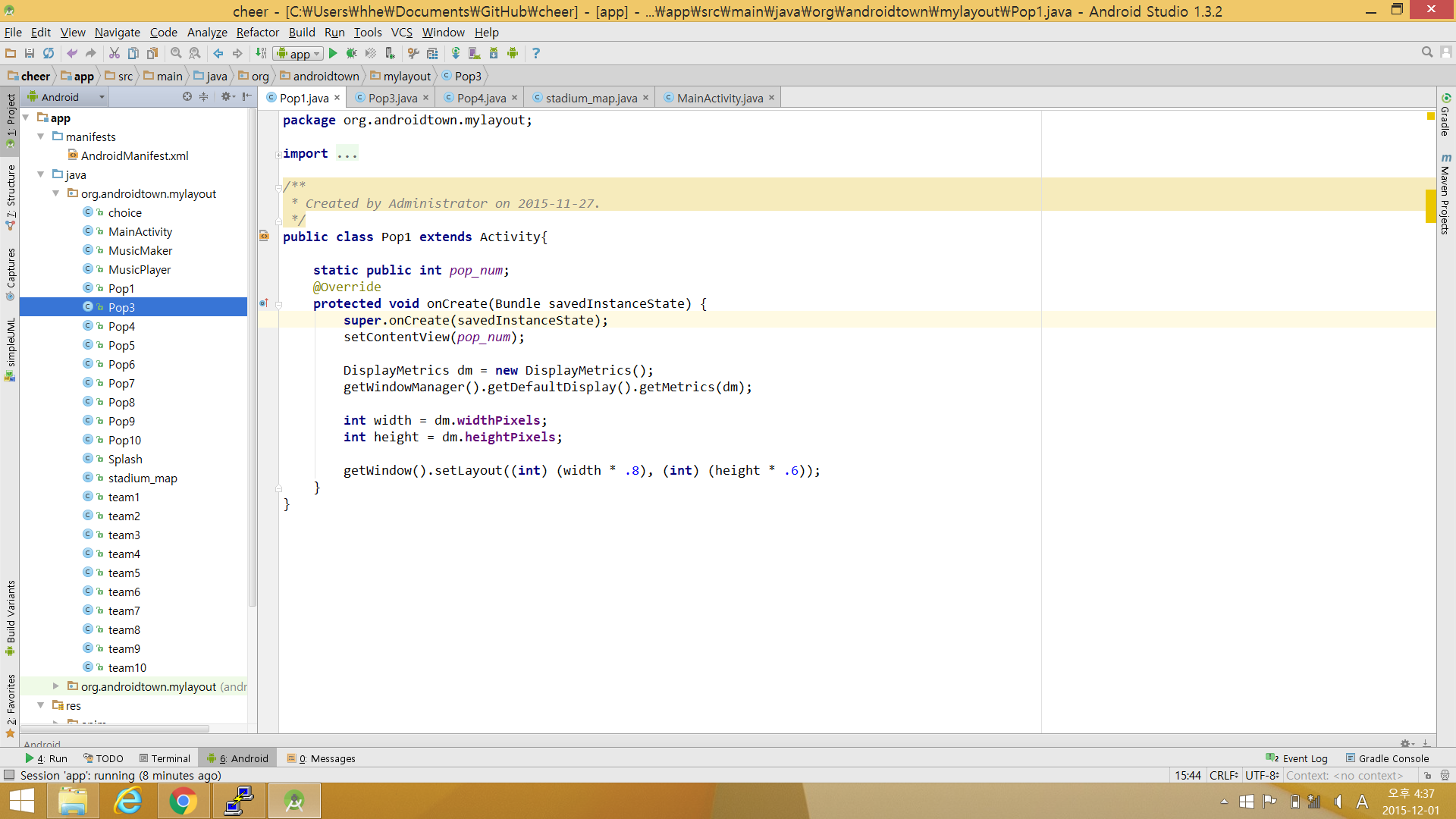
검색기능을 만들어 원하는 선수를 편하게 찾을 수 있게 한다.

1. SCRUM

* 각 xml 파일 당 class가 한 개씩 있었는데 static변수를 사용해 10개의 클래스를 한 개의 클래스로 줄여 중복된 코드 사용을 없앴다. 버튼마다 static 변수의 값을 다르게 설정해주고, team1클래스 또는 pop1클래스에서 변수의 값을 확인 후 열고자 하는 xml파일을 바꿔주었다.

static 변수 pop\_num 사용해 pop2~pop10 클래스 제거

static 변수 team\_num 사용해 team2~team10 클래스 제거



* MediaPlayer 객체를 선수 인원만큼 9개 만들고 따로 따로 실행시켰던 코드를 Music\_Maker 클래스를 만들어 하나의 객체로 노래 파일만 바꿔 주며 실행 가능하도록 했다.

원래 코드 ->

music = MediaPlayer.create(this, R.raw.kimtaekyun);

music.setLooping(false);

button = (ImageButton) findViewById(R.id.imageButton);

//**총 9개 생성 music ~ music9**

public void button(View v) {

if(music1.isPlaying()){

music1.stop();

try {

music1.prepare();

} catch (IllegalStateException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

}else{

music1.start();

}

} **//총 9개의 똑 같은 코드 사용**

if(music!=null ||music2!=null ||music3!=null ||music4!=null ||music5!=null ||music6!=null ||music7!=null ||music8!=null ||music9!=null){

music.release();

music=NULL;

}

* 원래 120MB였던 앱 용량을 320kbps였던 음원을 192kbps로 압축하여 90MB로 줄였다.
* 선수 프로필 화면을 띄우기 위해서 90개의 xml파일이 필요할 거라고 생각했었는데 setImageResource함수와 setText함수를 이용해 하나의 xml 파일로 모든 선수의 프로필 화면을 띄울 수 있었다.
* 원래 팀 별 선수 응원가만 재생시키는 앱을 만드려 했지만 너무 단순해서 기능을 추가하면 좋겠다는 피드백을 받고 선수 검색을 통한 프로필과 응원가 재생, 구단 별 홈구장의 위치정보를 추가 기능으로 넣었다.

1. 코드소개

Choice.java

**public class** choice **extends** AppCompatActivity{  
  
 **private** Back\_Key **back\_handler**;  
  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_choice***);  
  
 startActivity(**new** Intent(**this**, Splash.**class**));  
  
 ImageButton choice1 = (ImageButton) findViewById(R.id.***choice1***);  
 choice1.setOnClickListener(**new** View.OnClickListener() {

@Override

//첫 번째 버튼을 눌렀을 때 MainActivity.class가 켜지도록 한다  
 **public void** onClick(View v) {  
 Intent intent1 = **new** Intent(choice.**this**, MainActivity.**class**);  
 startActivity(intent1);  
 }   
 });

.

.

.

**back\_handler**=**new** Back\_Key(**this**);   
  
}

//뒤로가기 버튼을 눌렀을 때 실행  
 **public void** onBackPressed(){  
 **back\_handler**.onBackPressed();  
 }   
}

MainActivity.java

듣고자 하는 팀을 선택한다.

Choice.java와 거의 동일하다.

MusicMaker.java

**public class** MusicMaker {  
 String **name**;  
 MediaPlayer **song**,**remember**;  
 **int incount**=0;  
 String **prev\_name**;  
  
 **public** MediaPlayer play(MediaPlayer a,String b,Context context,**int** count) {  
 **song** = a;  
 **name** = b;  
 **int** c;  
 **incount**=count;

//Team1.java에서 선수마다 설정해준 int형 변수 c를 받아 음악 파일을 설정한다.

**switch** (b) {  
  
 **case "kimtaekyun"** :  
 c = R.raw.***kimtaekyun***;  
 **break**;  
 **case "choijinhang"**:  
 c = R.raw.***choijinhang***;  
 **break**;

**.**

**.**

**.**

**case "parkkihyuk0"** :  
 c = R.raw.***parkkihyuk0***;  
 **break**;  
 **case "parkkyungsoo0"** :  
 c = R.raw.***parkkyungsoo0***;  
 **break**;  
  
 **default**:  
 c = 0;  
 **break**;  
 }

//처음 실행할 때 MediaPlayer 생성  
 **if**(**incount**==0){   
 **song**=MediaPlayer.*create*(context,c);  
 }

//두 번째부터는 실행되는 Mediaplayer 종료 후 새로 생성  
 **if**(**prev\_name**!=**name**){   
 killPlayer();  
 **song**=MediaPlayer.*create*(context,c);  
 **incount**=0;

}

//Play되고 있을 때는 재생을 멈추고 현재 위치를 기억한다.  
 **if**(**song**.isPlaying()){  
 **song**.stop();  
 **try** {  
 a.prepare();  
 } **catch** (IllegalStateException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }}

//Play되고 있지 않을 때는 생성된 MediaPlayer를 재생한다.  
 **else**{  
 **song**.start();  
 }  
  
  
 **prev\_name**=**name**;  
  
 **return song**;  
 }  
  
  
 **public int** count(){  
 **incount**++;  
 **if**(**incount**==10)**incount**=0;  
 **return incount**;  
 }

//MediaPlayer를 release 하고 null로 만들어 주는 함수이다.  
 //새로운 노래를 시작할 때 또는 프로그램을 종료할 때 사용한다.  
 **protected void** killPlayer(){  
 **if**(**song**!=**null**){  
 **song**.release();  
 **song**=**null**;  
 }  
 }

}

Play\_song.java

**public class** Play\_Song **extends** Activity {  
 **public int c**;  
 **static public** String *name*;  
 MediaPlayer **music**;  
 **int position**;  
  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***play\_song***);  
  
 ImageView image=(ImageView)findViewById(R.id.***imageView4***);

//선수 이미지  
 TextView text=(TextView)findViewById(R.id.***play\_song***);  
 TextView profile=(TextView)findViewById(R.id.***profile***);  
 //선수 프로필  
 **switch** (*name*){  
 **case "himenes"**:  
 **case "히메네스"**:  
 text.setText(**"LG 트윈스"**);  
 **c** = R.raw.***himenes5***;  
 image.setImageResource(R.drawable.***himenesname***);

//setImageResource: 검색한 선수에 따라 이미지를 바뀌게 한다.  
 profile.setText(**"신장/몸무게 185cm/92kg\n"** +  
 **"포지션 3루수\n"** +  
 **"투타 우투우타"**);  
 **break**;  
 **case "leebyunggyu"** :  
 **case "이병규"**:  
 text.setText(**"LG 트윈스"**);  
 **c** = R.raw.***leebyunggyu5***;  
 image.setImageResource(R.drawable.***leebyunggyuname***);  
 profile.setText(**"신장/몸무게 185cm/85kg\n"** +  
 **"포지션 외야수\n"** +  
 **"투타 좌투좌타"**);  
 **break**;

**.**

**.**

**.**

**default**:  
 **c** = 0;  
 **break**;  
 }   
  
 image.invalidate(); //바뀐 이미지가 화면에 뜰 수 있도록 설정해준다  
 **music** = MediaPlayer.*create*(**this**, **c**); //MediaPlayer생성  
 **music**.setLooping(**false**);  
  
}

//Play 버튼

**public void** play(View v) {  
 **music**.start();  
 }

//Pause 버튼  
 **public void** pause(View v){  
 **if**(**music**.isPlaying()) {  
 *// 재생중이면 실행될 작업 (정지)* **music**.pause();  
 **try** {  
 **music**.prepare();  
 } **catch** (IllegalStateException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }

//Stop 버튼  
 **public void** stop(View v){  
 **if**(**music** !=**null** ) {  
 **music**.pause();  
 **music**.seekTo(0);  
 }  
  
 }   
  
 **protected void** killplayer(){  
 **if**(**music**!=**null**){  
 **music**.release();  
 **music**=**null**;  
 }  
 }

//뒤로 가기 버튼을 눌렀을 때 노래 재생이 멈추도록 killplayer()를 불러온다  
 **protected void** onDestroy(){  
 **super**.onDestroy();  
 killplayer();  
 }   
  
}

Pop1.java

**public class** Pop1 **extends** Activity{  
  
 **static public int** *pop\_num*; //Stadium\_Map.java에서 선택한 팀 정보를 받는다  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(*pop\_num*);  
  
 **switch**(*pop\_num*){

//1번 팀 선택 시 1번 팀의 홈구장 위치 정보로 변경한다

**case** R.layout.***pop\_team1***:   
 TextView pop1=(TextView)findViewById(R.id.***pop1***);  
 pop1.setText(Html.*fromHtml*(**"<a href=\"http://map.naver.com/index.nhn?pinTitle=%ED%95%9C%ED%99%94%EC%83%9D%EB%AA%85%EC%9D%B4%EA%B8%80%EC%8A%A4%ED%8C%8C%ED%81%AC&pinType=site&pinId=11831114&mapMode=0\">구장 위치</a>"**));  
 pop1.setMovementMethod(LinkMovementMethod.*getInstance*());  
 **break**;

**.**

**.**

**.**

**default**:  
 **break**;  
  
 }  
  
 DisplayMetrics dm = **new** DisplayMetrics();  
 getWindowManager().getDefaultDisplay().getMetrics(dm);  
  
 **int** width = dm.**widthPixels**;  
 **int** height = dm.**heightPixels**;  
  
 getWindow().setLayout((**int**) (width \* .8), (**int**) (height \* .6));  
 }   
}

Search\_Player.java

**public class** Search\_Player **extends** Activity{

//선수 목록을 배열로 가지고 있다  
 String[] **items** = {**"himenes"**, **"leebyunggyu"**, **"leejinyoung"**, **"ohjihwan"**, **"parkjigyu"** … }

**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***search\_player***);  
  
 **final** AutoCompleteTextView edit = (AutoCompleteTextView) findViewById(R.id.***edit***);  
  
 Button del=(Button)findViewById(R.id.***delete\_button***);

//선수 검색을 하고 돌아왔을 때 autocompleteTextview에 남아있는 선수 이름을 공백으로 초기화한다  
 del.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 edit.setText(**""**);  
 }  
 });

//AutoCompleteTextView 클릭 시에 선수 명단이 나타나도록 한다  
 edit.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 edit.showDropDown();  
  
 }  
 });

//Adapter 선언  
 edit.setAdapter(**new** ArrayAdapter<String>(**this**,  
 android.R.layout.***simple\_dropdown\_item\_1line***, **items**));  
  
 edit.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {

@Override  
 //검색 시 자동으로 나타나는 선수 이름을 클릭하면 선수 프로필 Activity가 실행되도록 한다

**public void** onItemClick(AdapterView<?> parent, View view, **int** position, **long** id) {  
 Play\_Song.*name* = edit.getText().toString();  
 startActivity(**new** Intent(Search\_Player.**this**, Play\_Song.**class**));  
  
 }  
 });  
  
 }  
}

Stadium\_Map.java

각 팀의 위치를 지도로 파악할 수 있다.

Choice.java와 거의 동일하다.

**public class** stadium\_map **extends** AppCompatActivity {  
  
  
  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***stadium\_map***);  
   
 ImageButton map\_team1 = (ImageButton) findViewById(R.id.***map\_team1***);  
 map\_team1.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Pop1.*pop\_num*=R.layout.***pop\_team1***;  
 startActivity(**new** Intent(stadium\_map.**this**, Pop1.**class**));  
 }  
 });

.

.

.

Back\_Key.java

뒤로 가기 버튼을 눌렀을 때 앱이 바로 종료되지 않고 한 번 더 의사를 물어보도록 하는 기능

**public class** Back\_Key {  
 **private long backKeyPressedTime** = 0;  
 **private** Toast **toast**;  
  
 **private** Activity **activity**;  
  
  
 **public** Back\_Key(Activity context) {  
 **this**.**activity** = context;  
 }  
  
 **public void** onBackPressed() {  
 **if** (System.*currentTimeMillis*() > **backKeyPressedTime** + 2000) {  
 **backKeyPressedTime** = System.*currentTimeMillis*();  
 showGuide();  
 **return**;  
 }  
 **if** (System.*currentTimeMillis*() <= **backKeyPressedTime** + 2000) {  
 **activity**.finish();  
 **toast**.cancel();  
 }  
 }  
 **private void** showGuide() {  
 **toast** = Toast.*makeText*(**activity**, **"\'뒤로\'버튼을 한번 더 누르시면 종료됩니다."**, Toast.***LENGTH\_SHORT***);  
 **toast**.show();  
 }  
}

1. 진행과정

9월

1. 소프트웨어 디자인 및 UI 디자인
2. Requirement 작성
3. class diagram 그리기
4. Do it 안드로이드 스튜디오 강의 시청
   * 안드로이드 스튜디오 사용법
   * Layout별 사용법
   * 기본 위젯 사용법

10월

1. splash 클래스 생성
2. 각 팀의 로고와 선수 명단 등 팀정보 수집
3. Team1~10클래스 생성

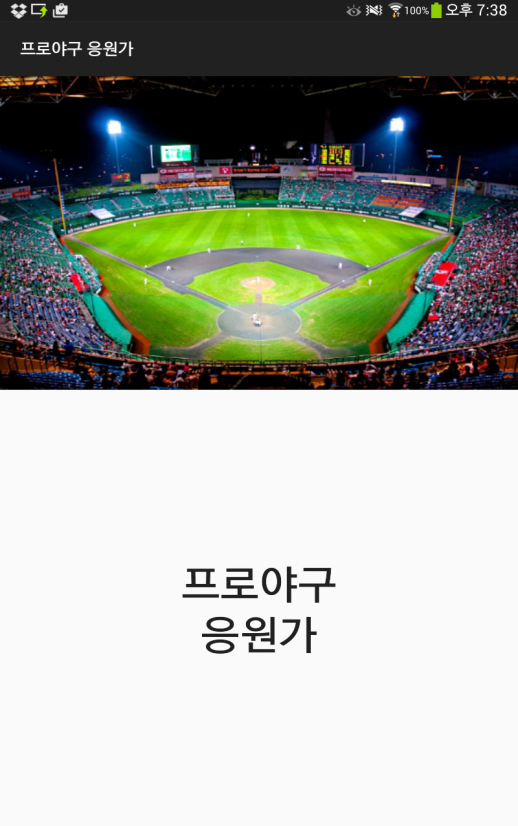
11월

1. Music\_Maker 클래스 생성
   * 불필요한 MediaPlayer객체 사용을 줄이고 코드 길이 줄임
2. 중간 프레젠테이션 준비 및 발표
3. 피드백 및 추가 구현에 대한 회의
   * 선수 검색 기능과 구장 위치정보를 추가
   * 선수 프로필 추가
4. Choice클래스, Stadium\_Map클래스, Search\_Player클래스, Pop1~10클래스 생성
5. 클래스 통합을 통해 중복 코드 제거
   * Team1~10을 Team1클래스 한 개로 통합
   * Pop1~10을 Pop1클래스 한 개로 통합

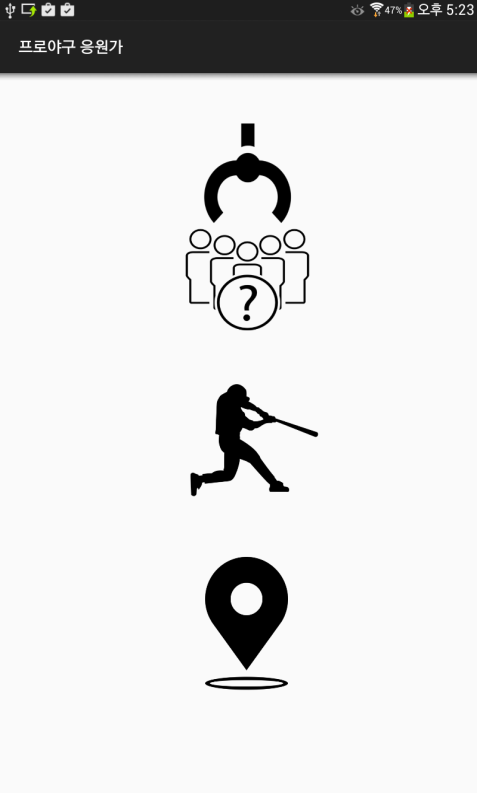
12월

1. 마무리 및 PPT 제작
   * Back\_Key 클래스 생성해 앱 종료 시 수행 구체화
   * 파이널 프레젠테이션 준비
2. 앱 디자인
   * 모든 텍스트, 이미지 통일화
   * 배치 및 여백 수정
3. 최종 완성본(Screen shot)

1)첫 Lording page



2)메뉴 선택 page



3) 팀 선택 page



4) 팀 내 선수 선택 page



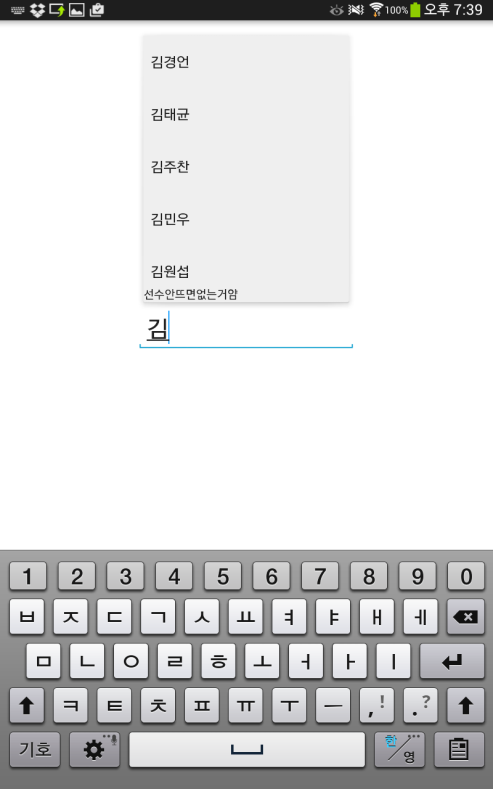
5) 위치 별 팀 선택 page



6) 팀별 구장 정보 page



7) 선수 검색 page



3) 선수 검색시 프로필 & 응원가 화면



1. 추후 추가 예정

1) 시즌순위

2) 후보 선수를 포함한 모든 선수 응원가 및 프로필 등록

3) MAP에 있는 구장 정보에 좌석 정보 추가

4) 구장 위치를 GPS를 이용해 링크를 통해서가 아닌 앱 자체에서 확인 가능하도록 함

5) scalable layout을 적용해 모든 기기에서 사용 가능하도록 함

1. 어려웠던 점, 보완점

모든 것을 처음 시작한다는 점

* 자바와 안드로이드 스튜디오 사용의 미숙
* Github을 이용한 코드 공유의 어려움
* Layout에 대한 이해가 부족해 디자인을 하는 데에 있어서 마음대로 할 수가 없었던 점
* 현실과 이상의 괴리

1. 후기

이번 수업에서 자바와 안드로이드 스튜디오를 처음 접했었는데, 자바와 안드로이드 스튜디오의 이해 및 이용법을 몰라서 인터넷과 책을 통해 처음부터 공부를 하여서 수업의 이해 및 application을 만드는데 어려움을 겪었다. 또한 어떤 application을 만들지 결정할 때 다양한 의견들이 있어서 의견을 조율하는데 어려움을 겪었다. Class design을 할 때도 처음 하는 것 이여서 그릴 때 많은 어려움을 겪었다. 하지만 다같이 고민하고 협력하여 문제를 하나하나 해결해 나갈 때마다 성취감을 느끼고, application의 기능을 하나씩 추가 할 때마다 눈으로 보여져서 재미도 있고 신기했다. 처음으로 application을 만든 이 수업이 자바와 안드로이드 스튜디오를 공부할 좋은 기회였고, 방학 때 자바와 안드로이드 스튜디오를 좀 더 공부하여 좀더 능숙하게 application 개발을 할 수 있도록 할 것이다.

1. 참조

동영상 강의 : Do it 안드로이드 앱 프로그래밍

<https://www.youtube.com/watch?v=vWaibTqTFcM>

미르 IT 블로그:

<http://itmir.tistory.com/>

Filip Vujovic 동영상 강의:

<https://www.youtube.com/watch?v=fn5OlqQuOCk>