





여행정보비교 트레블러”

팀원 : 한창엽 - 20122298 (팀장)
김다영 - 20184253
이현호 - 20162168



목차

- 
- 
- 1 작품 개요
 - 2 작품 소개
 - 3 작품기능 및 주요 코드
 - 4 향후 계획 및 기대효과

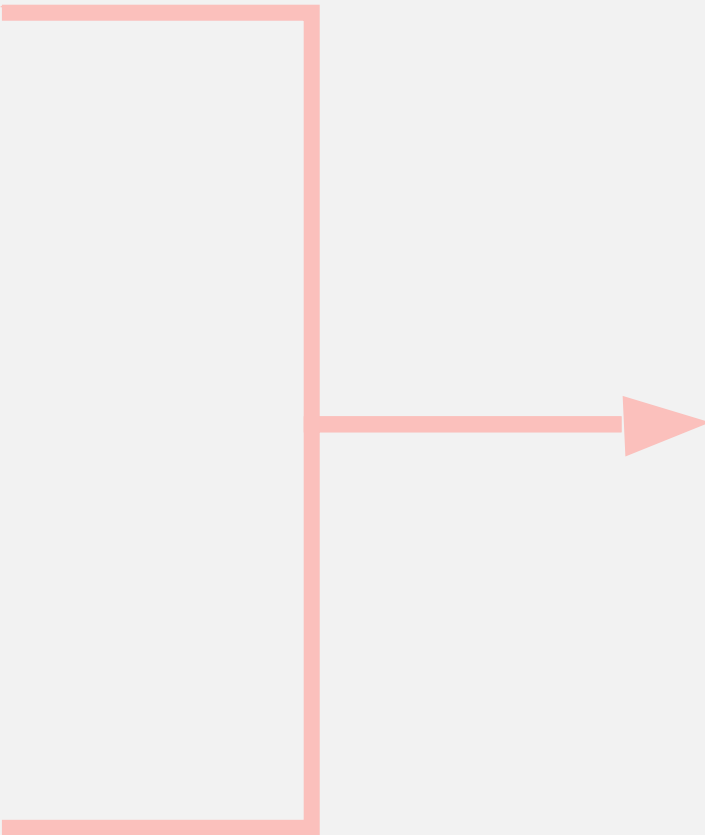
1. 작품개요

1) 작품개요

- 1인 또는 3명 이내로
해외여행을 계획하는 사람

- 해외여행이 처음이거나
경험이 적은 사람

- 해외 여행지에 대한
정보가 필요한 사람



◦ 해외의 정보, 후기를
한번에 확인하여 비교
하는 어플이 없어 불편을
겪을 것

1) 작품개요



- 후기, 신뢰성 있는 실시간정보(날씨, 환율) 관광명소, 교통정보 등의 정보를 한번에 확인 가능한 어플을 제작해 편의성을 제공
- 여행국가 선정에 어려움이 있을 때, 설문지를 통해 여행국가를 추천

2. 작품소개

2) 작품소개

(1) 작품기능

1. 연령별 많이 방문한 국가순위
2. 만족도 상위 10개 국가
3. 설문지를 통한 여행국가 추천
4. 다양한 국가정보 제공
5. 실시간 국가정보 제공
6. 국가정보 비교
7. 후기 작성
8. 국가별 후기 제공
9. 관심후기 선정

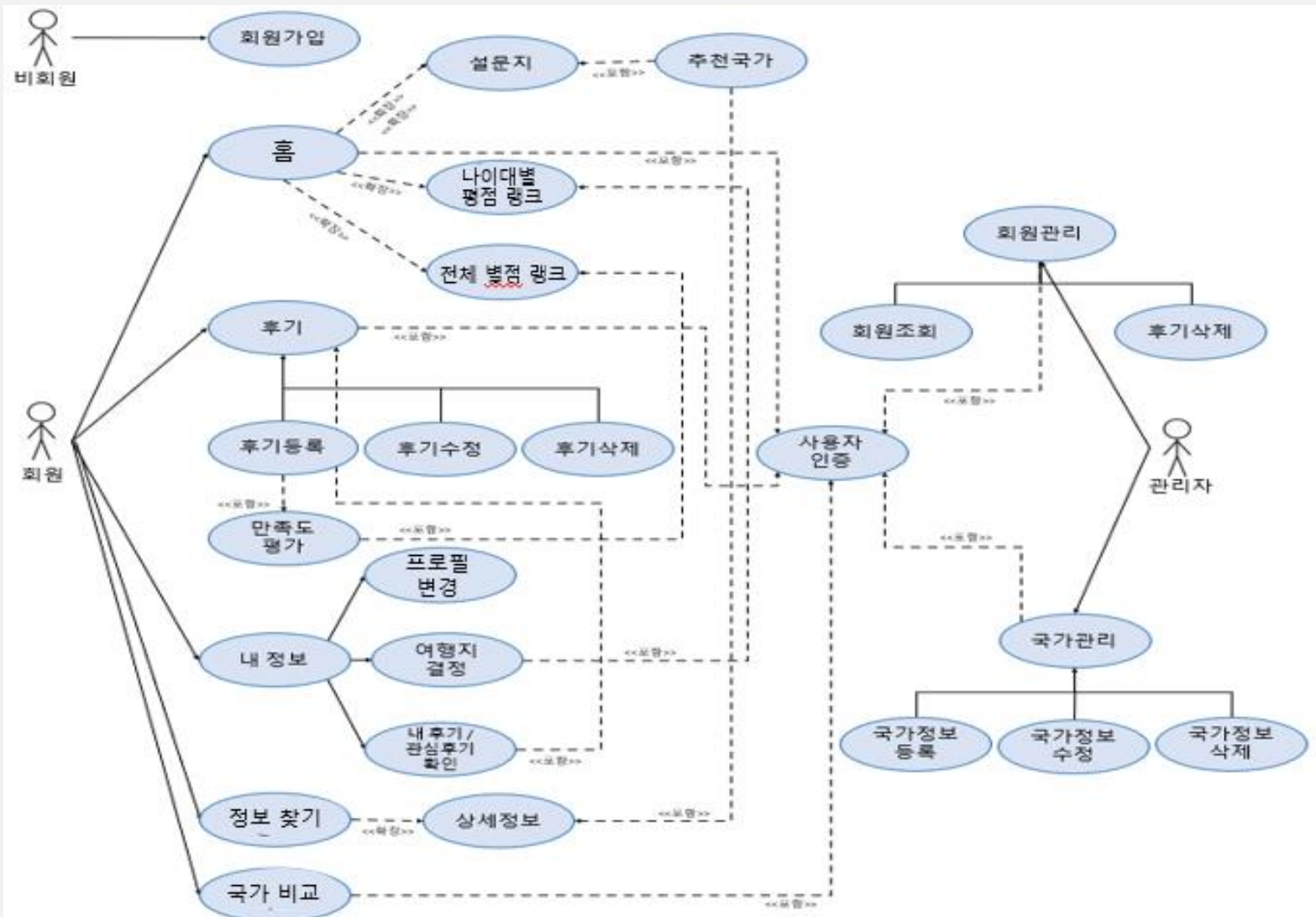
2) 작품소개

(2) 시스템 구성도



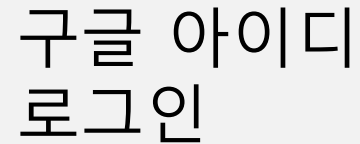
2) 작품소개

(3) Use Case Diagram



3. 작품기능 및 주요 코드

- 로그인(구글 통합 로그인)



자체
회원가입

트레블러

회원가입

이름

이름을 입력 하세요

이메일

이메일을 입력 하세요

비밀번호

비밀번호를 입력 하세요

연령

가입

3) 작품기능 및 주요 코드

◦ 로그인(구글 통합 로그인)

트레블러



계정 선택

트레블러 앱을 계속 이용하려면 계정을 선택하세요.



김다영
dayoung7833@gmail.com



가나다
testingrse@gmail.com



kim다영
eekedud0202@gmail.com



나는야다영
eekedud0202@gmail.com

👤 다른 계정 추가

계속하기 위해 Google에서 사용자의 이름, 이메일 주소,
프로필 사진을 트레블러에(와) 공유합니다.

비밀번호 찾기

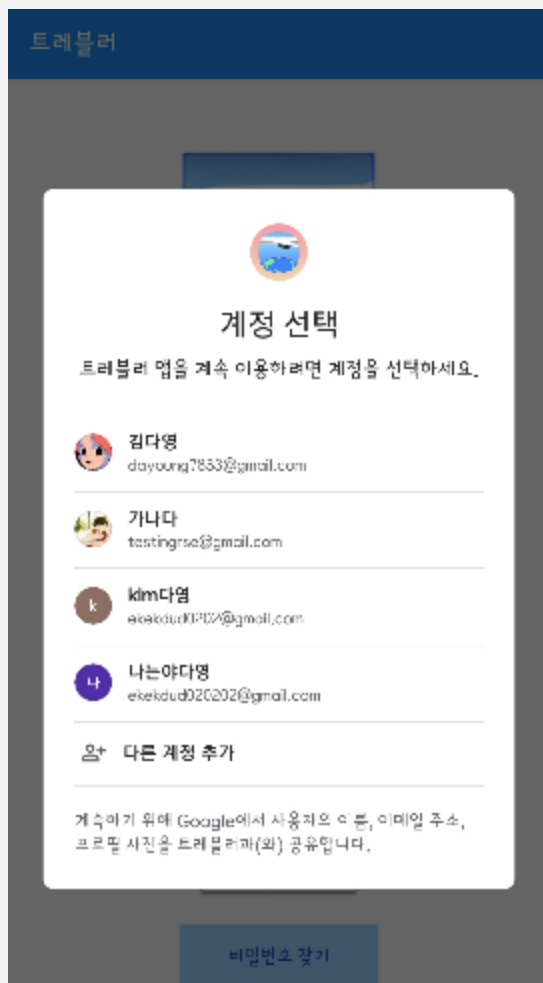
```
// Google 로그인을 앱에 통합, GoogleSignInOptions 개체를 구성할 때 requestIdToken을 호출
GoogleSignInOptions googleSignInOptions = new GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN)
    .requestIdToken("273658160109-tg1oj31g5k0onp4ouaksh47jn3umrk5k.apps.goog...")
    .requestEmail()
    .build();

GoogleSignInClient googleSignInClient = GoogleSignIn.getClient(activity, googleSignInOptions);
buttonGoogle.setOnClickListener((view) -> {
    Intent signInIntent = googleSignInClient.getSignInIntent();
    startActivityForResult(signInIntent, RC_SIGN_IN);
});
```

```
// 구글로그인 버튼 응답
if (requestCode == RC_SIGN_IN) {
    Task<GoogleSignInAccount> task = GoogleSignIn.getSignedInAccountFromIntent(data);
    try {
        // 구글 로그인 성공
        final GoogleSignInAccount account = task.getResult(ApiException.class);
        firebaseAuthWithGoogle(account);
        ref.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                for(DataSnapshot snapshot : dataSnapshot.getChildren()){
                    String email = snapshot.child("email").getValue(String.class);
                }
            }
        });
    } catch (ApiException e) {
        // 구글 로그인 실패
    }
}
```

3) 작품기능 및 주요 코드

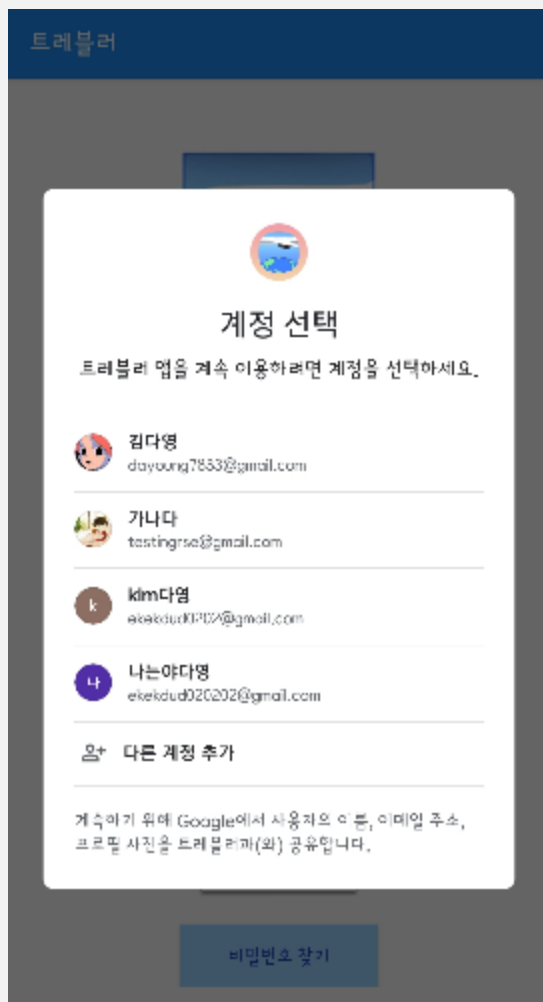
◦ 로그인(구글 통합 로그인)



```
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
    for(DataSnapshot snapshot : dataSnapshot.getChildren()){  
        String email = snapshot.child("email").getValue(String.class);  
        if(!account.getEmail().equals(email)){  
            a = false;  
        } else if(account.getEmail().equals(email)){  
            a = true;  
            break;  
        }  
    }  
    if(a == false){  
        Toast.makeText( context: Login.this, text: "아이디가 없습니다! 회원가입 해주세요!",Toast.LENGTH_SHORT).show();  
    } else if(a == true){  
        Toast.makeText( context: Login.this, text: "아이디가 있습니다! "+email,Toast.LENGTH_SHORT).show();  
        Intent intent = new Intent( packageContext: Login.this, MainActivity.class);  
        startActivity(intent);  
    }  
}  
  
@Override  
public void onCancelled(@NonNull DatabaseError databaseError) {  
  
}  
});
```

3) 작품기능 및 주요 코드

◦ 로그인(구글 통합 로그인)



```
// 사용자가 정상적으로 로그인한 후에 GoogleSignInAccount 개체에서 ID 토큰을 가져와서
// Firebase 사용자 인증 정보로 교환하고 Firebase 사용자 인증 정보를 사용해 Firebase에 인증합니다.
private void firebaseAuthWithGoogle(GoogleSignInAccount acct) {
    AuthCredential credential = GoogleAuthProvider.getCredential(acct.getIdToken(), null);
    firebaseAuth.signInWithCredential(credential)
        .addOnCompleteListener(activity: this, (task) -> {
            if (task.isSuccessful()) {
                // 로그인 성공
                Toast.makeText(context: Login.this, text: "구글 인증 성공", Toast.LENGTH_SHORT).show();
            } else {
                // 로그인 실패
                Toast.makeText(context: Login.this, text: "인증 실패", Toast.LENGTH_SHORT).show();
                // updateUI(null);
            }
        });
}
```

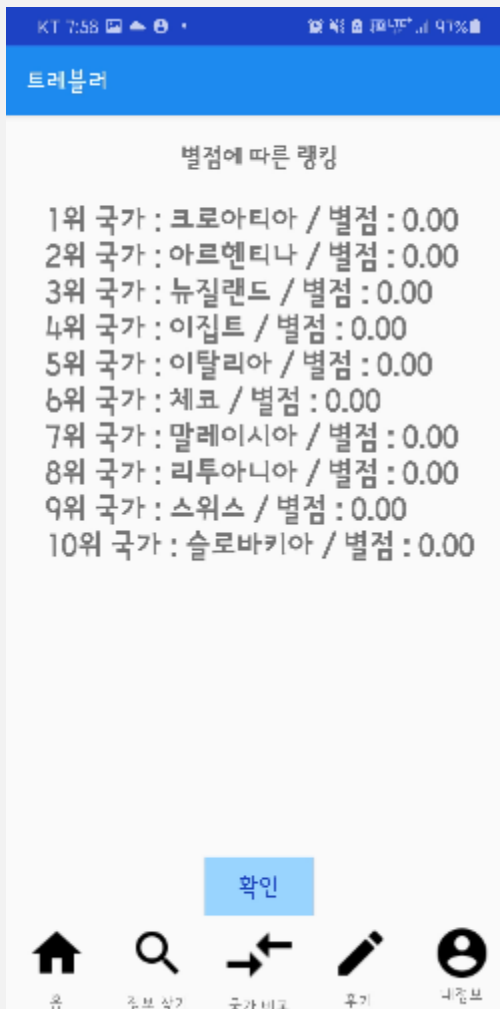
파이어 베이스 유저 인증 부분

3) 작품기능 및 주요 코드

◦ 홈 화면 → 별점 순위

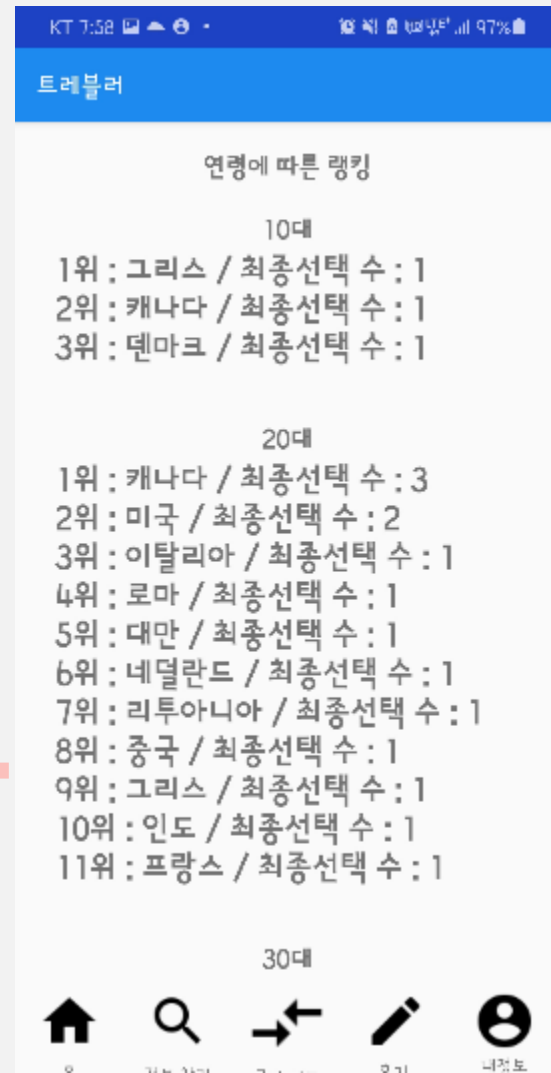


로그인 후
홈 화면



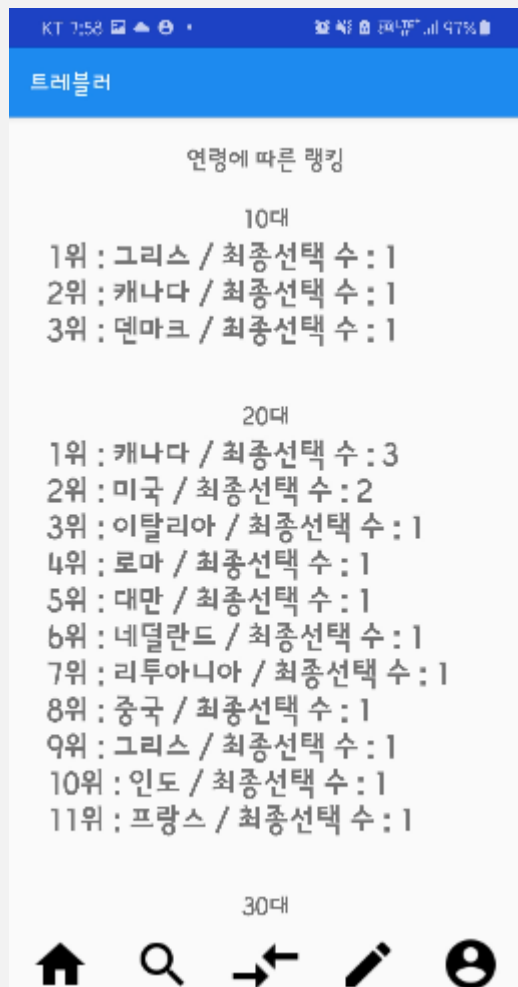
전체 별점 순위

연령별 별점 순위



3) 작품기능 및 주요 코드

◦ 홈 화면 → 별점 순위

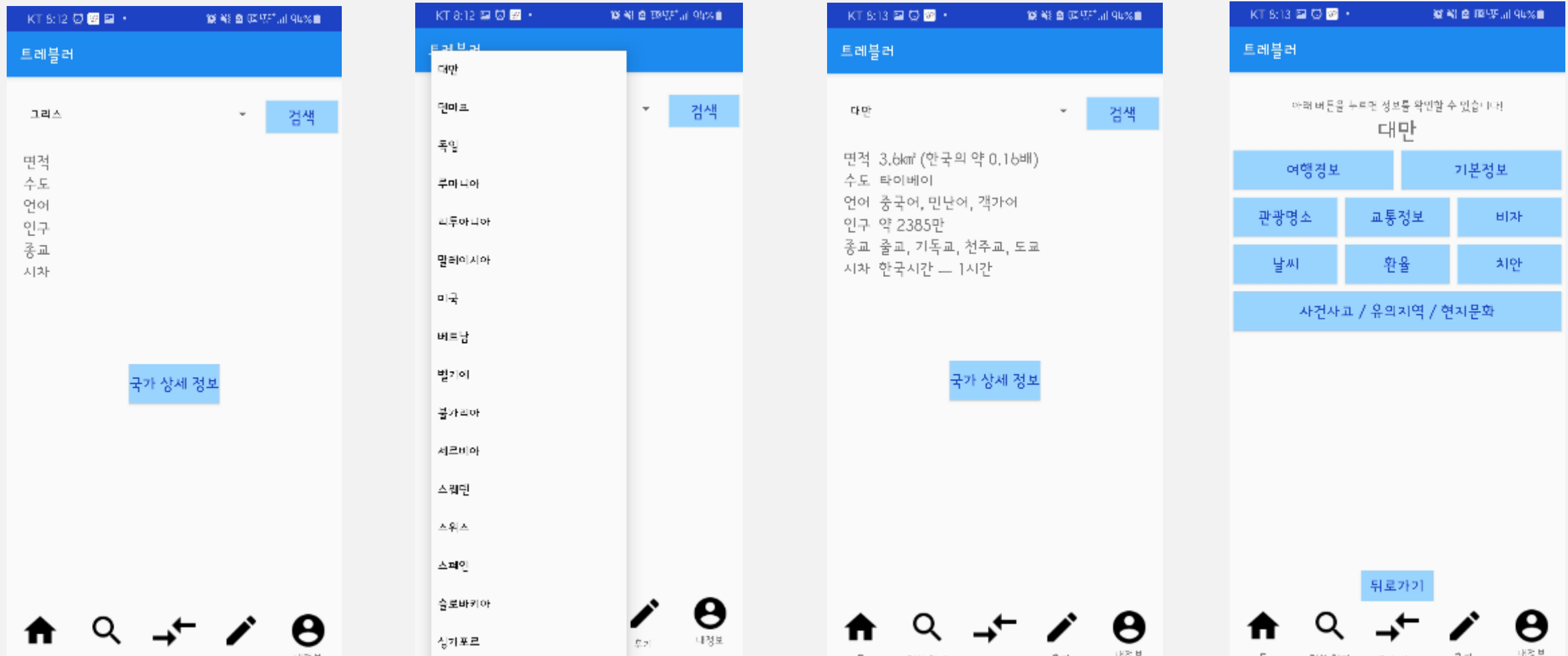


연령별 별점 순위

```
private void age10() {
    for (int i = 0; i < exlist_age10.size(); i++) {
        if (count1.containsKey(exlist_age10.get(i))) {
            count1.put(exlist_age10.get(i).toString(), count1.get(exlist_age10.get(i)) + 1);
        } else {
            count1.put(exlist_age10.get(i).toString(), 1);
        }
    }
    rank_list1 = new ArrayList<Map.Entry<String, Integer>>(count1.entrySet());
    Collections.sort(rank_list1, (Comparator) (obj1, obj2) -> {
        return obj1.getValue().compareTo(obj2.getValue());
    });
    Collections.reverse(rank_list1);
    int i = 1;
    if (rank_list1.size() > 10) {
        for (int a = 10; a < rank_list1.size(); a++) {
            rank_list1.remove(a);
        }
    }
    for (Map.Entry<String, Integer> entry : rank_list1) {
        String c1 = entry.getKey();
        String s1 = Integer.toString(entry.getValue());
        String s = i + "위 : " + c1 + " / " + "최종선택 수 : " + s1 + "\n";
        age_rank_10.append(s);
        i++;
    }
} else if (rank_list1.size() < 11) {
    for (Map.Entry<String, Integer> entry : rank_list1) {
        String c1 = entry.getKey();
        String s1 = Integer.toString(entry.getValue());
        String s = i + "위 : " + c1 + " / " + "최종선택 수 : " + s1 + "\n";
        age_rank_10.append(s);
        i++;
    }
}
```

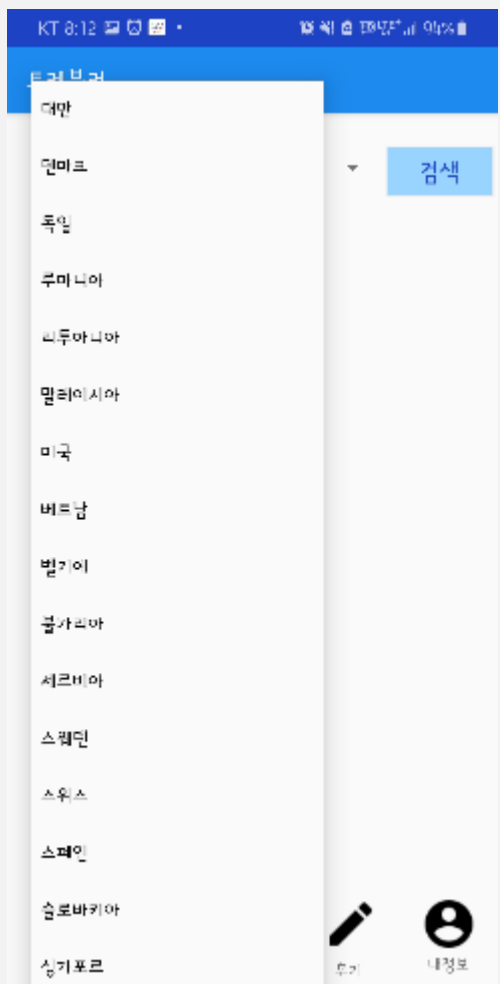

3) 작품기능 및 주요 코드

◦ DB에 저장된 국가 정보 불러오기



3) 작품기능 및 주요 코드

◦ DB에 저장된 국가 정보 불러오기

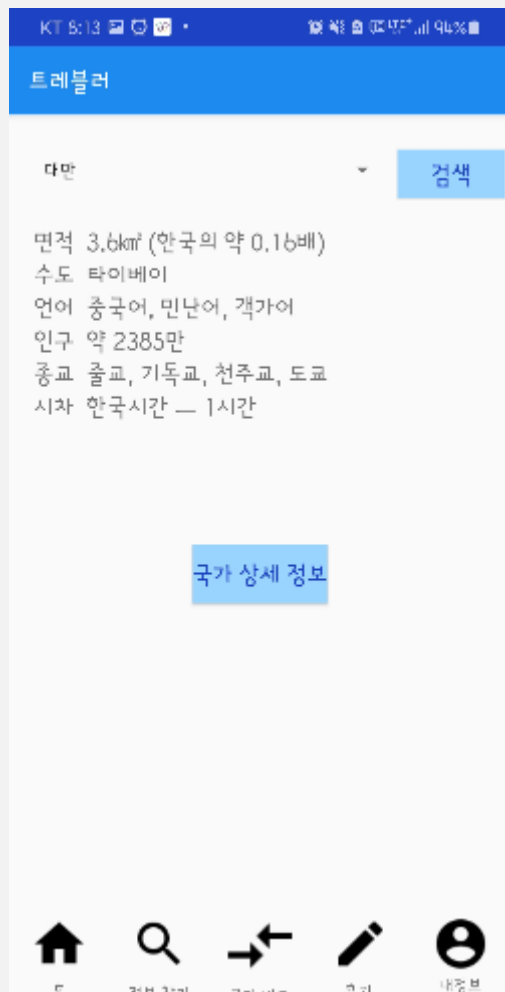


```
ref2.addValueEventListener(new ValueEventListener() {  
    @Override  
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
        for (DataSnapshot snapshot : dataSnapshot.getChildren()) {  
            String name = snapshot.child("name").getValue(String.class);  
  
            if(name.equals(country1)){  
                area1 = snapshot.child("area").getValue(String.class);  
                capital1 = snapshot.child("capital").getValue(String.class);  
                language1 = snapshot.child("language").getValue(String.class);  
                population1 = snapshot.child("population").getValue(String.class);  
                religion1 = snapshot.child("religion").getValue(String.class);  
                time1 = snapshot.child("time").getValue(String.class);  
                traffic1 = snapshot.child("traffic").getValue(String.class);  
                bigmack1 = snapshot.child("bigmack").getValue(String.class);  
            }  
            if(name.equals(country2)){  
                area2 = snapshot.child("area").getValue(String.class);  
                capital2 = snapshot.child("capital").getValue(String.class);  
                language2 = snapshot.child("language").getValue(String.class);  
                population2 = snapshot.child("population").getValue(String.class);  
                religion2 = snapshot.child("religion").getValue(String.class);  
                time2 = snapshot.child("time").getValue(String.class);  
                traffic2 = snapshot.child("traffic").getValue(String.class);  
                bigmack2 = snapshot.child("bigmack").getValue(String.class);  
            }  
        }  
        adapterEx.addItem(area1,area2,capital1,capital2, language1,language2, population1,population2,  
            religion1,religion2, time1,time2, traffic1,traffic2, bigmack1, bigmack2);  
        listView.setAdapter(adapterEx);  
        adapterEx.notifyDataSetChanged();  
    }  
});
```

DB비교 후 도시정보 가져오기

3) 작품기능 및 주요 코드

◦ DB에 저장된 국가 정보 불러오기



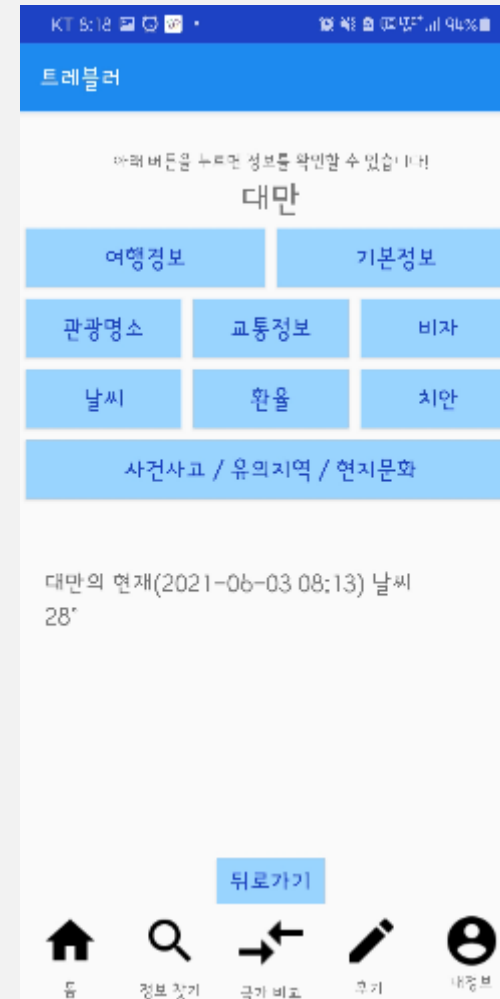
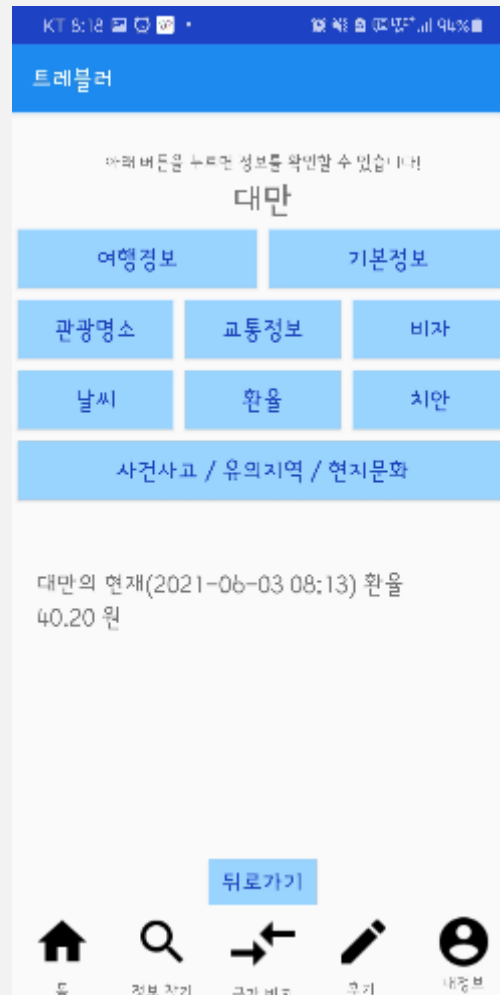
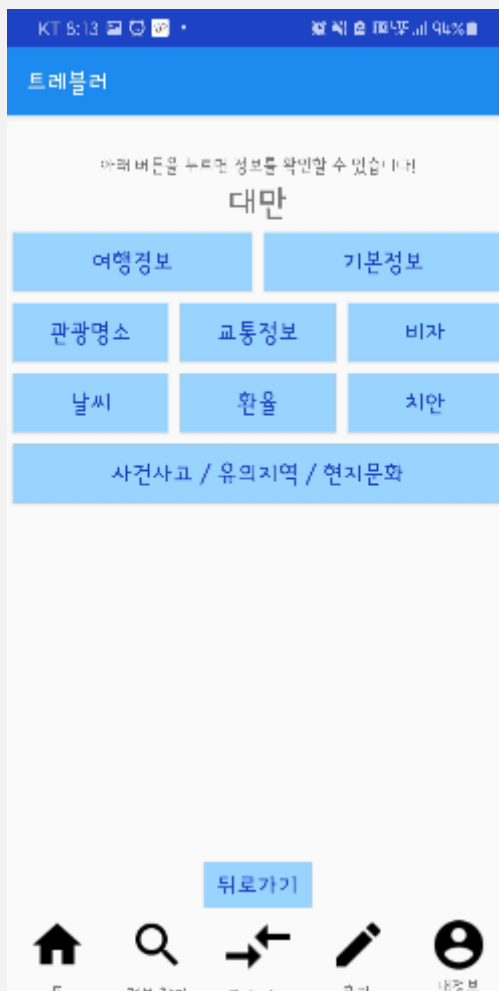
```
super.run())
ref.child("Basic").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
        for (DataSnapshot snapshot : dataSnapshot.getChildren()) {
            String name = snapshot.child("name").getValue(String.class);
            if (country.equals(name)) {
                String area = snapshot.child("area").getValue(String.class);
                String capital = snapshot.child("capital").getValue(String.class);
                String language = snapshot.child("language").getValue(String.class);
                String population = snapshot.child("population").getValue(String.class);
                String religion = snapshot.child("religion").getValue(String.class);
                String time = snapshot.child("time").getValue(String.class);

                travel_level2.setText(area);
                travel_level3.setText(capital);
                travel_level4.setText(language);
                travel_level5.setText(population);
                travel_level6.setText(religion);
                travel_level7.setText(time);
                book = true;
            }
        }
    }
})
```

선택 국가 기본정보 불러오기

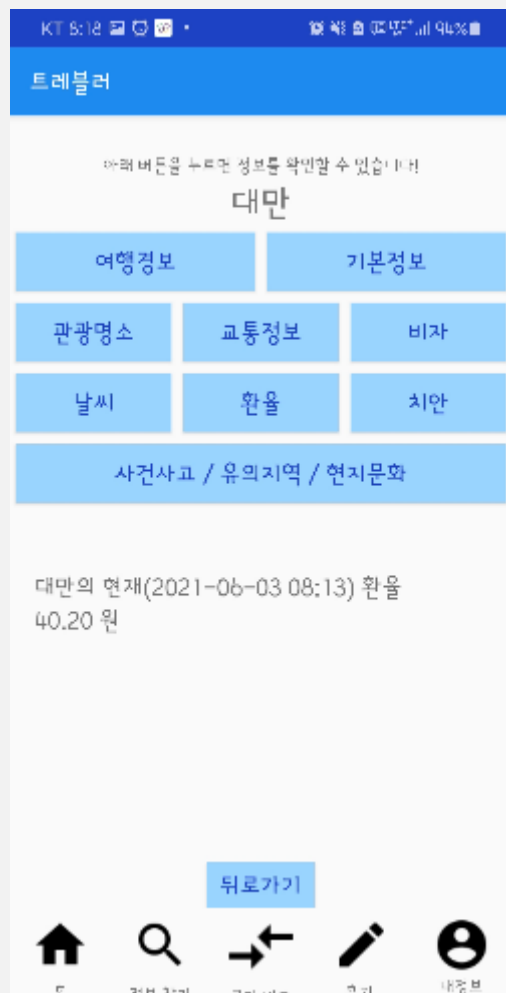
3) 작품기능 및 주요 코드

◦ 실시간 정보 크롤링



3) 작품기능 및 주요 코드

◦ 실시간 정보 크롤링

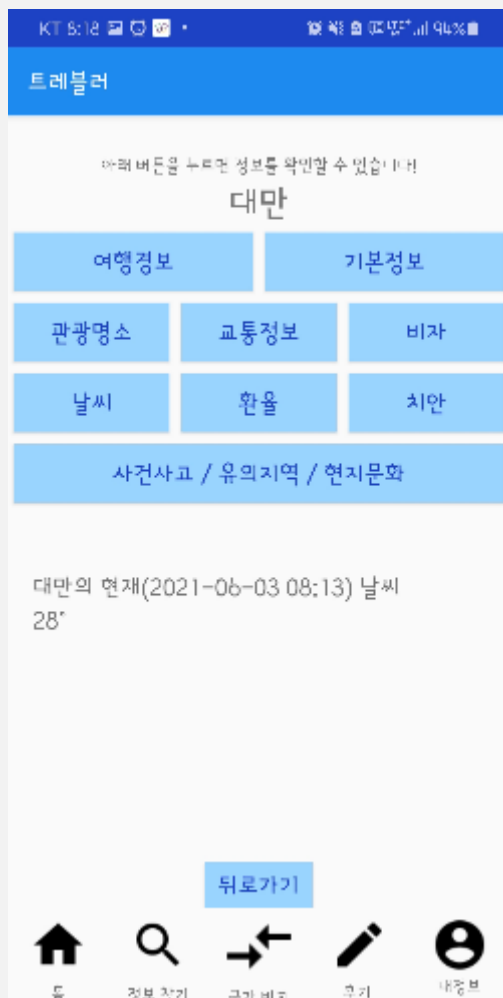


```
new Thread(){
    @Override
    public void run() {
        Document doc = null;
        try {
            doc = Jsoup.connect(gexchangeUri).get();
            Elements contents = doc.select(cssQuery: "span.recite._recite.result");
            nums = contents.text();
            bundle.putString("numbers", nums);
            Message msg = handler.obtainMessage();
            msg.setData(bundle);
            handler.sendMessage(msg);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}.start();
```

실시간 환율 정보 크롤링

3) 작품기능 및 주요 코드

◦ 실시간 정보 크롤링

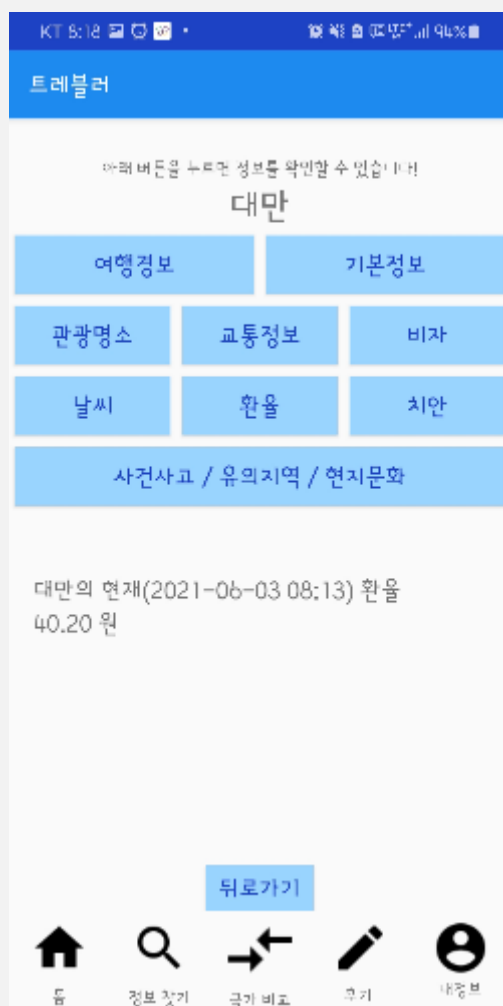
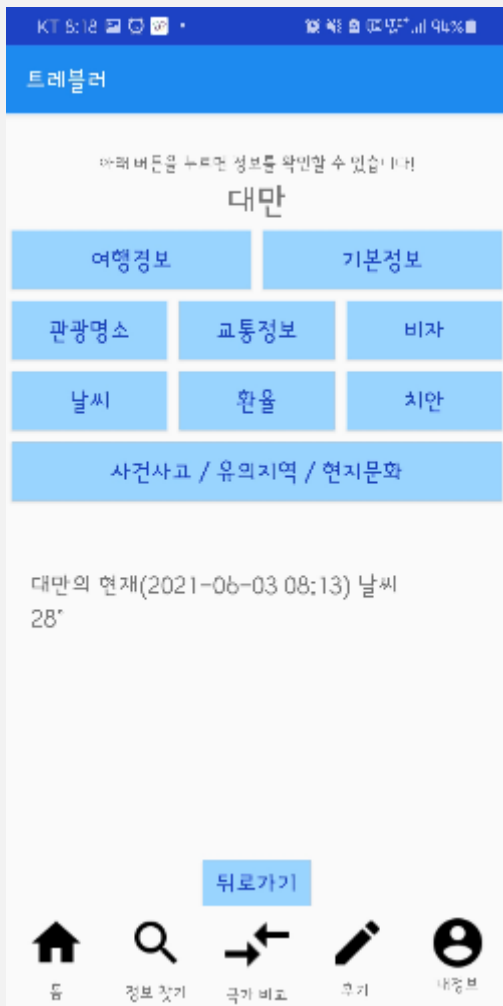


```
new Thread(){
    @Override
    public void run(){
        Document doc = null;
        try {
            doc = Jsoup.connect(gweatherUri).get();
            Elements contents = doc.select("span.CurrentConditions--tempValue--3KcTQ");
            nums = contents.text();
            //핸들러를 이용해서 Thread()에서 가져온 데이터를 메인 쓰레드에 보내준다.
            bundle2.putString("numbers", nums);
            Message msg = handler.obtainMessage();
            msg.setData(bundle2);
            handler.sendMessage(msg);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}.start();
```

실시간 날씨 정보 크롤링

3) 작품기능 및 주요 코드

◦ 실시간 정보 크롤링



```
public class MyHandler extends Handler {  
    private final WeakReference<MoreInformationFragment> weakReference;  
  
    public MyHandler(MoreInformationFragment moreInformationFragment){  
        this.weakReference = new WeakReference<>(moreInformationFragment);  
    }  
  
    @Override  
    public void handleMessage(@NonNull Message msg) {  
        super.handleMessage(msg);  
        Bundle bundle = msg.getData();  
        result1_text.setText(bundle.getString( key: "numbers"));  
    }  
}
```

크롤링에 쓰이는 공용 핸들러

3) 작품기능 및 주요 코드

◦ 국가 정보 비교

KT 5:23 93%

트레블러

비교 화면

비교 국가1 미국

비교 국가2 그리스

비교하기

KT 5:23 93%

트레블러

미국	그리스
면적 983만km ² (한국의 약 45배)	면적 131,957km ² (한국의 0.6배)
수도 워싱턴	수도 아테네
언어 영어	언어 그리스어
이그	이그

확인

KT 5:24 93%

트레블러

미국	그리스
인구 약 3억 3291만	인구 약 1,037만
종교 개신교, 카톨릭, 불교, 유대교	종교 그리스정교(90%), 기독교(3%), 이슬람교(1.3%) 등
시차 한국시간 - 14시간	시차 한국시간 - 7시간
이그	이그

확인

KT 5:24 93%

트레블러

미국	그리스
시차 한국시간 - 14시간	시차 한국시간 - 7시간
교통정보 버스, 택시	교통정보 버스, 택시, 지하철
빅맥지수 5.66 (3위)	빅맥지수 5.16 (6위)
이그	이그

확인

3) 작품기능 및 주요 코드

◦ 후기 작성 및 확인

KT 1:45 81%

트레블러

후기 작성

제목: _____

일정: 여행시작 여행 끝

여행국가: 그리스

별점

★ ★ ★ ★ ★

사진 사진 고르기

위치: _____

내용: _____

저장 취소

KT 5:25 92%

트레블러

전체 국가 후기

이름: changyub1768
제목: 캐나다 다녀왔습니다!

이름: testingrse
제목: 넘 좋아용

이름: dayoung3579
제목: 미국 다녀왔어요!

이름: changyub1768
제목: 와유 그레이트

이름: changyub1768
제목: 와따 멋지네요

이름: dayoung3579
제목: 대만 다녀왔어용

국가별 후기 보기

국가 선택: 그리스 후기보기

홈으로 후기 작성

KT 1:45 81%

트레블러

캐나다

이름: changyub1768
제목: 캐나다 다녀왔습니다!

KT 5:27 92%

트레블러

♡


작성자: changyub1768

제목: 캐나다 다녀왔습니다!

일정: 2021-5-3 ~ 2021-5-7

여행국가: 캐나다

별점: 5.0



3) 작품기능 및 주요 코드

◦ 후기 작성 및 확인

트레블러

후기 작성

제목 : _____

일정 : 여행시작 여행끝




여행국가 : 그리스

별점

★ ★ ★ ★ ★

사진 사진 고르기 사진 업로드

사진을 선택하세요(최대2장)


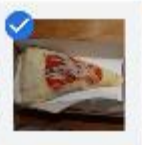


  





카메라 갤러리 모바일




KT 8:31 91%

× 2 완료





5월 9일 (일)





   

5월 8일 (토)

KT 8:32 91%

트레블러

후기 작성

제목 : 여행다녀왔어요~

일정 : 여행시작 2021-5-8 여행끝 2021-5-9

여행국가 : 독일

별점

★ ★ ★ ★ ★

사진 사진 고르기 사진 업로드

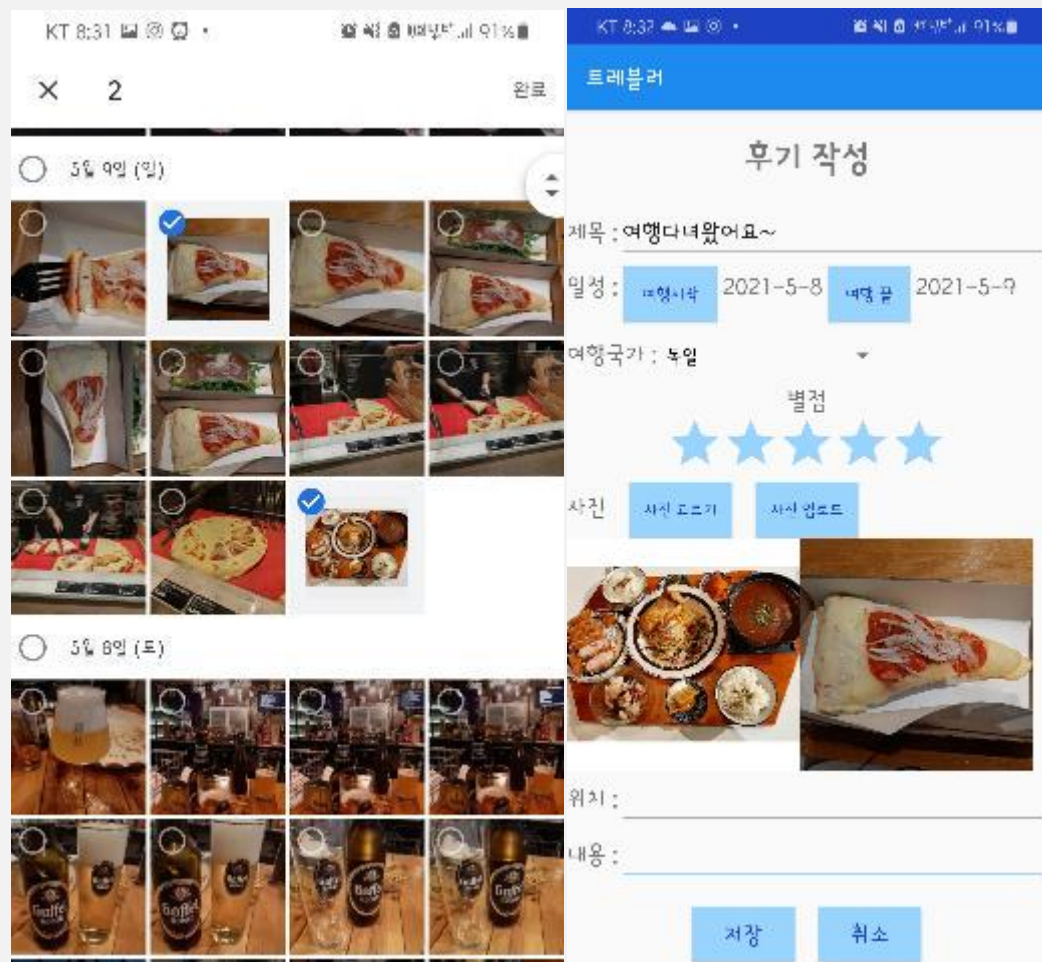
 

위치 : _____

내용 : _____

3) 작품기능 및 주요 코드

◦ 후기 작성 및 확인



```
Intent intent = new Intent(Intent.ACTION_PICK, MediaStore.Audio.Media.EXTERNAL_CONTENT_URI);
intent.putExtra(Intent.EXTRA_ALLOW_MULTIPLE, value: true);
intent.setType("image/*");
startActivityForResult(Intent.createChooser(intent, "사진을 선택하세요(최대2장)", PICTURAL_REQUEST_CODE);
```

갤러리 이미지 다중선택

```
if(requestCode == PICTURAL_REQUEST_CODE && resultCode == Activity.RESULT_OK) {
    photoURI = data.getData();
    ClipData clipData = data.getClipData();

    if (clipData != null) {
        for (int i = 0; i < clipData.getItemCount(); i++) {
            if (i < clipData.getItemCount()) {
                photoURI = clipData.getItemAt(i).getUri();
                switch (i) {
                    case 0:
                        img_view.setImageURI(photoURI);
                        break;
                    case 1:
                        img_view2.setImageURI(photoURI);
                        break;
                }
            }
        }
    } else if (photoURI != null) {
        img_view.setImageURI(photoURI);
    }
}
```

선택 사진
정보 가져오기

3) 작품기능 및 주요 코드

◦ 후기 작성 및 확인



작성자:dayoung35

제목:여행다녀왔어요~

일정:2021-5-8 ~ 2021-5-9

여행국가:독일

별점:5.0



```
ClipData clipData = data.getClipData();
if(clipData !=null){
    for(int i=0; i<clipData.getItemCount();i++){
        if(i<clipData.getItemCount()){
            photoURI = clipData.getItemAt(i).getUri();

            filename = "photo_"+i+".png";

            StorageReference storageReference = storage.getReferenceFromUrl( fullUrl: "gs://example2020-9a3ed.appspot.com/")
            .child("User/"+name).child(key).child(filename);

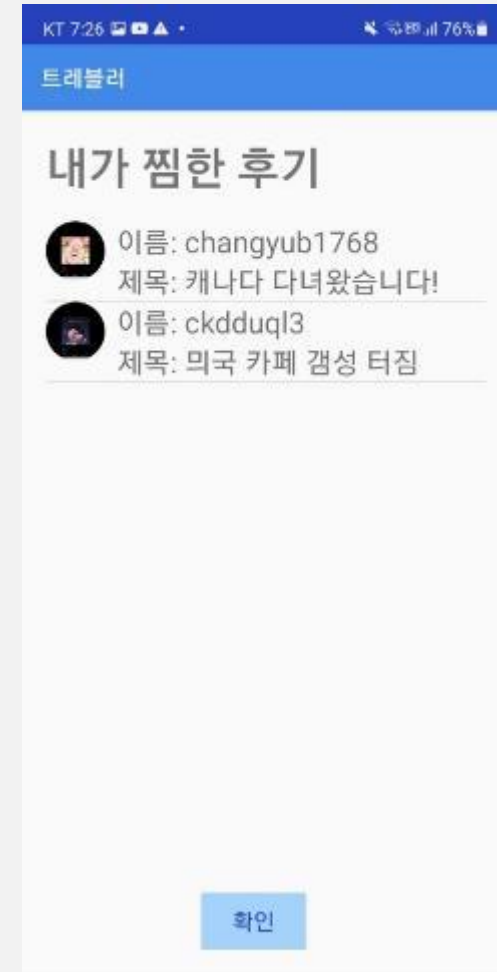
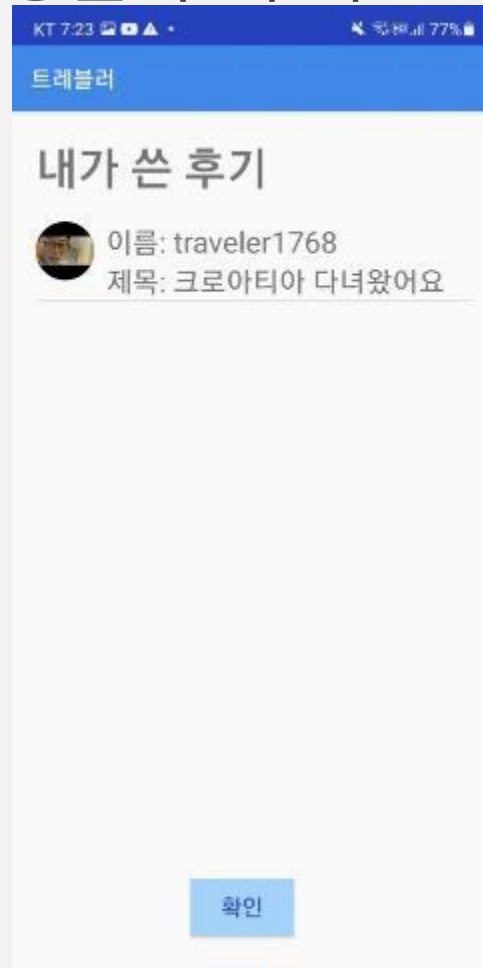
            UploadTask uploadTask = storageReference.putFile(photoURI);

            uploadTask.addOnSuccessListener((OnSuccessListener) (taskSnapshot) → {
                //성공시.
            }).addOnFailureListener((e) → {
                //실패시
            });
        }
    }
}
```

작성 후기사진 스토리지에 저장

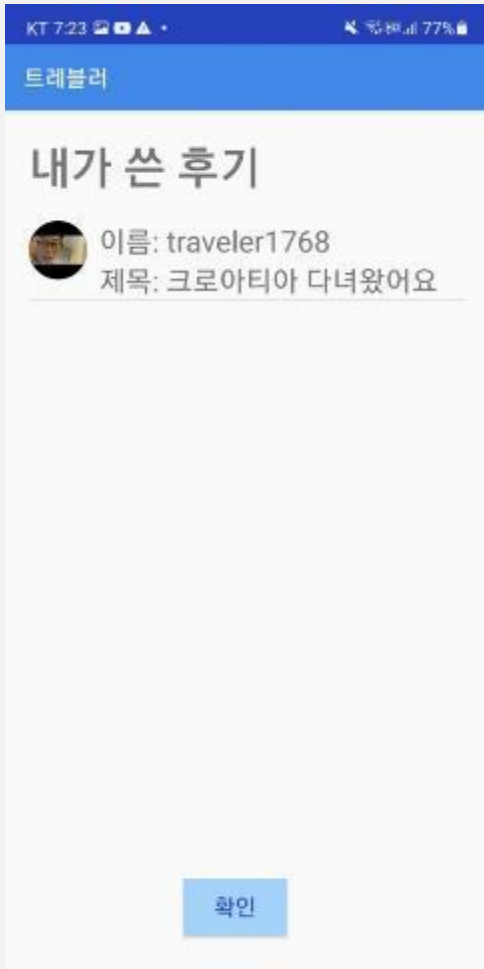
3) 작품기능 및 주요 코드

◦ 내 관심 후기 및 내가 작성한 후기 확인



3) 작품기능 및 주요 코드

◦ 내 관심 후기 및 내가 작성한 후기 확인

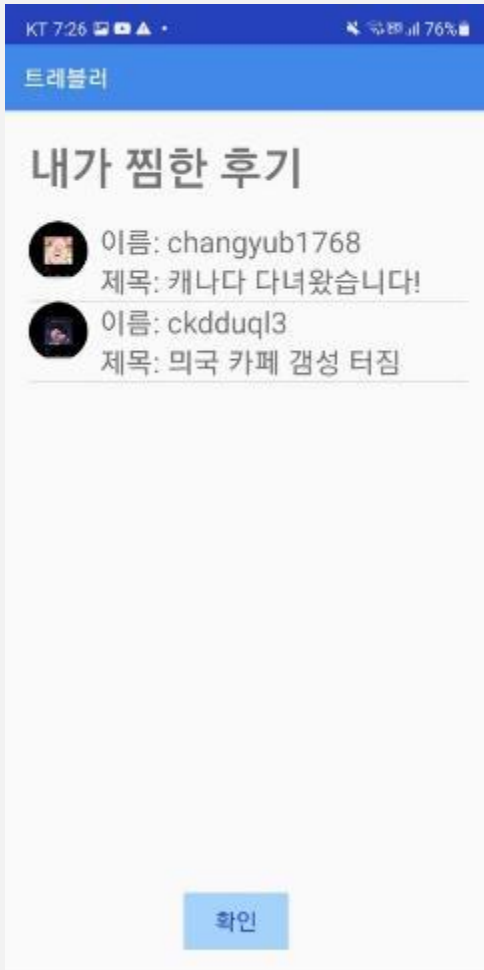


```
ref.addValueEventListener(new ValueEventListener() {  
    @Override  
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
        for(DataSnapshot snapshot : dataSnapshot.getChildren()){  
            username = snapshot.child("name").getValue(String.class);  
            if(name.equals(username)){  
                String title = snapshot.child("title").getValue(String.class);  
                listViewAdapter.addItem(username,title, username);  
                my_list.setAdapter(listViewAdapter);  
                String date = snapshot.child("date").getValue(String.class);  
                String location = snapshot.child("location").getValue(String.class);  
                String text = snapshot.child("text").getValue(String.class);  
                String star_value = snapshot.child("star_value").getValue(String.class);  
                String country = snapshot.child("country").getValue(String.class);  
                String mykey = snapshot.getKey();  
  
                keylist.add(mykey);  
                nlist.add(name);  
                titlelist.add(title);  
                datelist.add(date);  
                clist.add(country);  
                llist.add(location);  
                textlist.add(text);  
                starlist.add(star_value);  
            }  
        }  
        listViewAdapter.notifyDataSetChanged();  
    }  
});
```

내가 쓴 후기 (DB에서 유저와 현재로그인한 유저 비교 후 가져옴)

3) 작품기능 및 주요 코드

◦ 내 관심 후기 및 내가 작성한 후기 확인



```
for(DataSnapshot snapshot : dataSnapshot.getChildren()){
    String choicename = snapshot.child("point").child(name).getKey();
    String point = snapshot.child("point").child(name).getValue(String.class);
    if(choicePoint.equals(point) && name.equals(choicename)){
        String username = snapshot.child("name").getValue(String.class);
        String title = snapshot.child("title").getValue(String.class);
        listViewAdapter.addItem(username,title, username);
        my_list.setAdapter(listViewAdapter);

        String date = snapshot.child("date").getValue(String.class);
        String location = snapshot.child("location").getValue(String.class);
        String text = snapshot.child("text").getValue(String.class);
        String star_value = snapshot.child("star_value").getValue(String.class);
        String country = snapshot.child("country").getValue(String.class);
        String mykey = snapshot.getKey();
        String check = snapshot.child("point").child(name).getValue(String.class);

        checklist.add(check);
        keylist.add(mykey);
        nlist.add(username);
        titlelist.add(title);
        datelist.add(date);
        clist.add(country);
        llist.add(location);
        textlist.add(text);
        starlist.add(star_value);
    }
}
listViewAdapter.notifyDataSetChanged();
```

내가 찜한 후기(이름이 같고 좋아요를 눌렀는지 확인)

4. 향후 계획 및 기대효과

4) 향후 계획 및 기대효과

향후 계획

- 제공하고 있는 국가 수를 늘려 최종적으로 전 세계 모든 국가 정보를 제공할 계획
- 정확한 실시간 정보를 얻을 수 있게 주기적인 업데이트

기대효과

- 제약 없는 정보비교와 검색이 편한 올인원 시스템이 될 것
- 소수 여행이 아니더라도 해외여행에 대한 정보를 쉽게 제공받을 것
- 추천국가를 제공받아 만족도 높은 여행이 될 수 있을 것

감사합니다”