Find My Pet Information
FMPI

이승재 임채선 민상규

목차

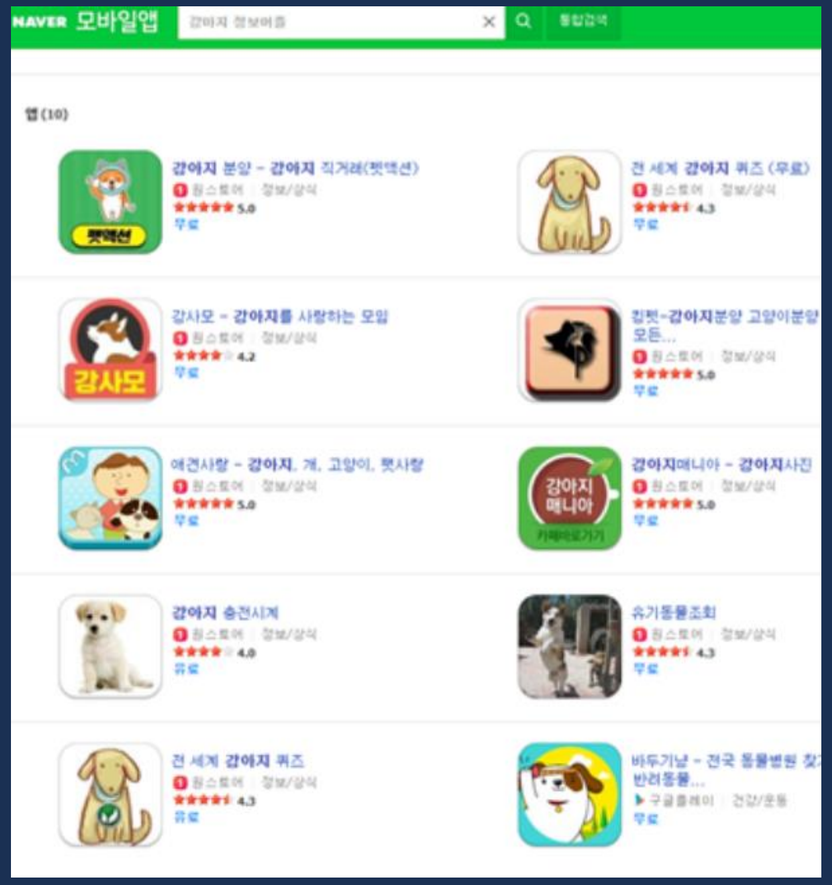
FMPI 개발 배경


나 혼자 산다
다른 애완동물보다
혼자 있는 걸 좋아하는 고슴도치

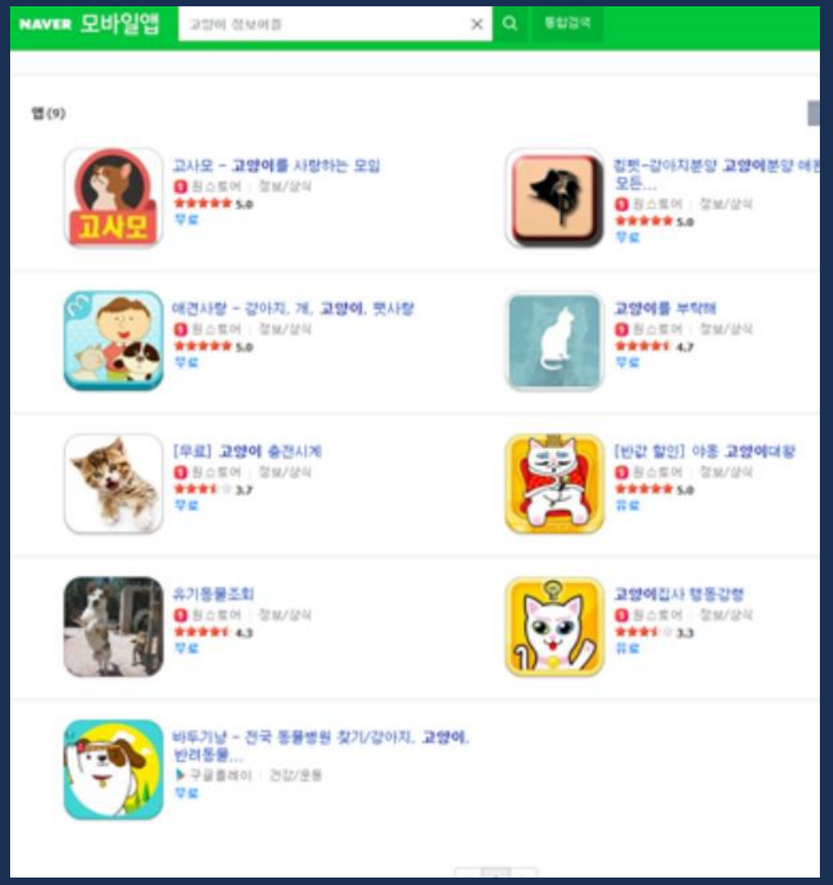

미운 우리새끼
아찔한 동거
SBS
조심
조심
조심스럽게 고치와 첫인사를 나누는 짠용

FMPI 개발 필요성

현재 개발된 강아지,고양이 어플



- 강아지 정보어플 (10개)



- 고양이 정보어플 (9개)

FMPI 개발 필요성

현재 개발된 이색(고슴도치, 기니피그 등..) 애완동물 어플

- 고슴도치 정보어플 (0개)
- 햄스터 정보어플 (0개)
- 기니피그 정보어플 (0개)

문제인식

종(Species)

내 애완동물은 어떤 품종일까?

내 애완동물에게 필요한 정보는 무엇일까?

정보 부족

개발환경(SW/HW)

SW개발환경

apache HTTP Server 2.4, , Synology WebStation, Sy
snology WebDav Server, HTML, Android studio 4.0.1

HW개발환경

안드로이드 휴대폰, 개인 pc

Built with Firebase

FMPI 개발 역할

201511044 컴퓨터과학과

- 데이터 수집
-프로젝트 문서관리
-HTML과 CSS를 통한 웹페이지 개발

201511053 컴퓨터과학과

-Android studio 를 이용한 앱개발
-HTML과 CSS를 통한 웹페이지 개발

201511019 컴퓨터과학과

-Android studio 를 이용한 앱개발
-FireBase(DB) 관리
-UI 디자인

FMPI 개발– 데이터셋

시나몬고슴도치

416 Image Samples

Webcam Upload

아메리칸기니피그

962 Image Samples

Webcam Upload

정글리안햄스터

959 Image Samples

Webcam Upload

플라티나고슴도치

550 Image Samples

Webcam Upload

실키기니피그

993 Image Samples

Webcam Upload

펄햄스터

984 Image Samples

Webcam Upload

알비노고슴도치

1257 Image Samples

Webcam Upload

스키니피그

1000 Image Samples

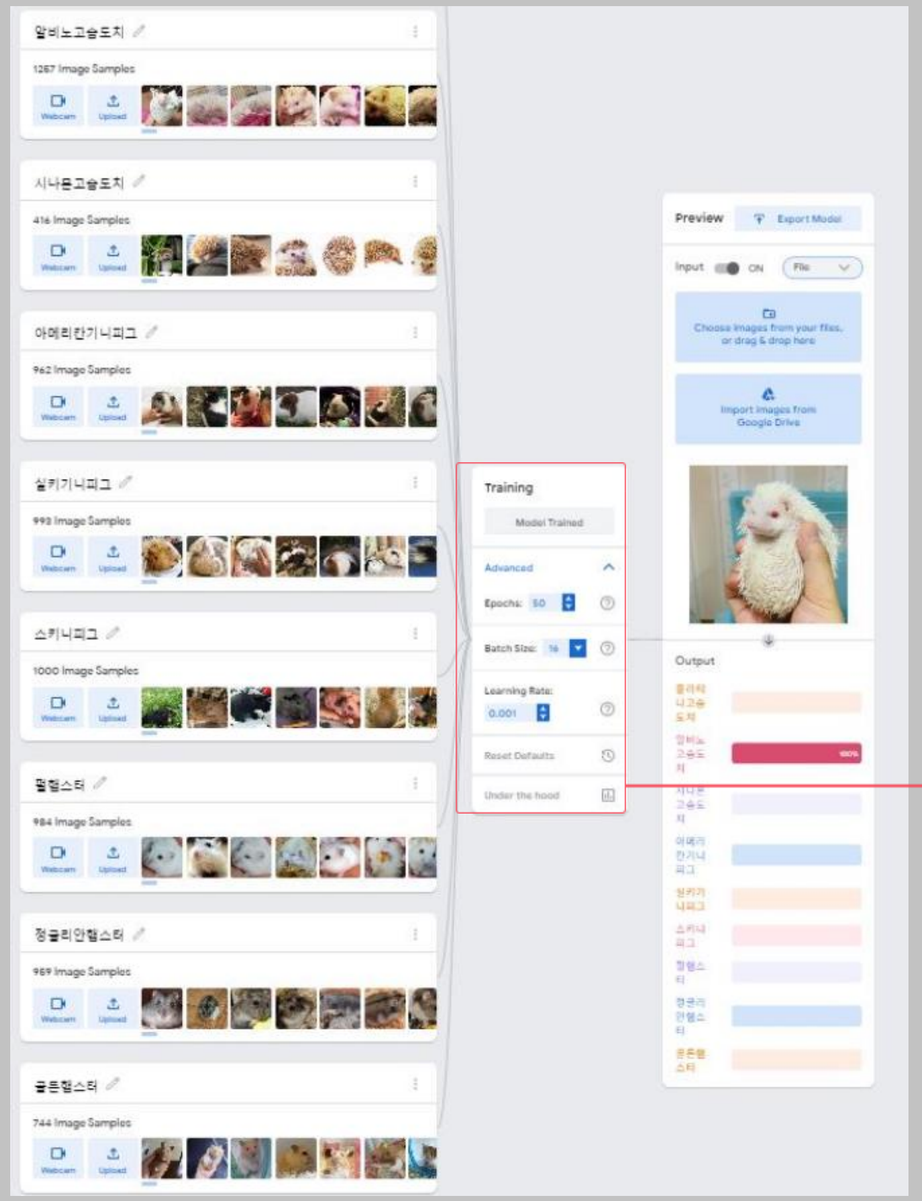
Webcam Upload

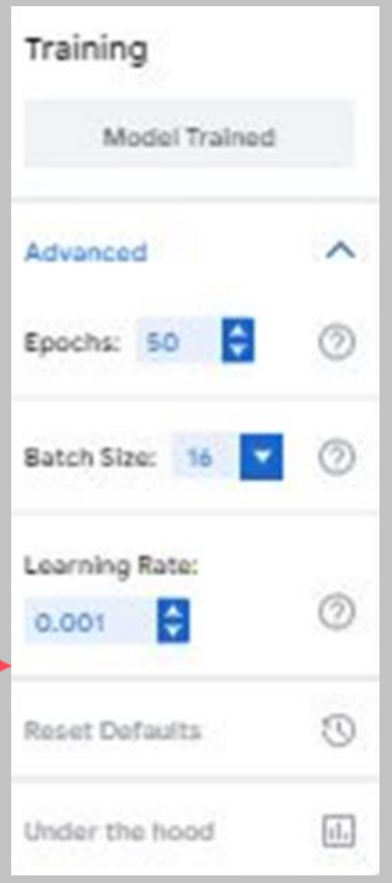
골든햄스터

744 Image Samples

Webcam Upload

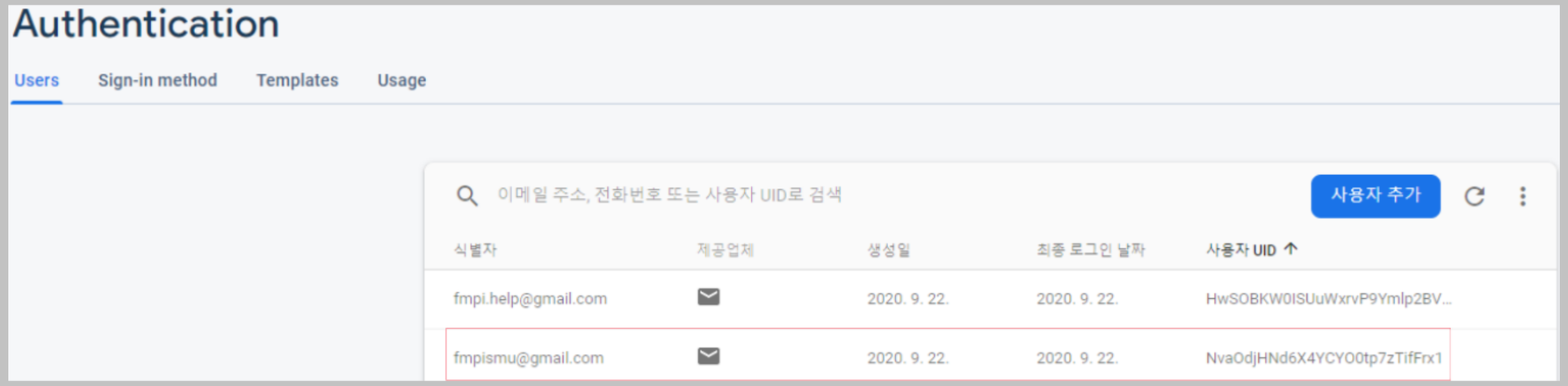
FMPI 개발 – 데이터 학습





FMPI 개발 – 회원가입

<FMPI APP 화면>

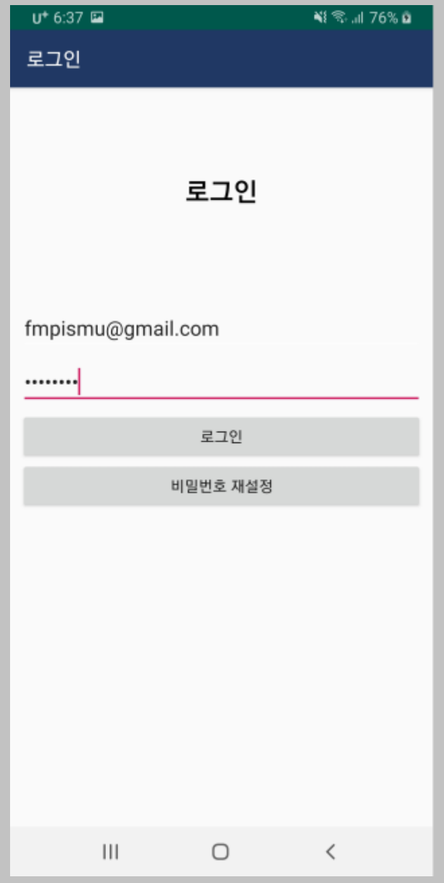


<Firestore에 회원가입의 정보가 들어간 화면>

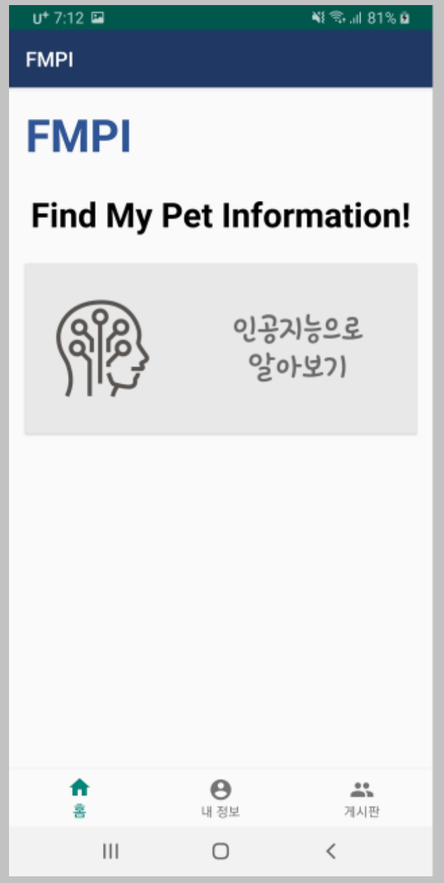
```
private void signUp() {
    String email = ((EditText)findViewById(R.id.emailEditText)).getText().toString();
    String password = ((EditText)findViewById(R.id.passwordEditText)).getText().toString();
    String passwordCheck = ((EditText)findViewById(R.id.passwordCheckEditText)).getText().toString();

    if(email.length() > 0 && password.length() > 0 && passwordCheck.length() > 0){
        if(password.equals(passwordCheck)){
            mAuth.createUserWithEmailAndPassword(email, password)
                    .addOnCompleteListener( activity: this, (task) -> {
                            if (task.isSuccessful()) {
                                FirebaseUser user = mAuth.getCurrentUser();
                                showToast( activity: SignUpActivity.this, msg: "회원가입에 성공하였습니다.");
                                myStartActivity(MainActivity.class);
                            } else {
                                if(task.getException() != null){
                                    showToast( activity: SignUpActivity.this, task.getException().toString());
                                }
                            }
                    });
        }else{
            showToast( activity: SignUpActivity.this, msg: "비밀번호가 일치하지 않습니다.");
        }
    }else {
        showToast( activity: SignUpActivity.this, msg: "이메일 또는 비밀번호를 입력해 주세요.");
    }
}
```

<JAVA 파일 코드>

FMPI 개발 – 로그인



<FMPI APP 화면>

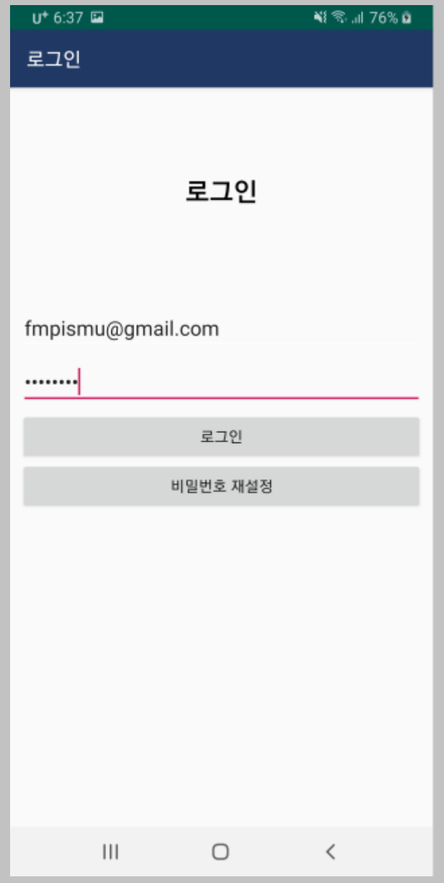


<FMPI APP 화면>

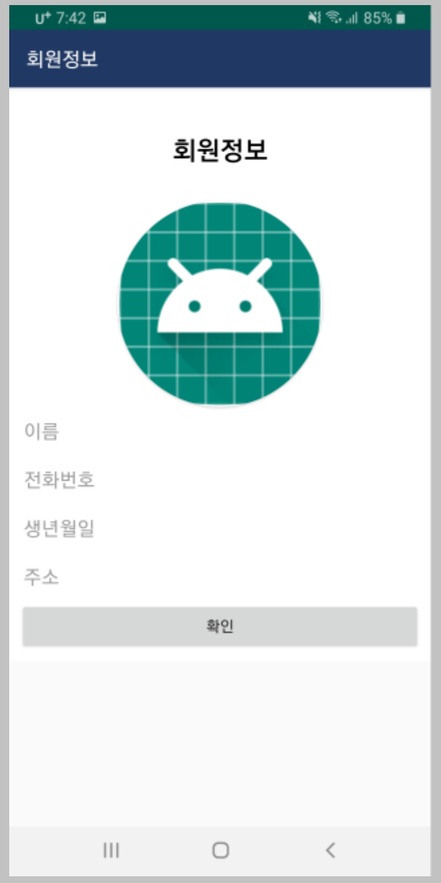
```java
private void login() {
    String email = ((EditText) findViewById(R.id.emailEditText)).getText().toString();
    String password = ((EditText) findViewById(R.id.passwordEditText)).getText().toString();

    if (email.length() > 0 && password.length() > 0) {
        mAuth.signInWithEmailAndPassword(email, password)
                .addOnCompleteListener( activity: this, (task) -> {
                        if (task.isSuccessful()) {
                            FirebaseUser user = mAuth.getCurrentUser();
                            showToast( activity: LoginActivity.this,  msg: "로그인에 성공하였습니다.");
                            myStartActivity(MainActivity.class);
                        } else {
                            if (task.getException() != null) {
                                showToast( activity: LoginActivity.this, task.getException().toString());
                            }
                        }
                });
    } else {
        showToast( activity: LoginActivity.this,  msg: "이메일 또는 비밀번호를 입력해 주세요.");
    }
}
```

<JAVA 파일 코드>

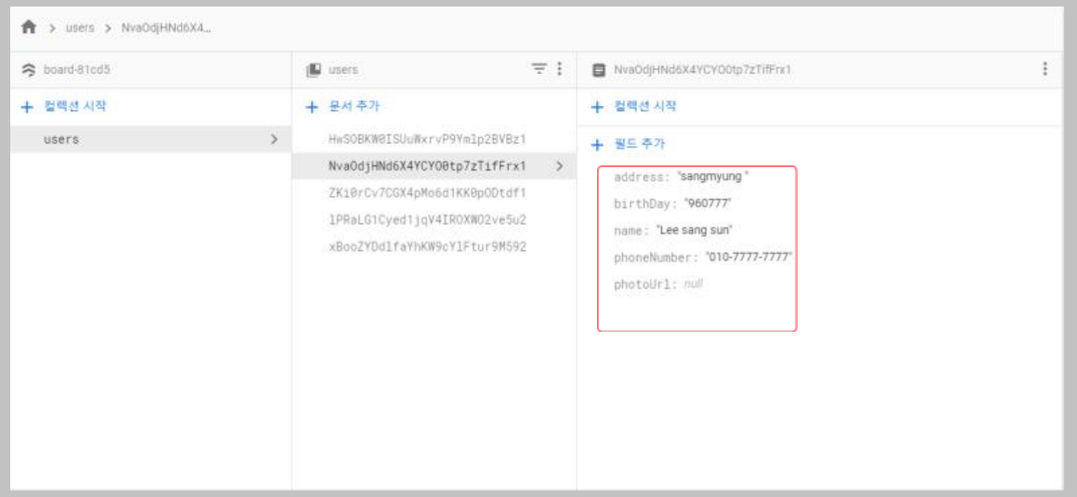
FMPI 개발 – 회원정보



<FMPI APP 화면>



<FMPI APP 화면>

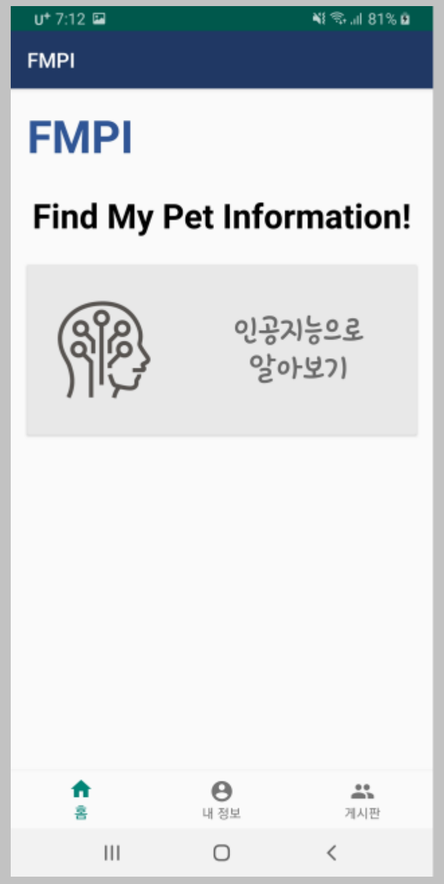


<Firestore에 회원의 정보가 들어간 화면>

```
private void storageUploader() {
    final String name = ((EditText) findViewById(R.id.nameEditText)).getText().toString();
    final String phoneNumber = ((EditText) findViewById(R.id.phoneNumberEditText)).getText().toString();
    final String birthDay = ((EditText) findViewById(R.id.birthDayEditText)).getText().toString();
    final String address = ((EditText) findViewById(R.id.addressEditText)).getText().toString();


    UserInfo userInfo = new UserInfo(name, phoneNumber, birthDay, address, downloadUri.toString());
    storeUploader(userInfo);
private void storeUploader(UserInfo userInfo) {
    FirebaseFirestore db = FirebaseFirestore.getInstance();
    db.collection( collectionPath: "users").document(user.getUid()).set(userInfo)
            .addOnSuccessListener((OnSuccessListener) (aVoid) -> {
                    showToast( activity: MemberInitActivity.this,  msg: "회원정보 등록을 성공하였습니다.");
                    loaderLayout.setVisibility(View.GONE);
                    finish();
            })
            .addOnFailureListener((e) -> {
                    showToast( activity: MemberInitActivity.this,  msg: "회원정보 등록에 실패하였습니다.");
                    loaderLayout.setVisibility(View.GONE);
                    Log.w(TAG,  msg: "Error writing document", e);
            });
}
```

<JAVA 파일 코드>

FMPI 개발 – 사진업로드

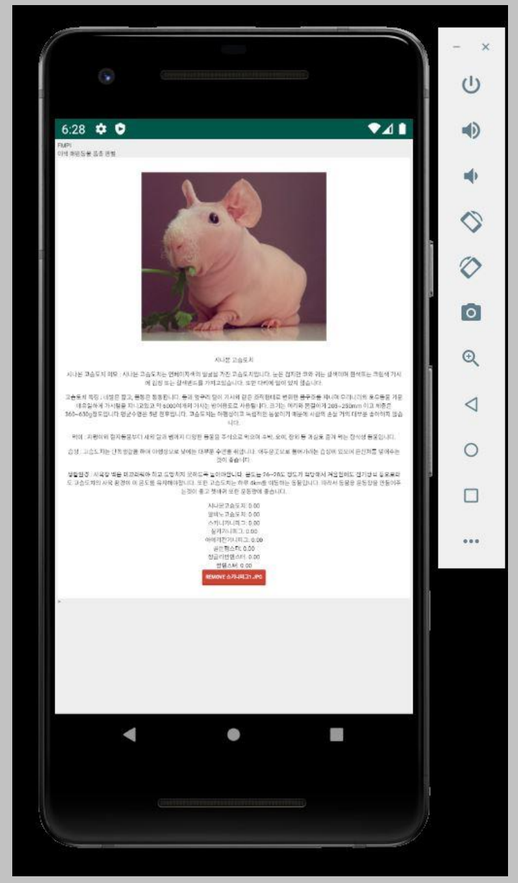


<FMPI APP 화면>

인공지능으로
알아보기 click

<FMPI APP 화면>

Click Here 누르고
갤러리에서 사진선택



<FMPI APP 화면>

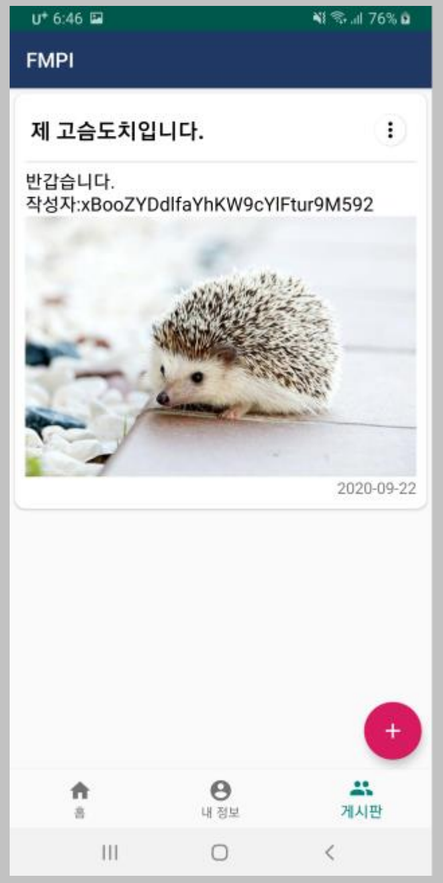
```
public void onCloseWindow(WebView w) { //웹뷰어 실행시켰을때랑 꺼졌을때
    super.onCloseWindow(w);
    finish();//액티비티종료
}

@Override
public boolean onCreateWindow(WebView view, boolean dialog, boolean userGesture, Message resultMsg) {
    final WebSettings settings = view.getSettings();
    settings.setDomStorageEnabled(true);
    settings.setJavaScriptEnabled(true); // 자바스크립트 사용가능하게 해주는것.
    settings.setAllowFileAccess(true);
    settings.setAllowContentAccess(true);
    view.setWebChromeClient(this);
    WebView.WebViewTransport transport = (WebView.WebViewTransport) resultMsg.obj;
    transport.setWebView(view);
    resultMsg.sendToTarget();
    return false;
}
    private void imageChooser() { //카메라랑 갤러리중에 선택
        Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
            // Create the File where the photo should go
            File photoFile = null;
            try {
                photoFile = createImageFile();
                takePictureIntent.putExtra( name: "PhotoPath", mCameraPhotoPath);
            } catch (IOException ex) {
                // Error occurred while creating the File
                Log.e(getClass().getName(),  msg: "Unable to create Image File", ex);
            }

            // Continue only if the File was successfully created
            if (photoFile != null) {
                mCameraPhotoPath = "file:"+photoFile.getAbsolutePath();
                takePictureIntent.putExtra(MediaStore.EXTRA_OUTPUT,
                        Uri.fromFile(photoFile));
            } else {
                takePictureIntent = null;
            }
        }

        Intent contentSelectionIntent = new Intent(Intent.ACTION_GET_CONTENT);
        contentSelectionIntent.addCategory(Intent.CATEGORY_OPENABLE);
        contentSelectionIntent.setType(TYPE_IMAGE); // 이미지불러오기

        Intent[] intentArray;
        if(takePictureIntent != null) {
            intentArray = new Intent[]{takePictureIntent};
        } else {
            intentArray = new Intent[0];
        }

        Intent chooserIntent = new Intent(Intent.ACTION_CHOOSER);// 카메라 갤러리 둘다 선택
        chooserIntent.putExtra(Intent.EXTRA_INTENT, contentSelectionIntent);
        chooserIntent.putExtra(Intent.EXTRA_TITLE,  value: "Image Chooser");
        chooserIntent.putExtra(Intent.EXTRA_INITIAL_INTENTS, intentArray);

        startActivityForResult(chooserIntent, INPUT_FILE_REQUEST_CODE);
    }
});
```

```
public boolean onShowFileChooser(WebView webView,
                                 ValueCallback<Uri[]> filePathCallback, FileChooserParams fileChooserParams) {
    System.out.println("WebViewActivity A>5, OS Version : " + Build.VERSION.SDK_INT + "\t onSFC(WV,VCUB,FCP), n=3");
    if (mFilePathCallback != null) {
        mFilePathCallback.onReceiveValue( t: null);
    }
    mFilePathCallback = filePathCallback;
    imageChooser();
    return true;
}
```

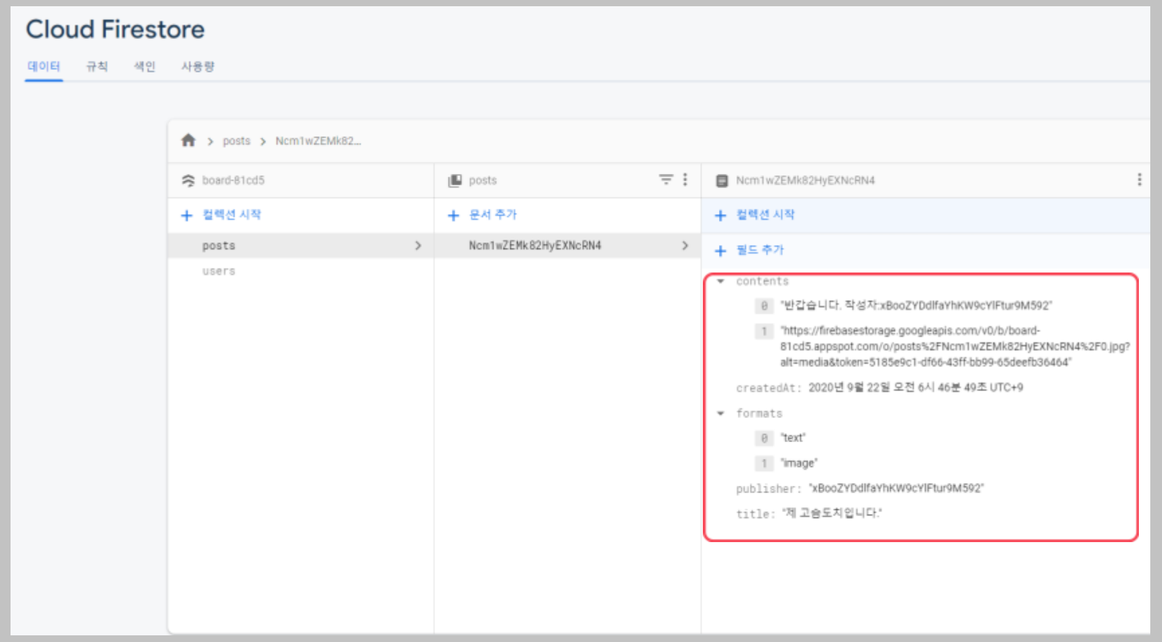
```
private Uri getResultUri(Intent data) { // 안드로이드에 존재하는 파일의 실제경로를 가지고오는것
    Uri result = null;
    if(data == null || TextUtils.isEmpty(data.getDataString())) {
        // If there is not data, then we may have taken a photo
        if(mCameraPhotoPath != null) {
            result = Uri.parse(mCameraPhotoPath);
        }
    } else {
        String filePath = "";
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            filePath = data.getDataString();
        } else {
            filePath = "file:" + RealPathUtil.getRealPath( context: this, data.getData());
        }
        result = Uri.parse(filePath);
    }

    return result;
}
```

FMPI 개발 – 게시글 등록

<FMPI APP 화면>



<FMPI APP 화면>



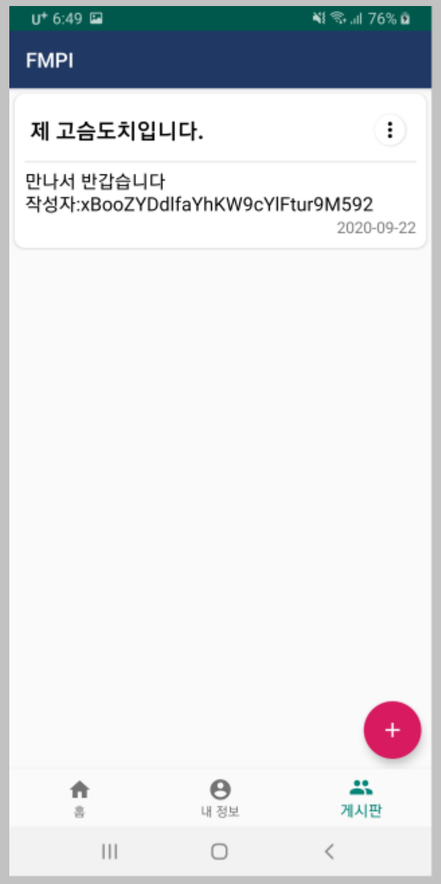
<Firestore에 게시글의 정보가 들어간 화면>

```java
private void storeUpload(DocumentReference documentReference, final PostInfo postInfo) {
    documentReference.set(postInfo.getPostInfo())
            .addOnSuccessListener((OnSuccessListener) (aVoid) -> {
                    Log.d(TAG, msg: "DocumentSnapshot successfully written!");
                    loaderLayout.setVisibility(View.GONE);
                    Intent resultIntent = new Intent();
                    resultIntent.putExtra( name: "postinfo", postInfo);
                    setResult(Activity.RESULT_OK, resultIntent);
                    finish();
            })
            .addOnFailureListener((e) -> {
                    Log.w(TAG, msg: "Error writing document", e);
                    loaderLayout.setVisibility(View.GONE);
            });
}
```

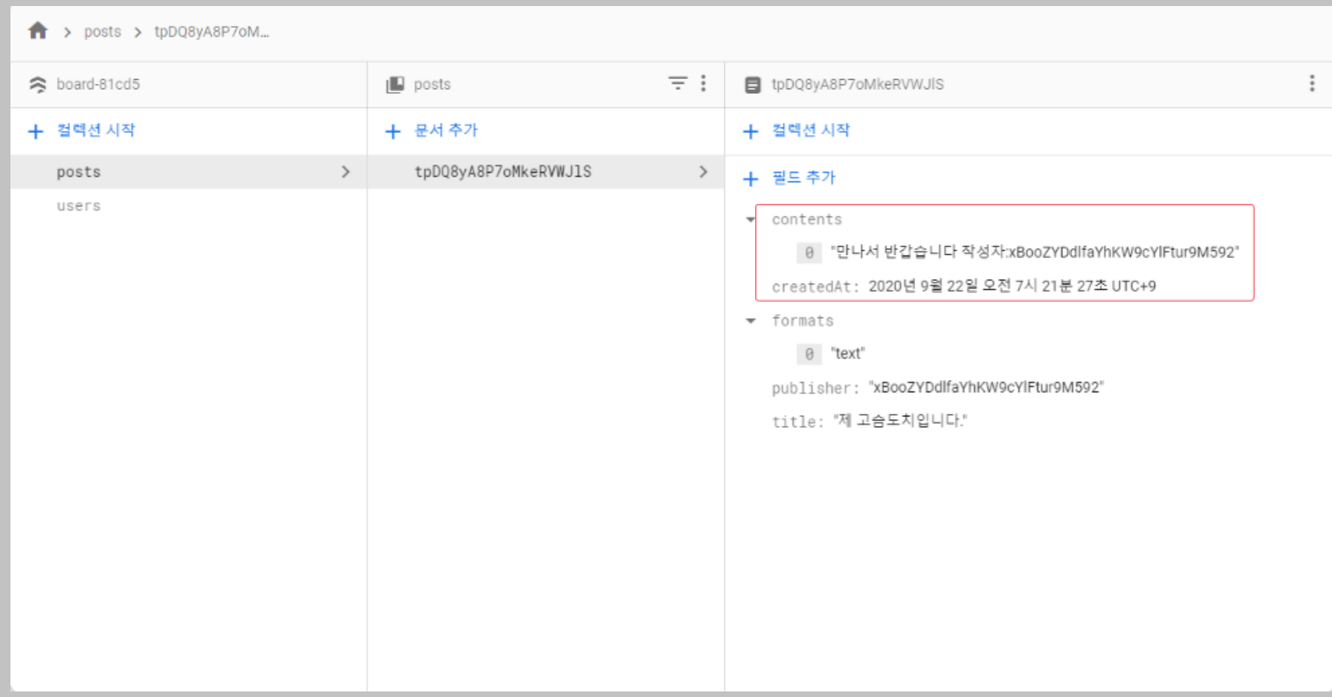
<JAVA 파일 코드>

FMPI 개발 – 게시글 수정

<FMPI APP 화면>

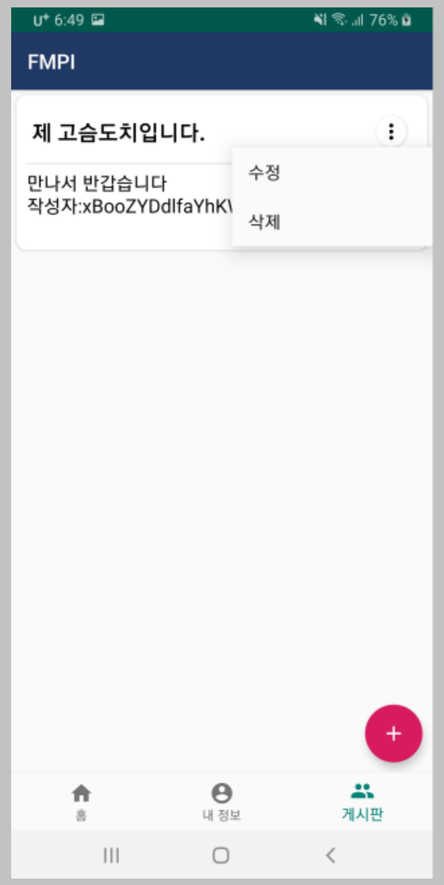


<FMPI APP 화면>

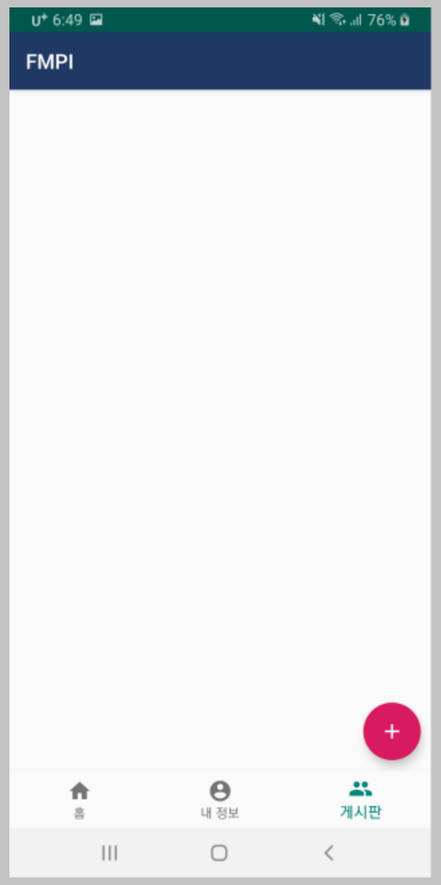


<Firestore에 게시글의 수정 정보가 들어간 화면>

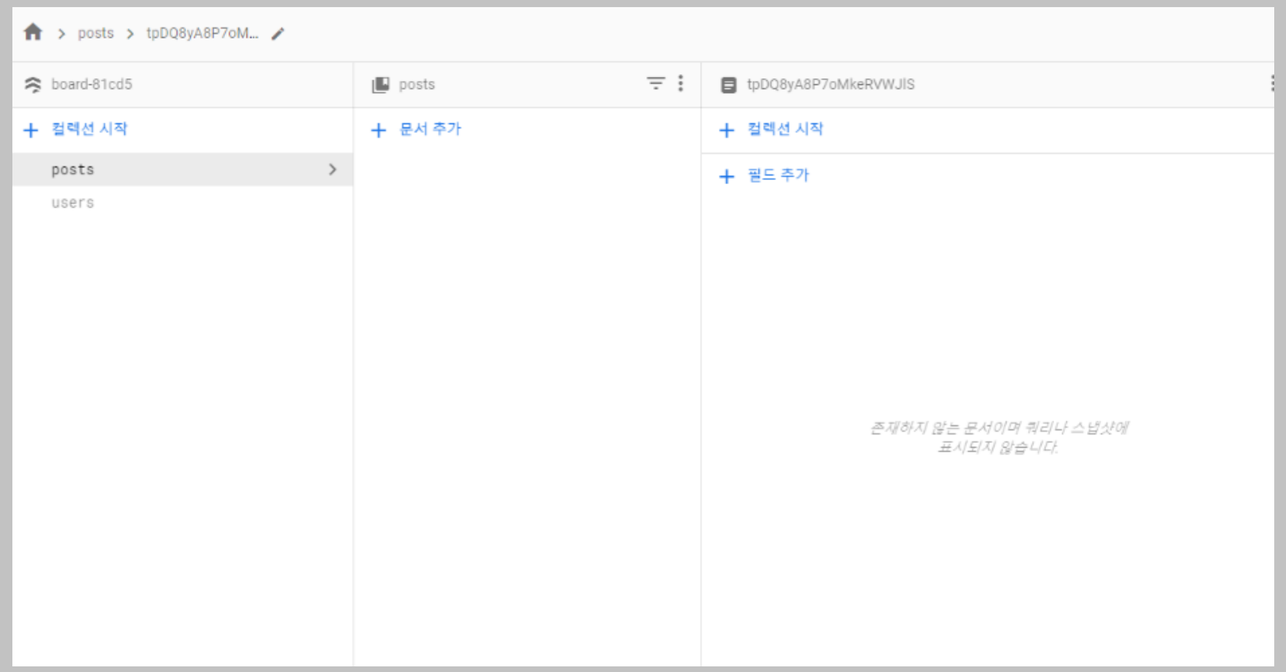
FMPI 개발 – 게시글 삭제



<FMPI APP 화면>



<FMPI APP 화면>

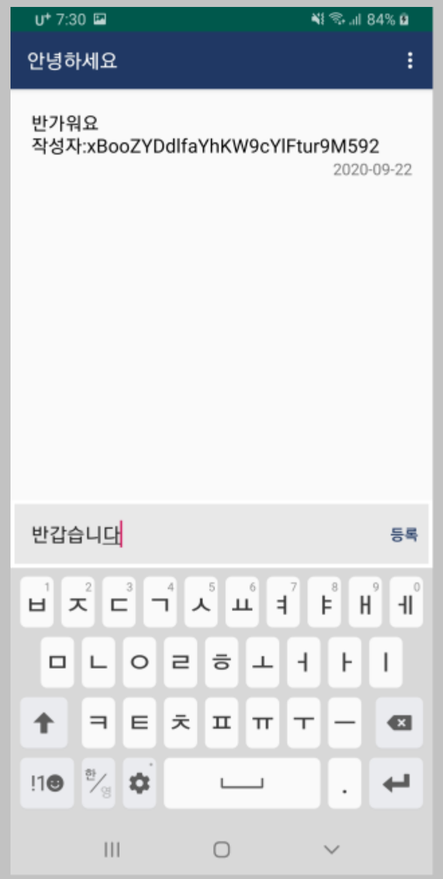


<Firestore에 게시글의 정보가 없어진 화면>

```java
private void storeDelete(final String id, final PostInfo postInfo) {
    FirebaseFirestore firebaseFirestore = FirebaseFirestore.getInstance();
        if (successCount == 0) {
        firebaseFirestore.collection( collectionPath: "posts").document(id) DocumentReference
                .delete() Task<Void>
                .addOnSuccessListener((OnSuccessListener) (aVoid) → {
                        showToast(activity,  msg: "게시글을 삭제하였습니다.");
                        onPostListener.onDelete(postInfo);
                        //postsUpdate();
                })
                .addOnFailureListener((e) → {
                        showToast(activity,  msg: "게시글을 삭제하지 못하였습니다.");
                });
    }
}
```

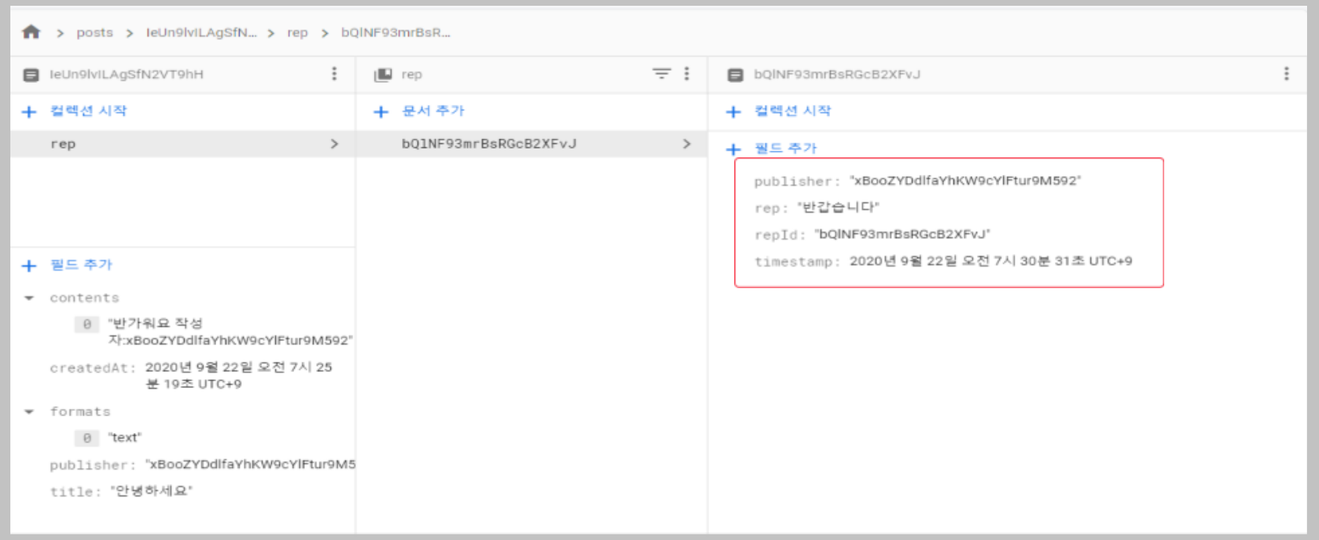
<JAVA 파일 코드>

FMPI 개발 – 댓글등록



<FMPI APP 화면>

<FMPI APP 화면>



<Firestore에 댓글의 정보가 들어간 화면>

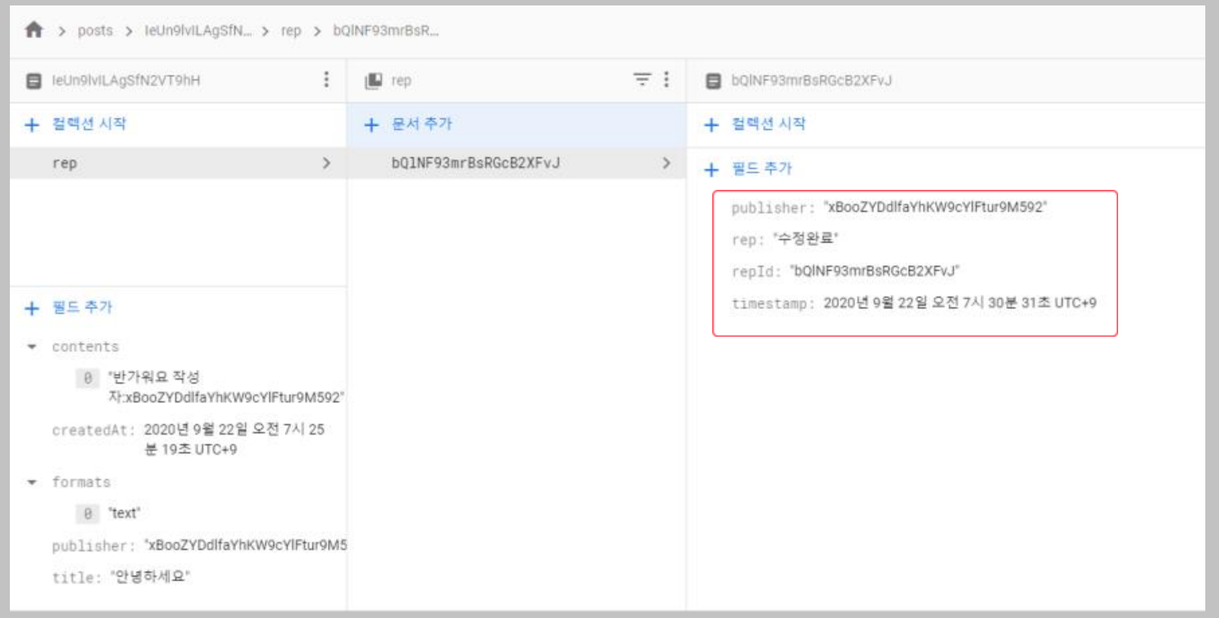
```java
private void addreply(){
    repId = firebaseFirestore.collection( collectionPath: "posts").document(id).collection( collectionPath: "rep").document().getId();
    Map<String, Object> reply = new HashMap<>();
    reply.put( k: "repId",repId);
    reply.put( k: "rep", addreply.getText().toString());
    reply.put( k: "publisher", user.getUid());
    reply.put( k: "timestamp", FieldValue.serverTimestamp());
    firebaseFirestore.collection( collectionPath: "posts").document(id).collection( collectionPath: "rep").document(repId).set(reply);
    finish();

}
```

<JAVA 파일 코드>

FMPI 개발 – 댓글수정

<FMPI APP 화면>

<FMPI APP 화면>



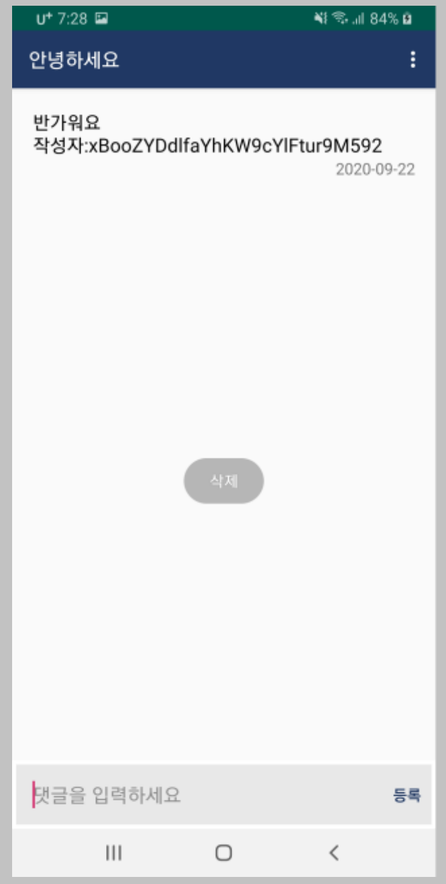
<Firestore에 댓글의 수정이 추가된 화면>

```
public void btnClick(View v){
    firebaseFirestore.collection( collectionPath: "posts") CollectionReference
            .document(id) DocumentReference
            .collection( collectionPath: "rep") CollectionReference
            .document(mRepList.get(position).getRepId()) DocumentReference
            .update( field: "rep",addreply.getText().toString());
    Toast.makeText(getApplicationContext(),  text: "수정",
            Toast.LENGTH_SHORT).show();

}
```

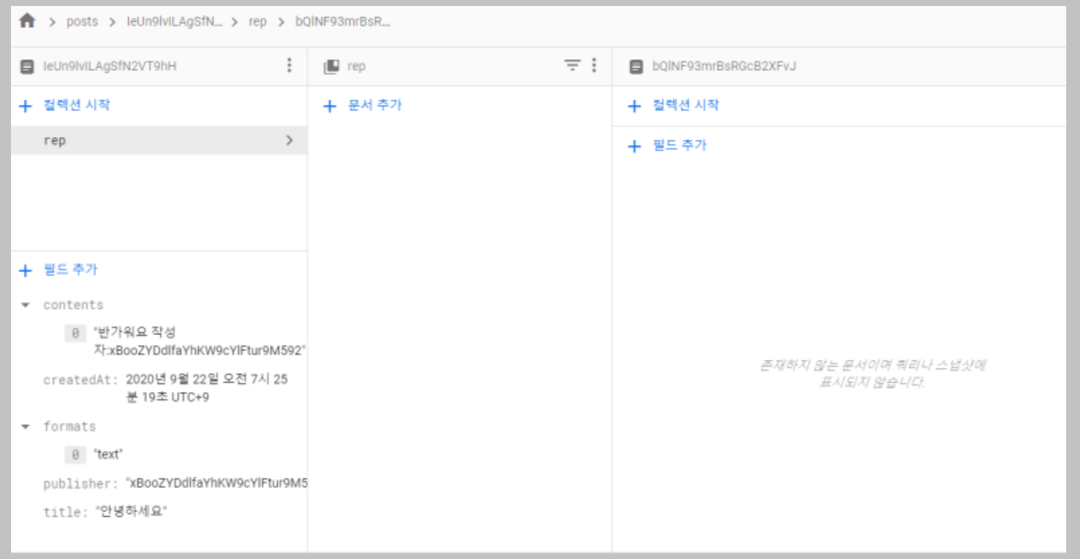
<JAVA 파일 코드>

FMPI 개발 – 댓글삭제

<FMPI APP 화면>



<FMPI APP 화면>

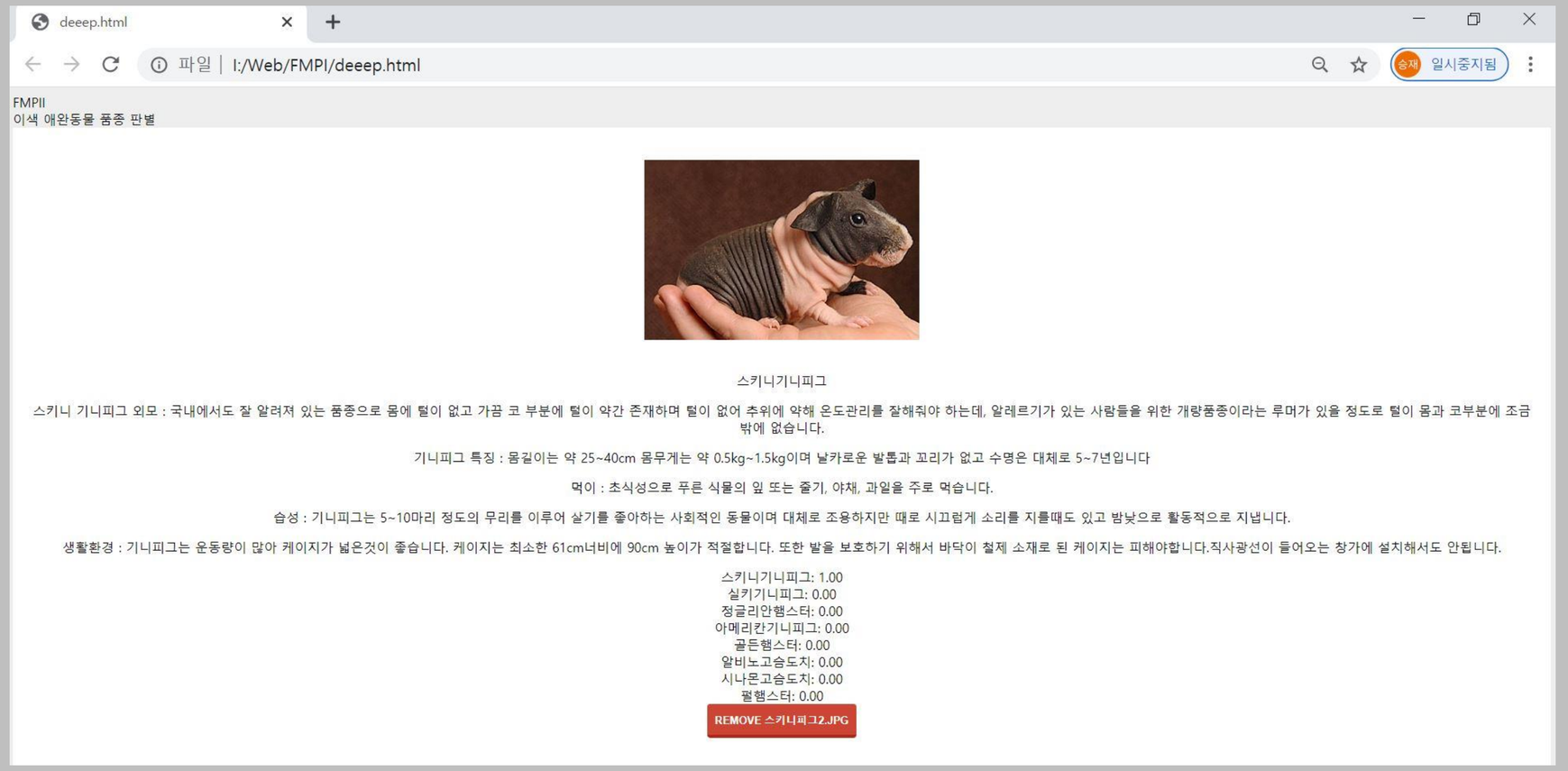


<Firestore에 댓글삭제된 화면>

```
public void btnClick1(View v){
    firebaseFirestore.collection( collectionPath: "posts")
            .document(id).collection( collectionPath: "rep")
            .document(mAdapter.mRepList.get(position).getRepId())
            .delete();
    Toast.makeText(getApplicationContext(), text: "삭제",
            Toast.LENGTH_SHORT).show();

}
```

<JAVA 파일 코드>

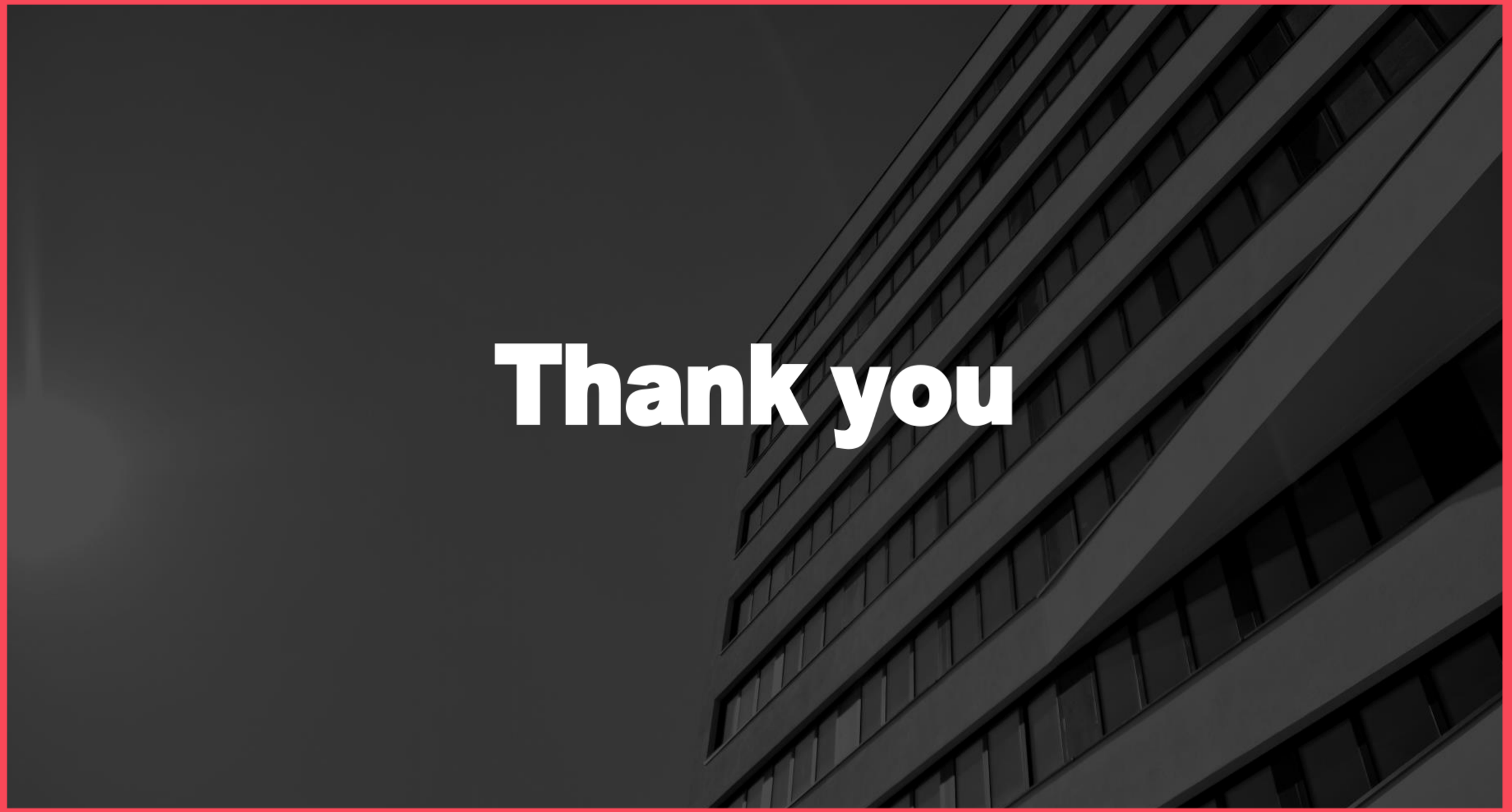
FMPI 개발 – FMPI웹 페이지



<FMPI의 웹사이트>

배운점 & 개선할점

시연 영상


Thank you