



언 어 학 습 프 로 그 램

4조

강유진, 우효진, 배광민, 강대준, 김강규, 박규영

CONTENTS

01. 프로젝트 시나리오

- 기획동기
- 벤치마킹

02. 프로젝트 기획

- 프로토 타입
- 개발 환경

03. 프로젝트 설계

- DB 구조
- 유스케이스 다이어그램
- 프로젝트 회의

04. 프로젝트 UI 설계

05. 프로젝트 구현

- 기능 구현 리뷰
- 오류 노트

06. 프로젝트 후기

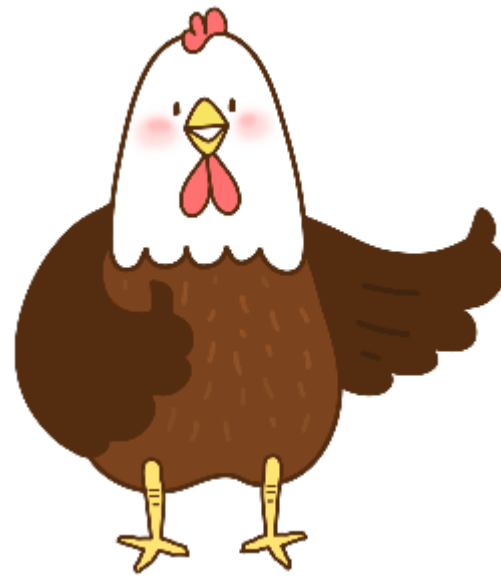
팀원 소개 및 역할



강유진

문법 학습

- 문법 리스트, 페이지 구현
- 발표 PPT 작성



김강규

디자인 프로토 , 영상OST 학습

- 프로토 타입 UI
- 영상으로 공부하기



박규영

잠금 화면

- 잠금 화면에 시계, 날짜, 단어장, 카메라 구현

팀원 소개 및 역할



배광민

단어장

- 번역기(api활용), 영어사전
- 프로젝트 발표



우효진

퀴즈 프로토 개발, 영상 학습
- 유튜브 api를 사용한 유튜브
플레이어 구현



강대준

퀴즈 개발, DB설계

- 퀴즈 페이지
- DB로 실시간 문제 업데이트
- 각자 만든 소스 하나로 합침

01. 프로젝트 시나리오

-기획동기

짧은기간내에우리가만들수있는게무엇일까생각하다가외국어공부에
관심이있어언어학습앱을만들기로했습니다.

언어학습앱을벤치마킹해보니광고도너무많고복잡하다는생각이들어
배우고싶은언어로각자공부하는방식대로앱을만들면어떨까싶어만들게
되었습니다.

01. 프로젝트 시나리오

-기획동기

프로젝트 이름은 투겡고 입니다.

투겡고라는 뜻은 two(2)+겡고(언어)라는 의미로 영어와 일본어만 학습하는 앱이기에 영어와 일본어로 합쳐 2개의 언어라는 뜻을 가지고 있습니다.

내가 배우고 싶거나, 부족한 문법, 단어를 넣어 학습한 후 퀴즈를 풀수 있고 영상으로 OST를 배울수 있고, 잠금 화면에서도 단어장을 확인할 수 있습니다.

01. 프로젝트 시나리오

-벤치마킹

[저절로 암기 영단어] 앱

[잠금 화면 구현]

- 우리가 핸드폰을 하면서 제일 많이 보는게 잠금 화면 입니다.
- 잠금 화면을 활용하여 DB에 있는 단어장과 해설, 이미지를 가져와서 잠금화면에서도 학습할 수 있다.



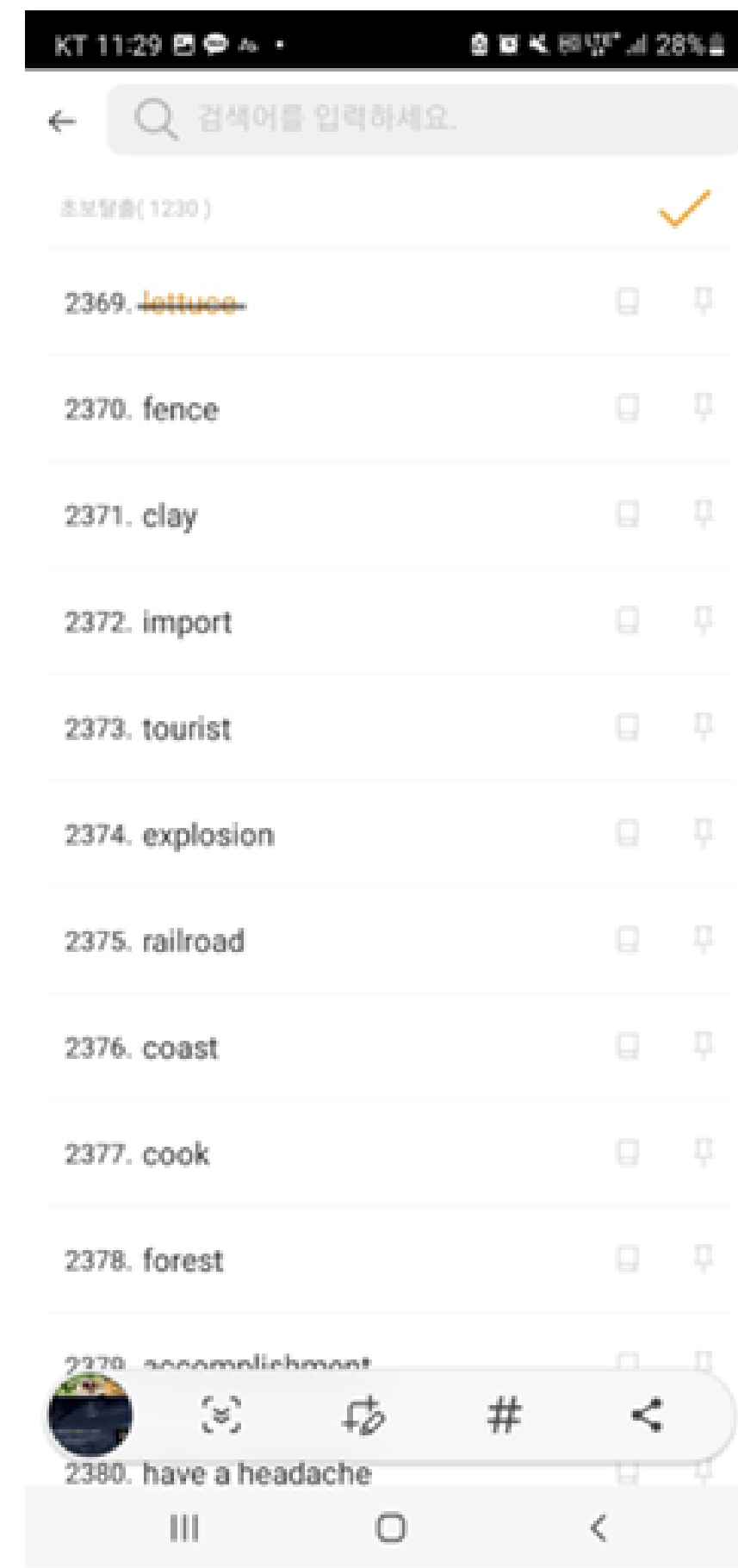
01. 프로젝트 시나리오

-벤치마킹

[저절로 암기 영단어] 앱

[단어장]

- 단어를 나열해 클릭하면 각 해설들을 볼 수 있고, 상단에 고정되며, 다 외운 단어들은 체크해 단어에 밑줄이 그어집니다.



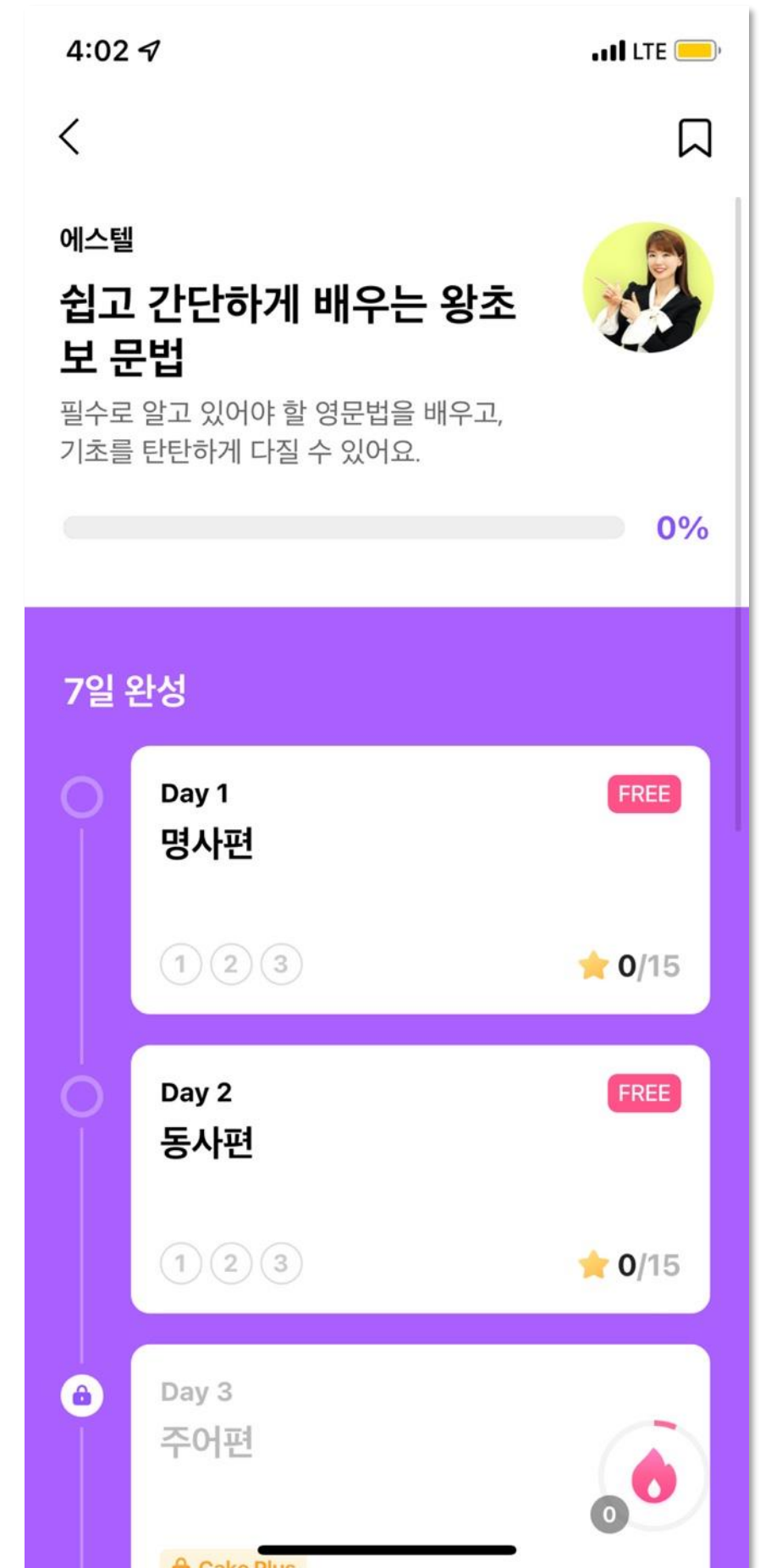
01. 프로젝트 시나리오

-벤치마킹

[cake] 앱

[문법 클래스] -> [day7 문법 리스트]

- 문법 배우는 페이지를 만들어 day7까지 학습할 수 있는 페이지입니다.
- 하루에 3단계의 학습을 만들어 완료하면 진행률이 표시가 됩니다.



01. 프로젝트 시나리오

-벤치마킹

[한본냥] 앱

[퀴즈 풀기]

- 단어, 문법 학습했던 내용으로 퀴즈 풀기
- 4지선다 활용하여 정답일 경우 PrograssBar 증가하게 구현합니다.



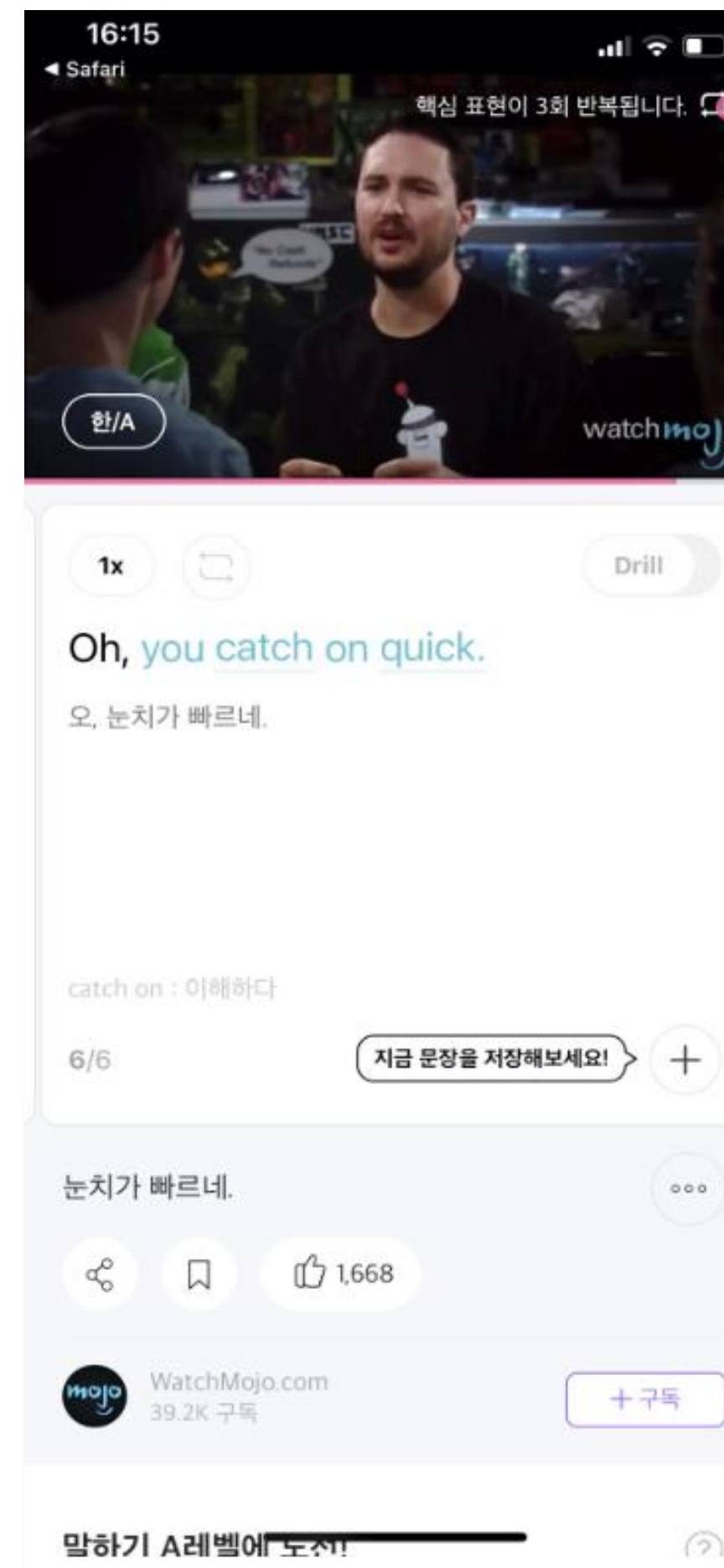
01. 프로젝트 시나리오

-벤치마킹

[cake] 앱

[영상 OST 배우기]

- 영상을 통해 OST를 듣고, 영어와 해석본을 보면서 학습할 수 있습니다.



02. 프로젝트 기획

- 프로토타입



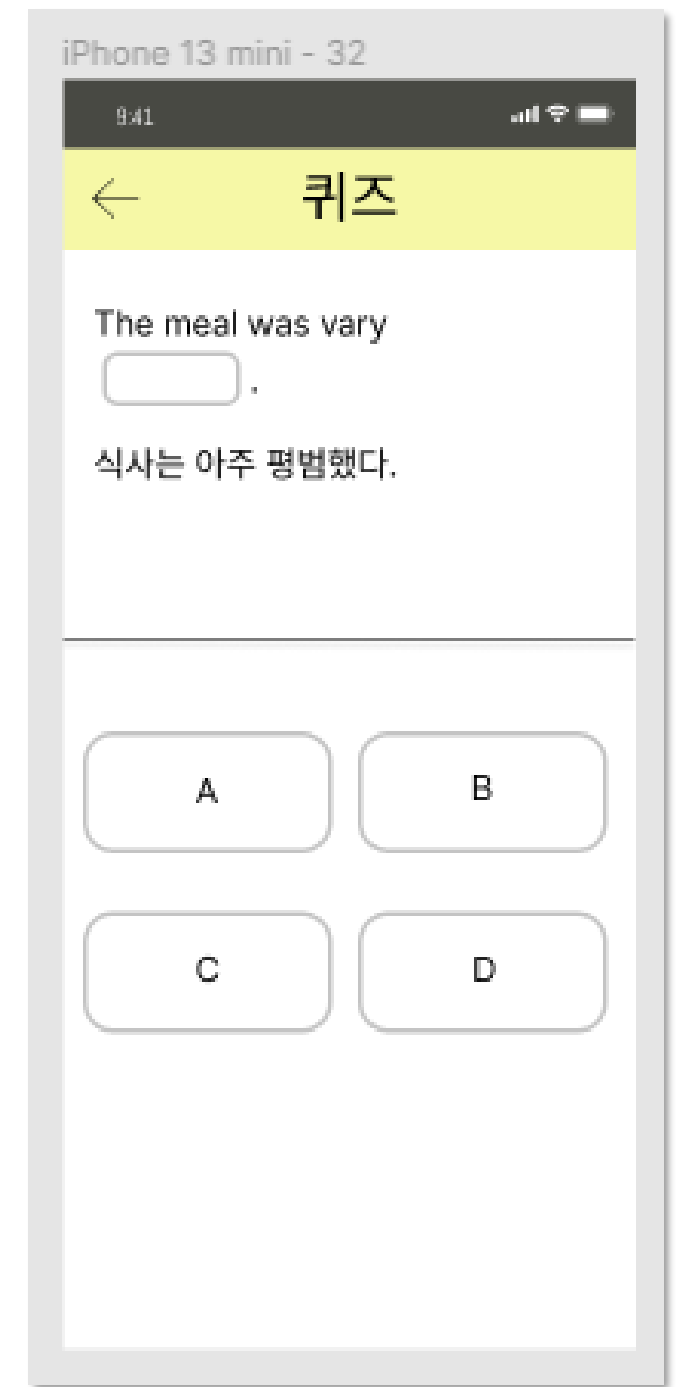
1. 프로젝트 홈 화면



2. 영어, 일본어 학습 화면



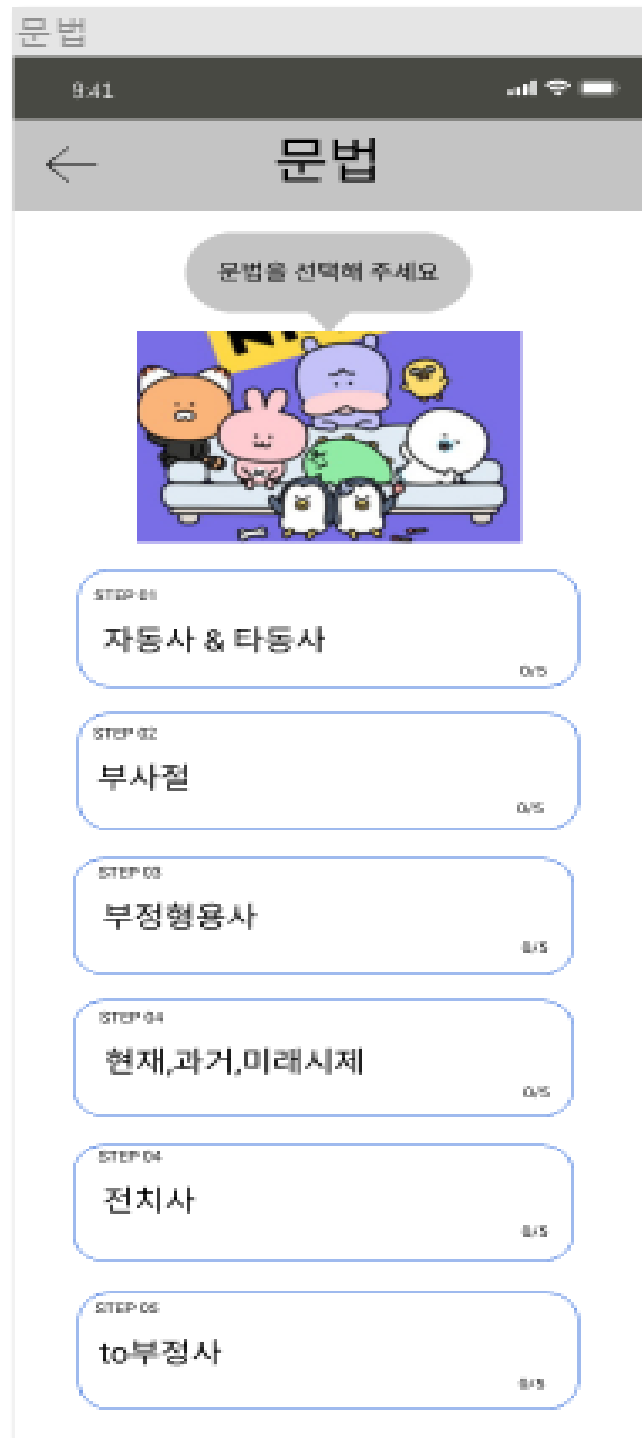
3. 잠금 화면



4. 퀴즈

02. 프로젝트 기획

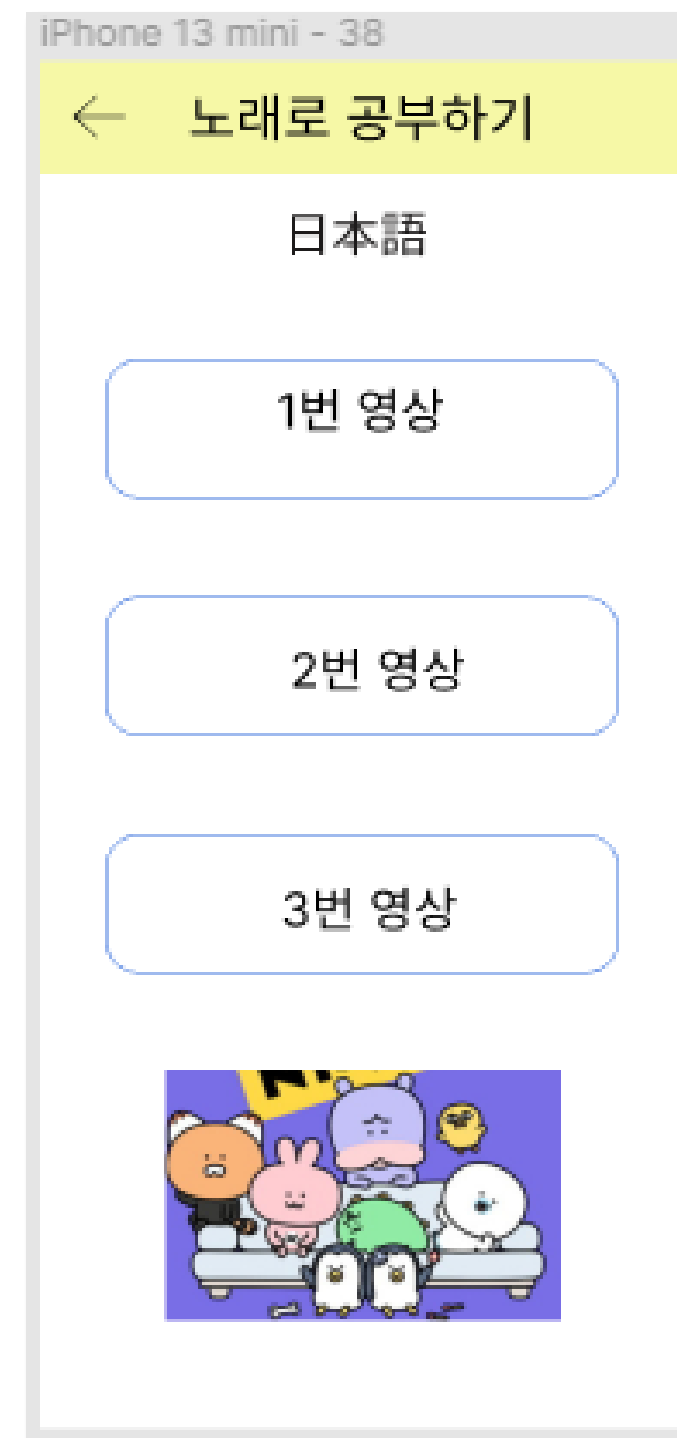
-프로토타입



5. 문법 학습 리스트



6. 문법 학습 페이지



7. 영상 학습 리스트



8. 영상 학습 페이지

02. 프로젝트 기획

-개발 도구&환경



Android
Studio



Figma



MariaDB



03. 프로젝트 설계

-DB구조

퀴즈DB

English Quiz

Question
Problem
Example
Answer

Japanese Quiz

Question
Problem
Example
Answer

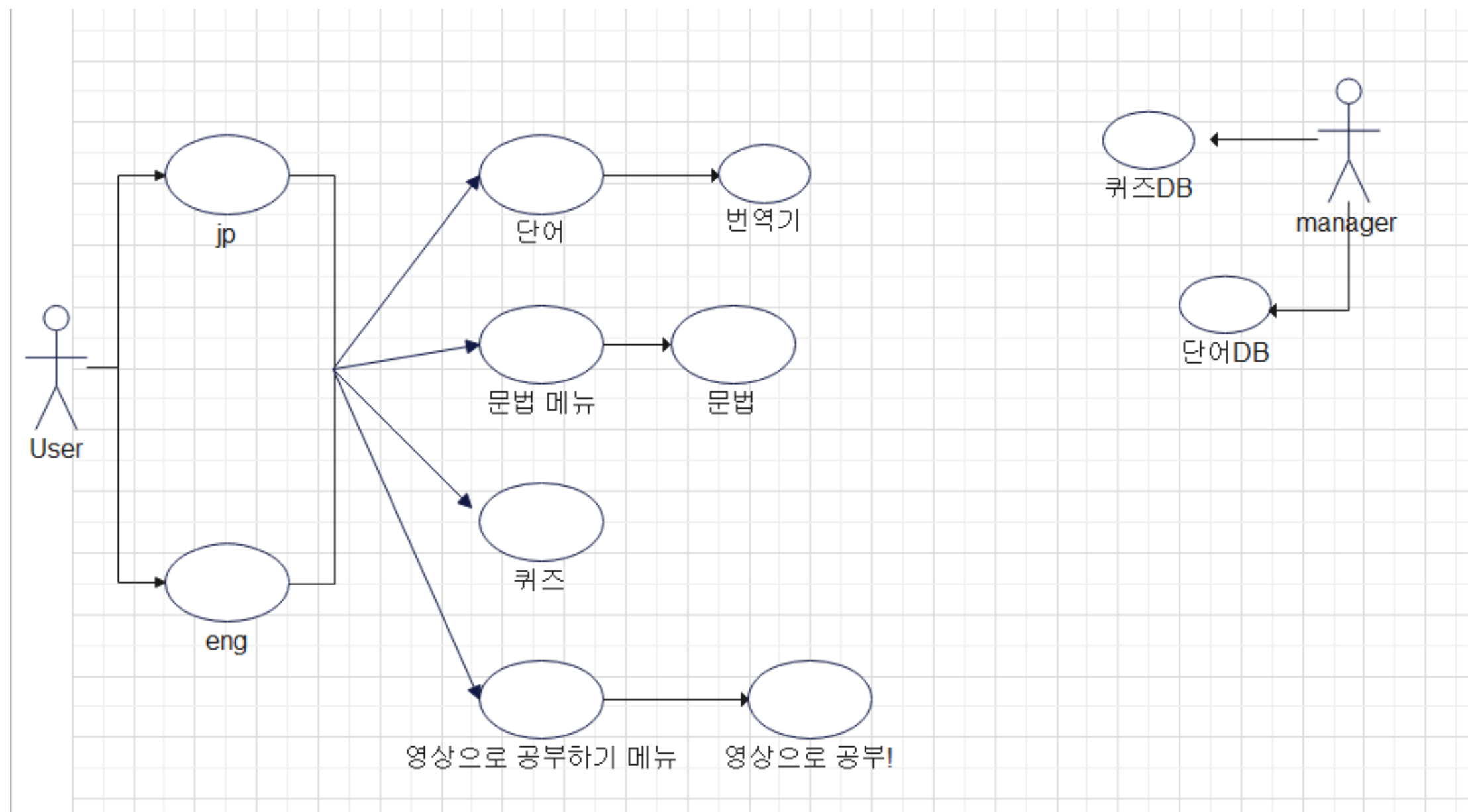
사전DB

engword		Untitled		
	idx	항목번호	Domain	TEXT
	word	단어	Domain	TEXT
	meaning	뜻	Domain	TEXT
	Clear	공부끝	Domain	TEXT
	STAR	중요단어	Domain	TEXT

jpanword		CopyOfUntitled		
	idx	항목번호	Domain	TEXT
	word	단어	Domain	TEXT
	meaning	뜻	Domain	TEXT
	Clear	공부끝	Domain	TEXT
	STAR	중요단어	Domain	TEXT

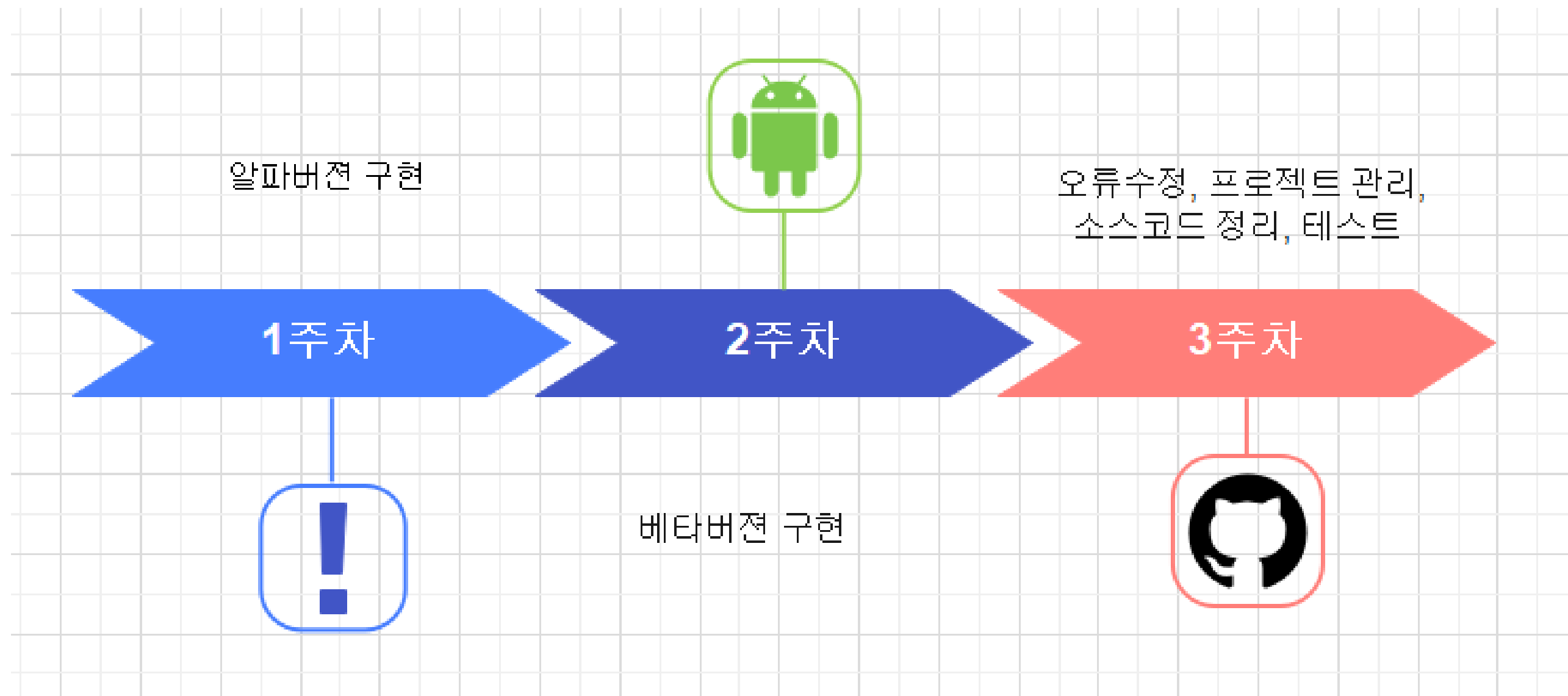
03. 프로젝트 설계

-유스케이스 다이어그램



03. 프로젝트 설계

-주차별 개발 현황



03. 프로젝트 설계

-회의록



- 브레인 스토밍
- 주제 선정
- 역할 분담
- DB설계
- 단계별 기획 설계

- 코드리뷰

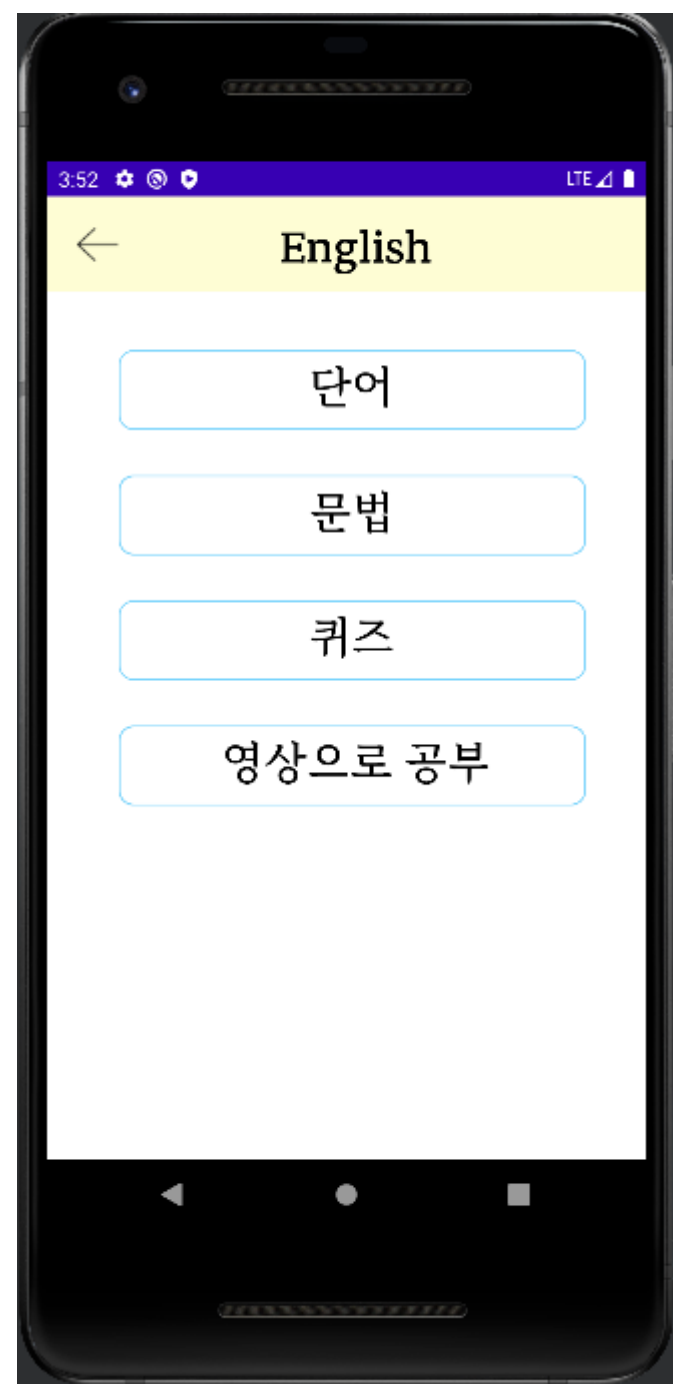
- 중간 점검
- 요구분석

- 오류노트 리뷰
- 마지막 코드리뷰

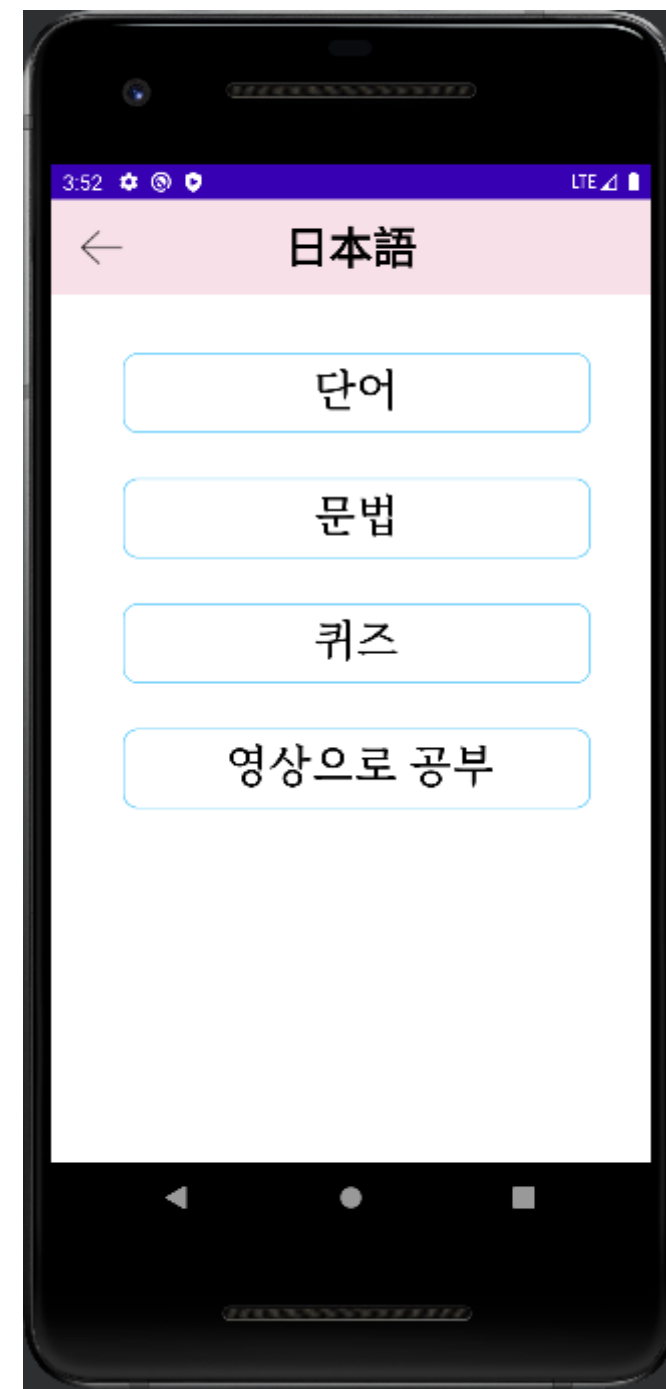
04. 프로젝트 UI 설계



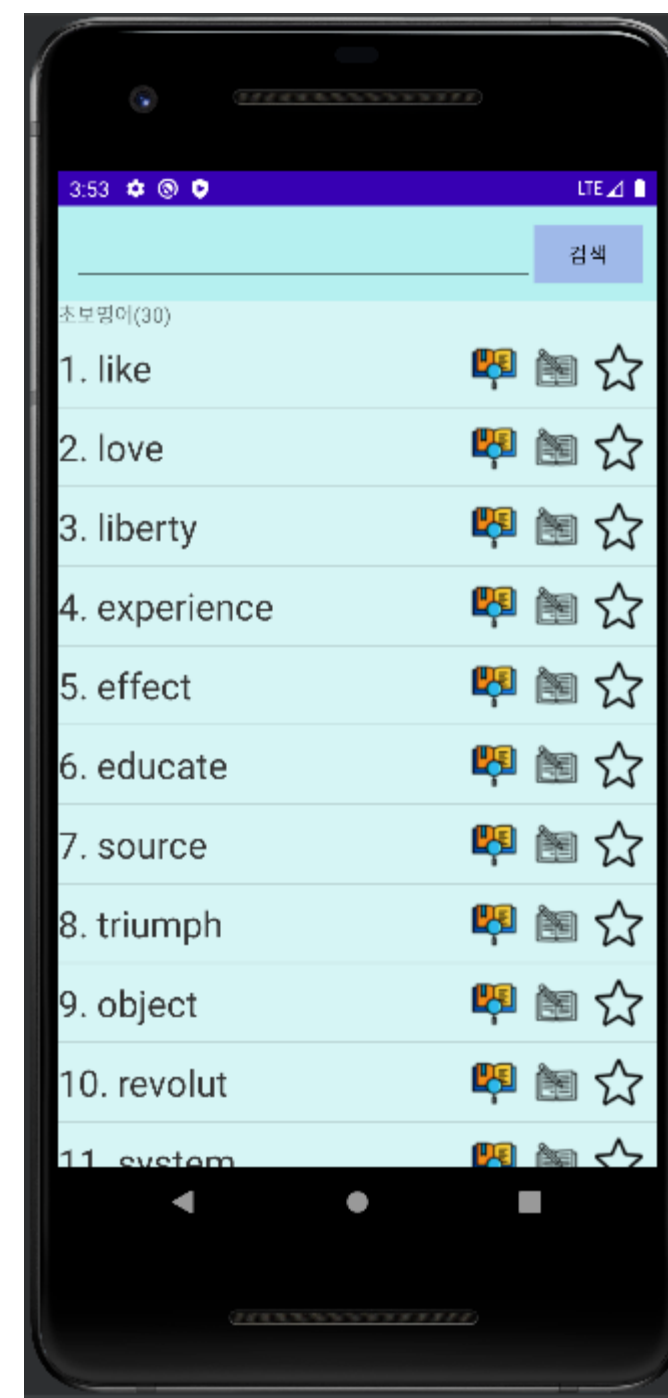
프로젝트 홈 화면



영어 메인화면

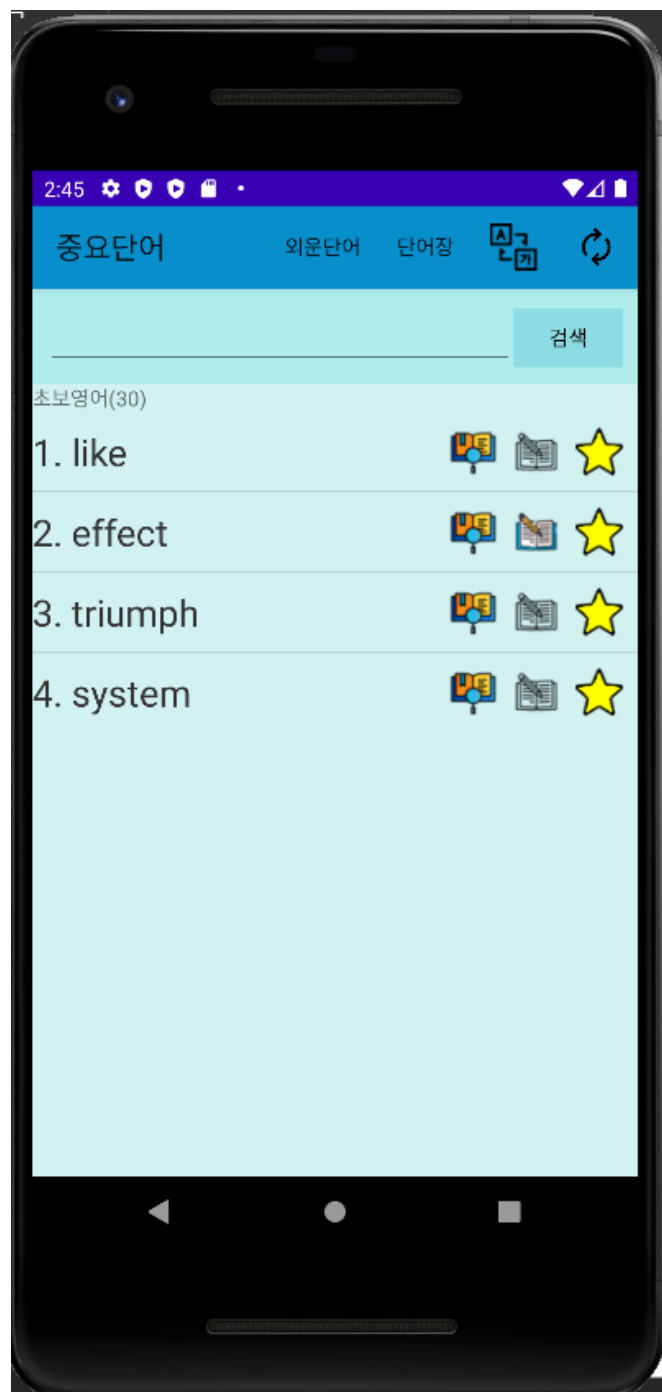


일본어 메인화면



영어 단어장

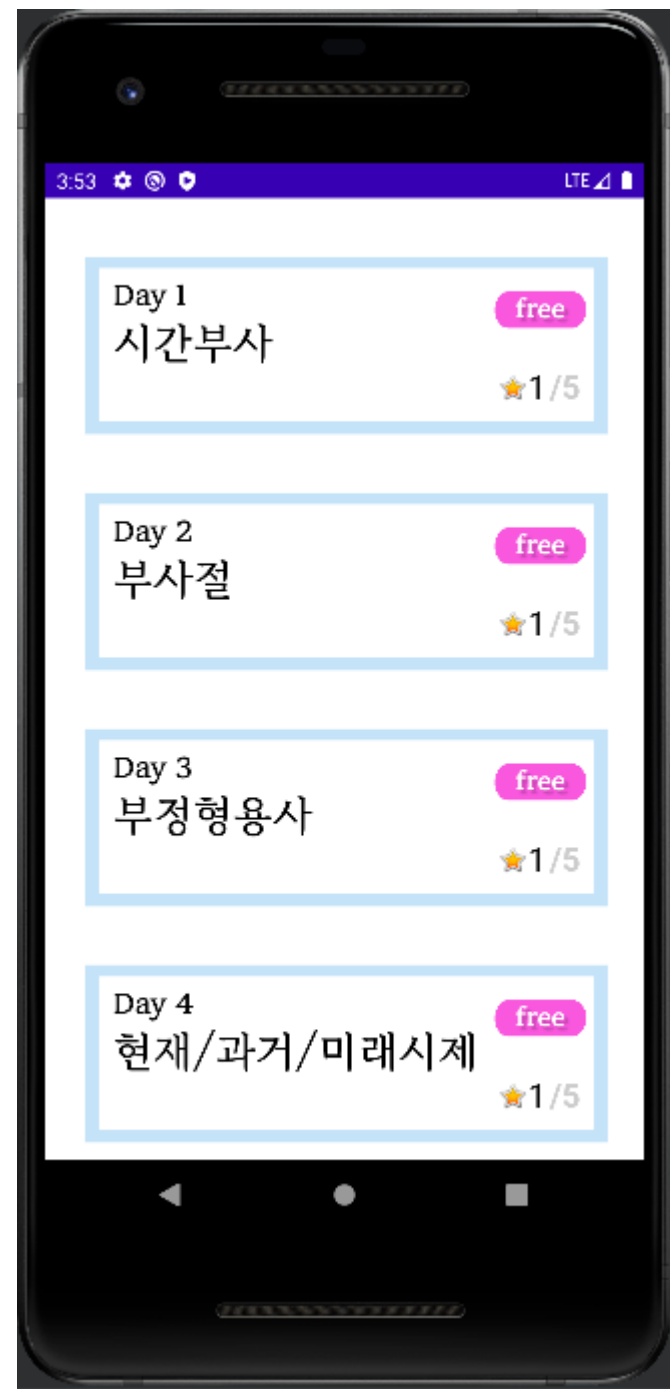
04. 프로젝트 UI 설계



중요, 외운 단어



번역기



영어 문법 리스트

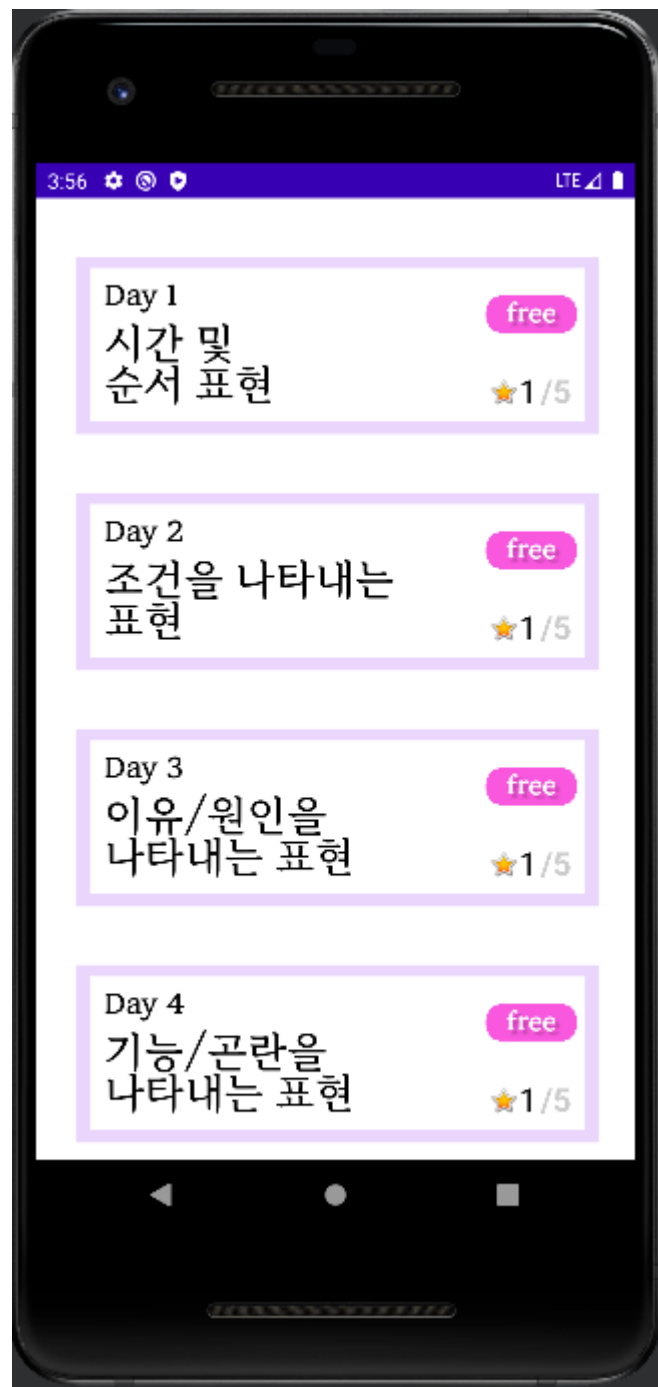


영어 문법 페이지

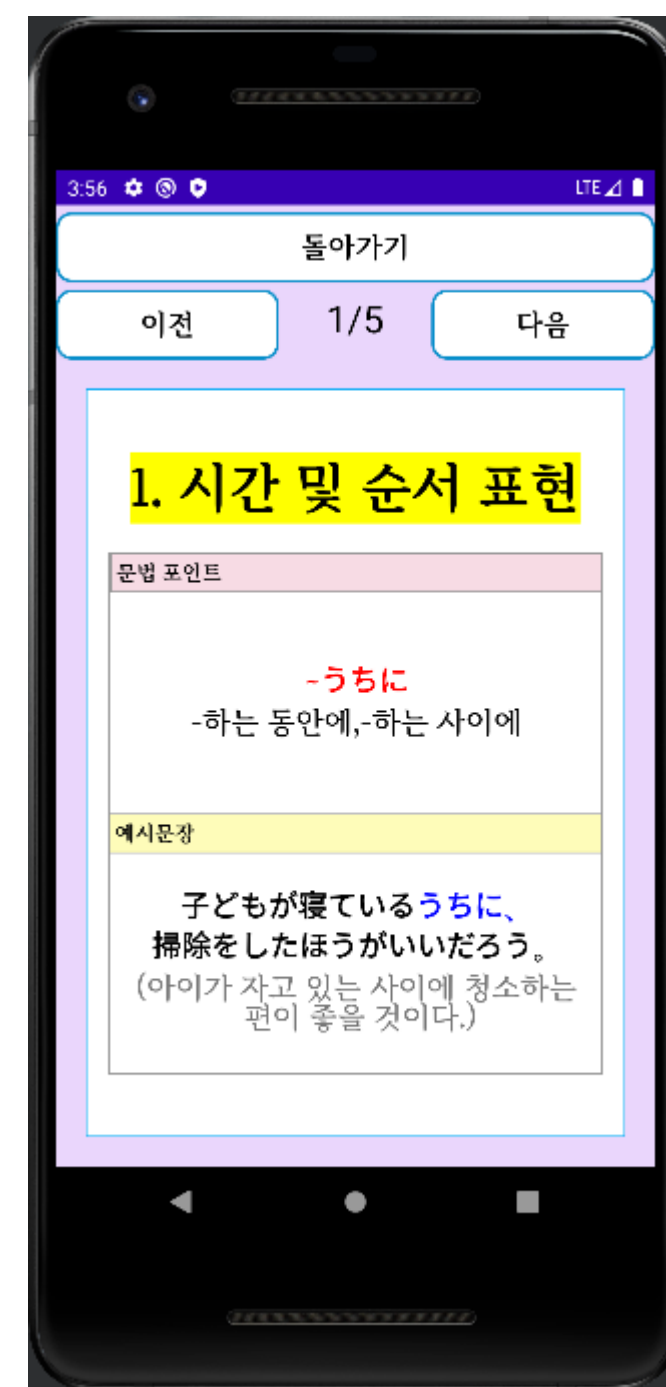
04. 프로젝트 UI 설계



일본어 단어장

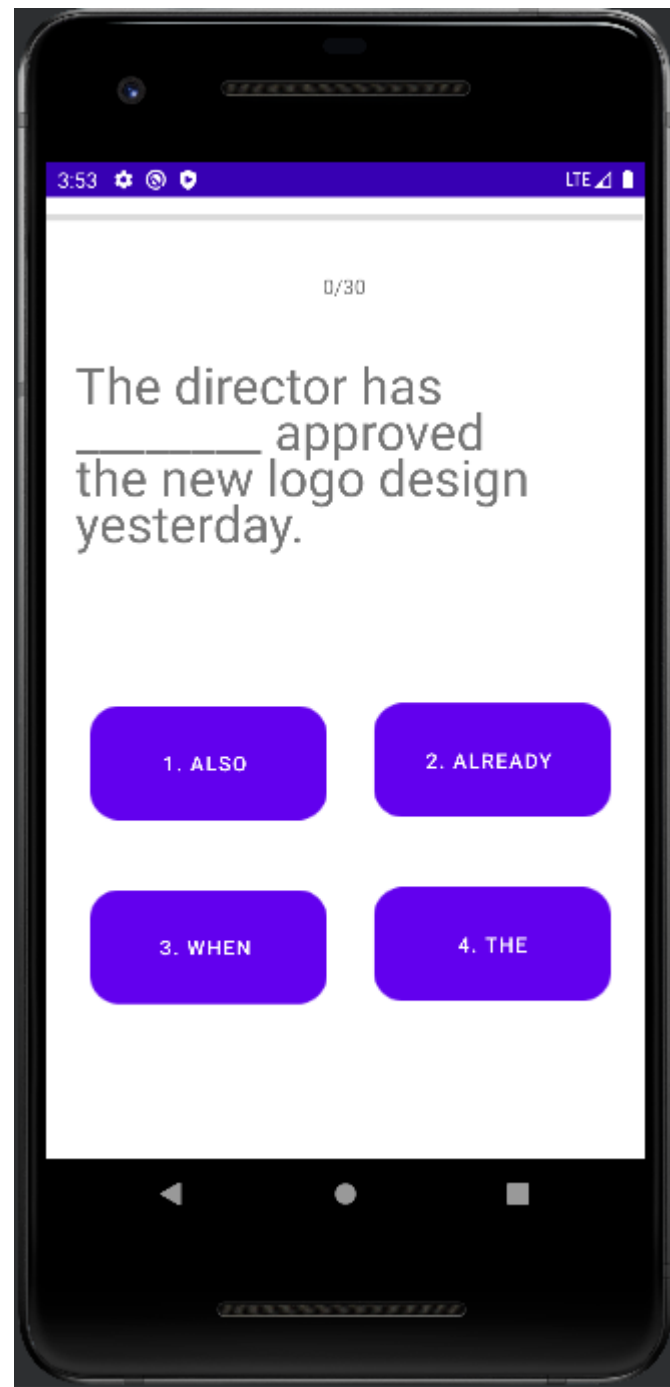


일본어 문법 리스트

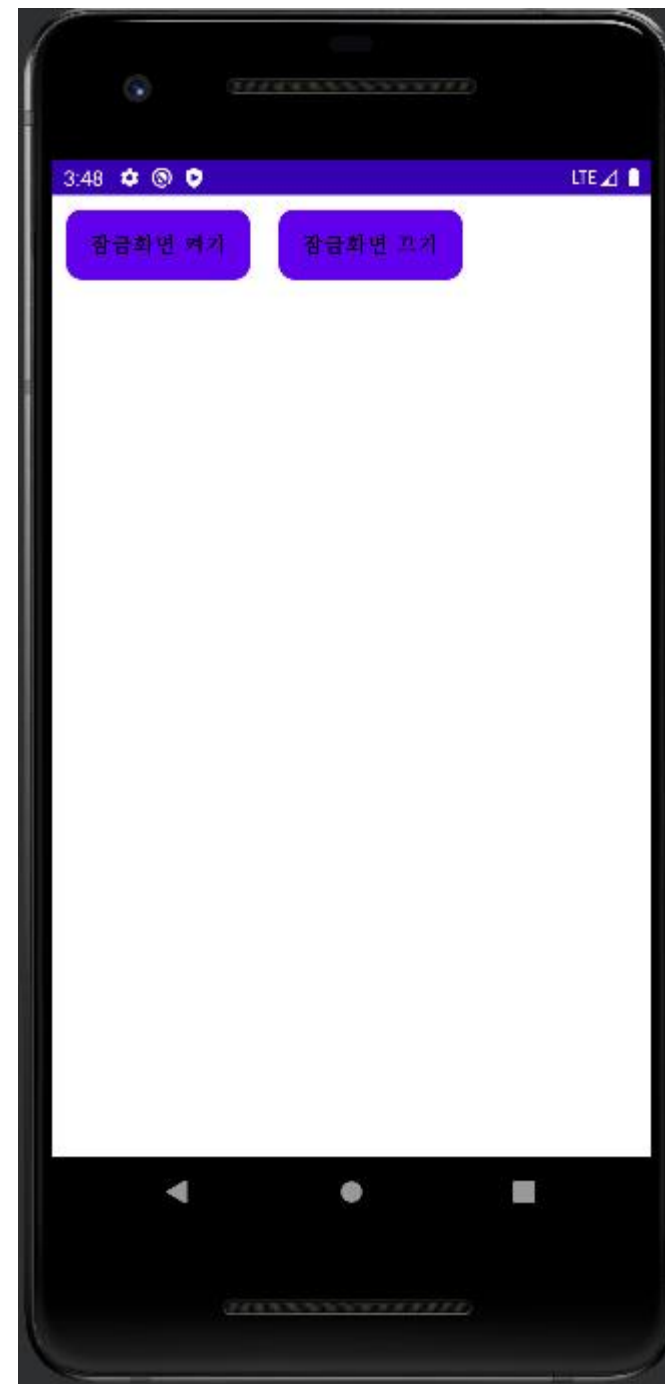


일본어 문법 페이지

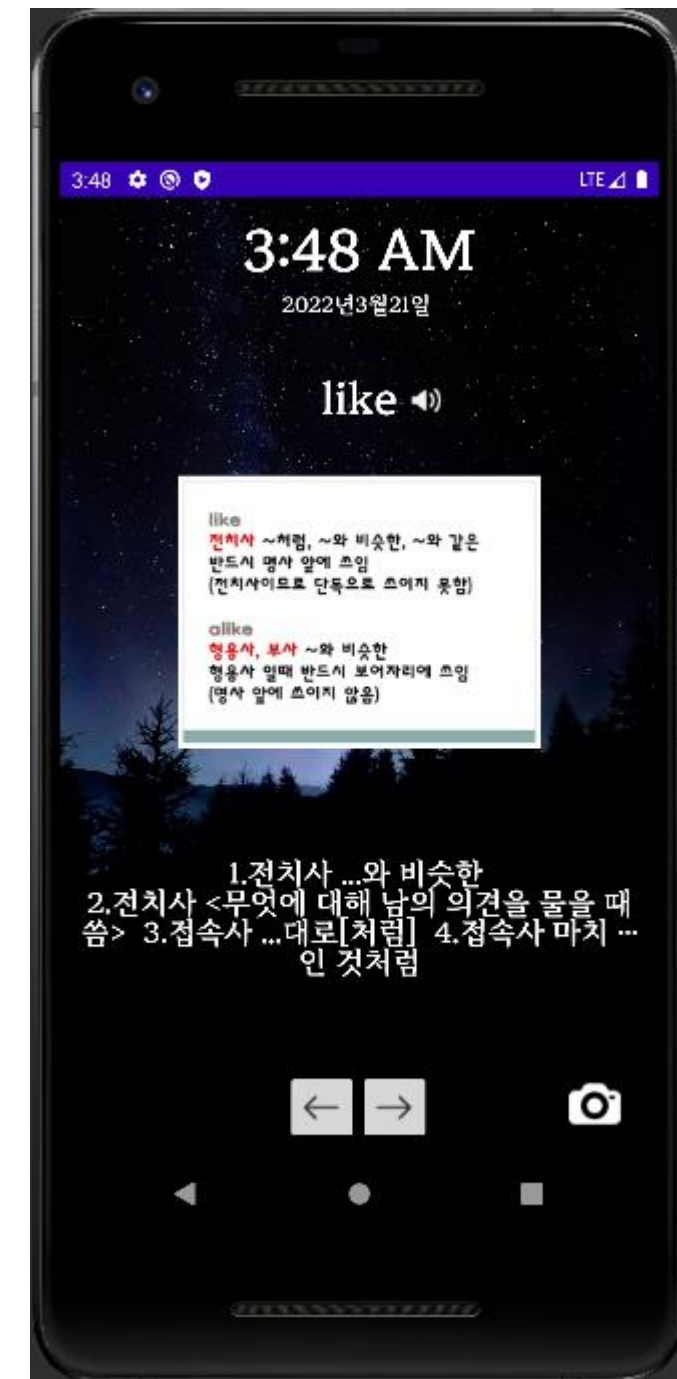
04. 프로젝트 UI 설계



영어 퀴즈

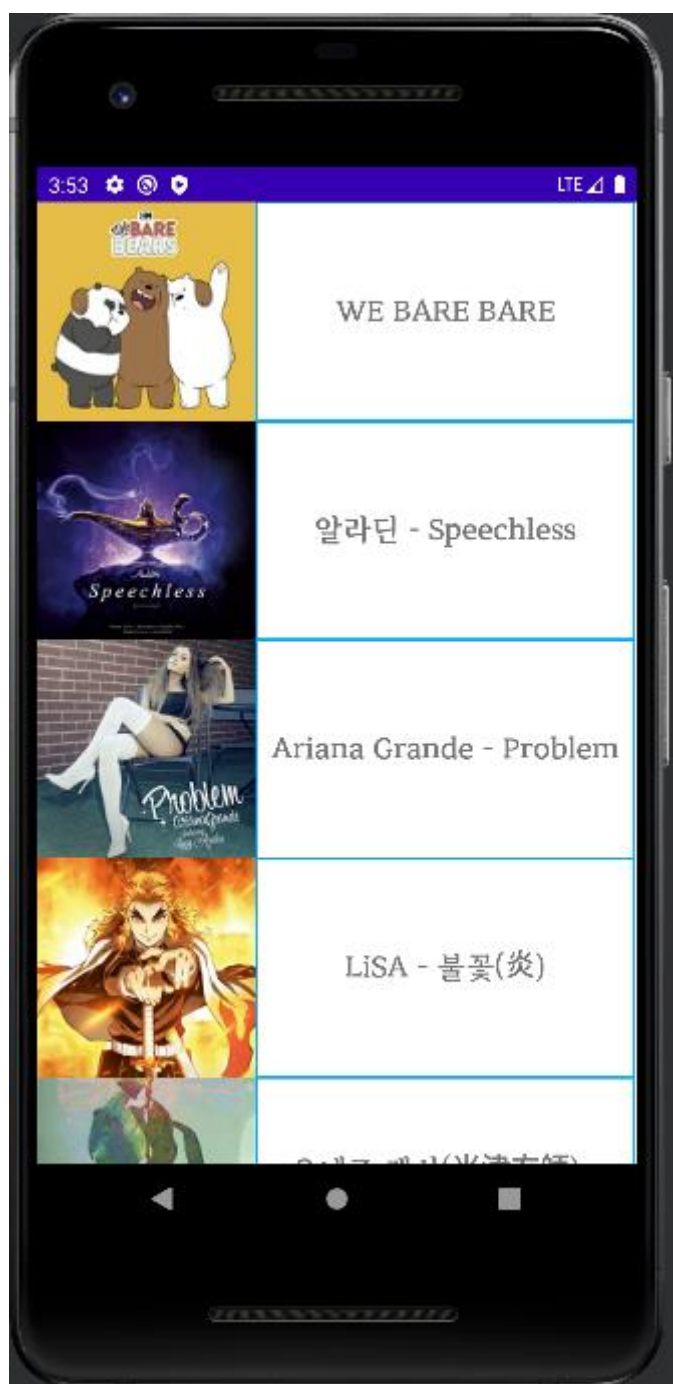


잠금 화면

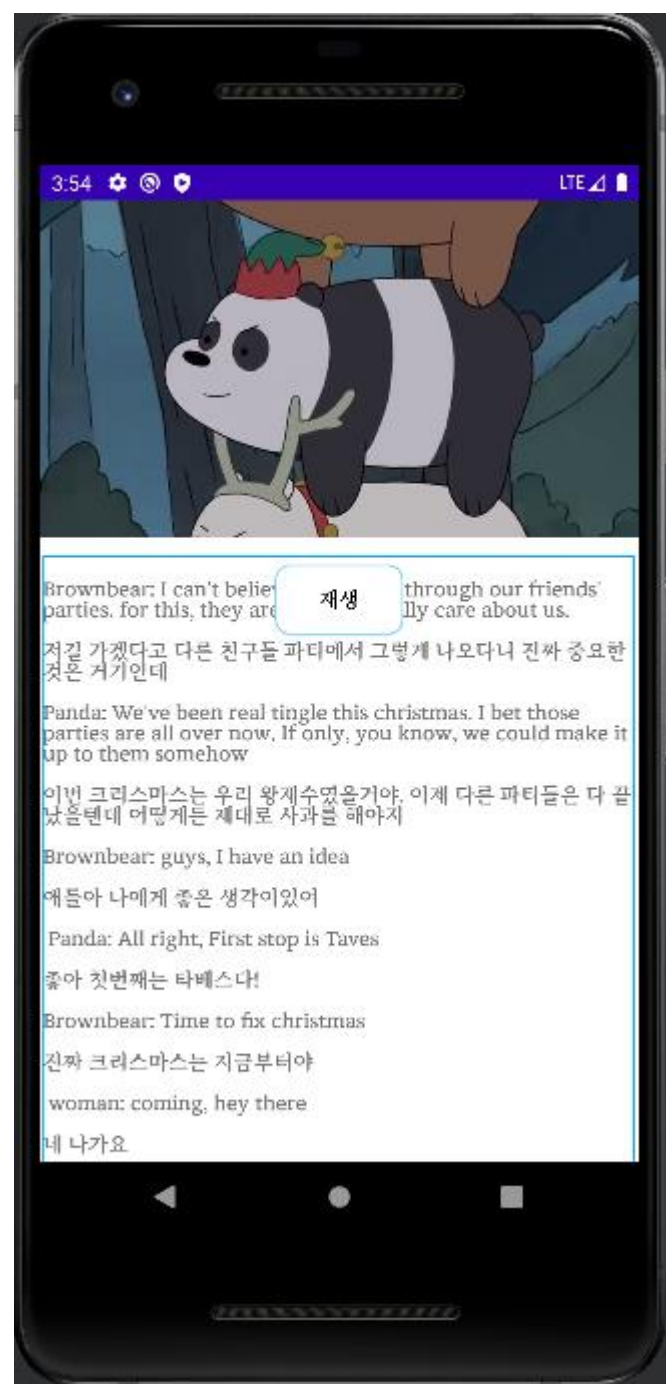


잠금 화면

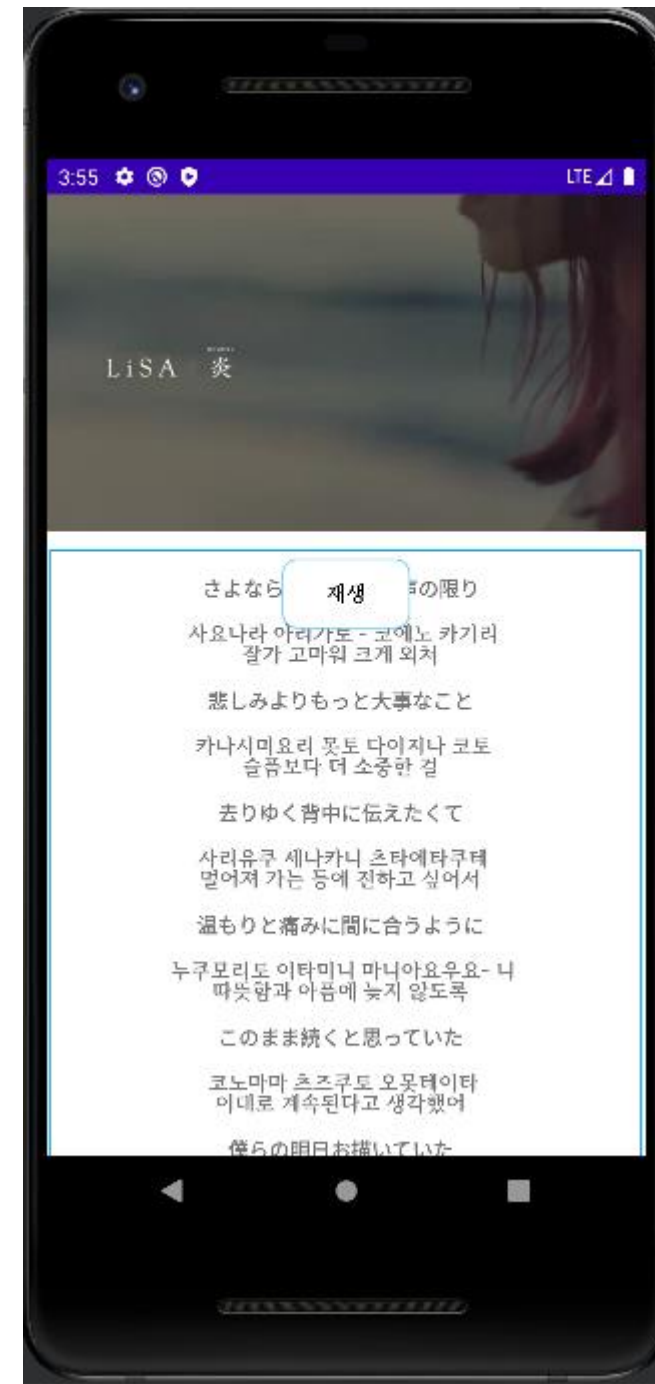
04. 프로젝트 UI 설계



영상 학습 리스트



영어 영상 학습

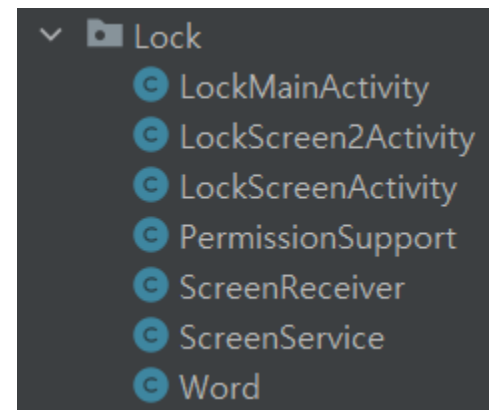
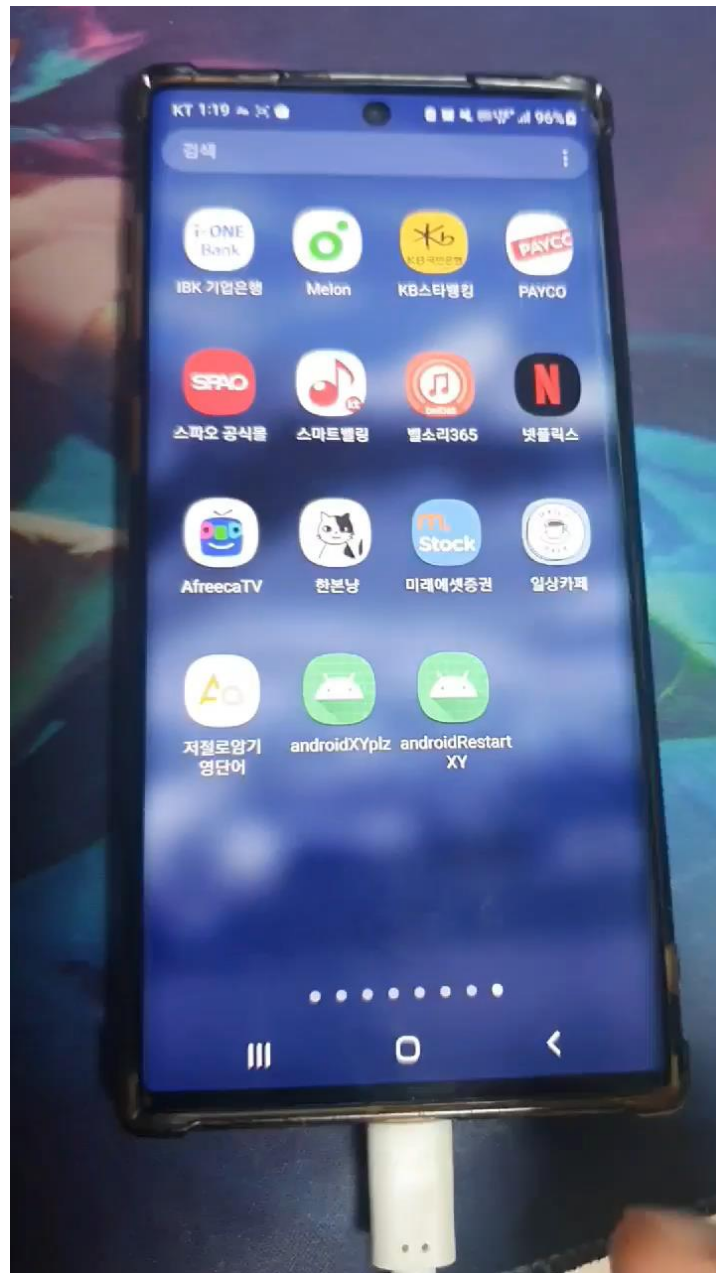


일본어 영상 학습

04.시연 영상

05. 프로젝트 구현

- 잠금화면(권한)



잠금 화면 관련 클래스

```
// 권한 체크
private void permissionCheck() {

    // PermissionSupport.java 클래스 객체 생성
    permission = new PermissionSupport( _activity: this, _context: this);

    // 권한 체크 후 리턴이 false로 들어오면
    if (!permission.checkPermission()){
        //권한 요청
        permission.requestPermission();
    }
}
```

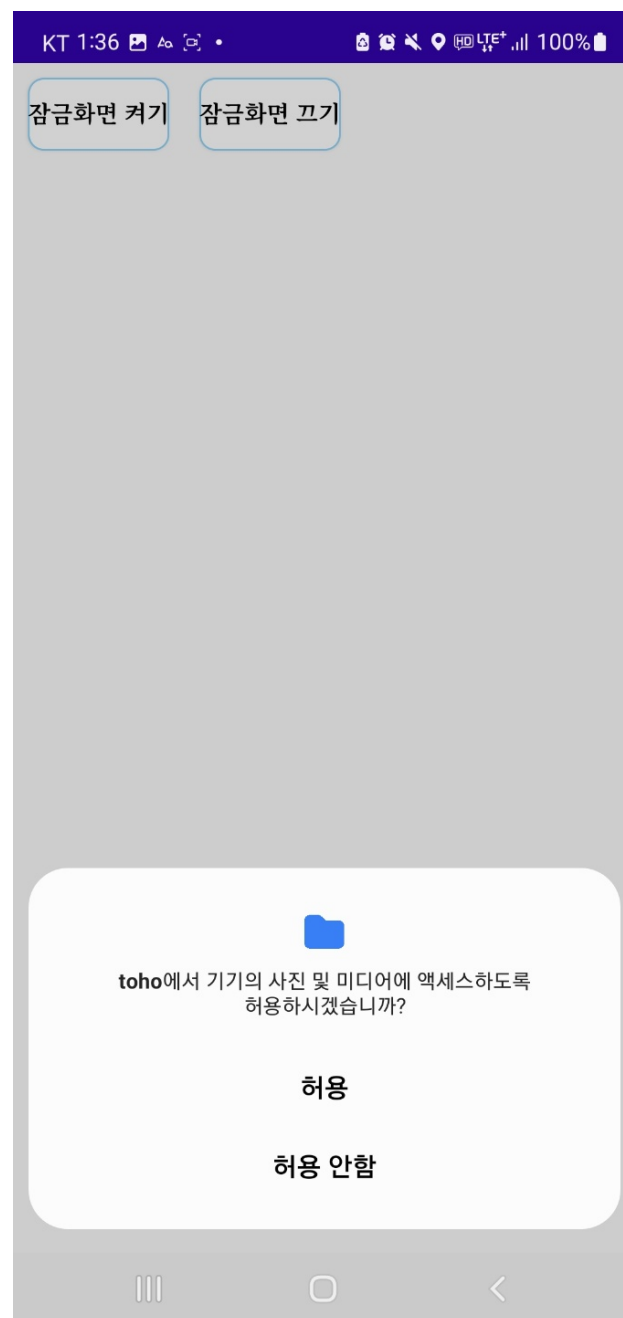
버튼을 누르면 권한 여부를 확인 후 권한요청을 합니다.

```
//생성자에서 Activity와 Context를 파라미터로 받아
public PermissionSupport(Activity _activity, Context _context){
    this.activity = _activity;
    this.context = _context;
}
```

생성자에서 Activity와 context를 인자로 받습니다.

05. 프로젝트 구현

- 잠금화면(권한)



```
//요청할 권한 배열 저장
private String[] permissions = {
    Manifest.permission.CAMERA,
    Manifest.permission.READ_EXTERNAL_STORAGE,
    Manifest.permission.ACCESS_FINE_LOCATION
};
```

요청할 권한을 배열로 저장하여

```
//배열로 선언한 권한 중 허용되지 않은 권한 있는지 체크
public boolean checkPermission() {
    int result;
    permissionList = new ArrayList<>();

    for(String pm : permissions){
        result = ContextCompat.checkSelfPermission(context, pm);
        if(result != PackageManager.PERMISSION_GRANTED){
            permissionList.add(pm);
        }
    }
    if(!permissionList.isEmpty()){
        return false;
    }
    return true;
}
```

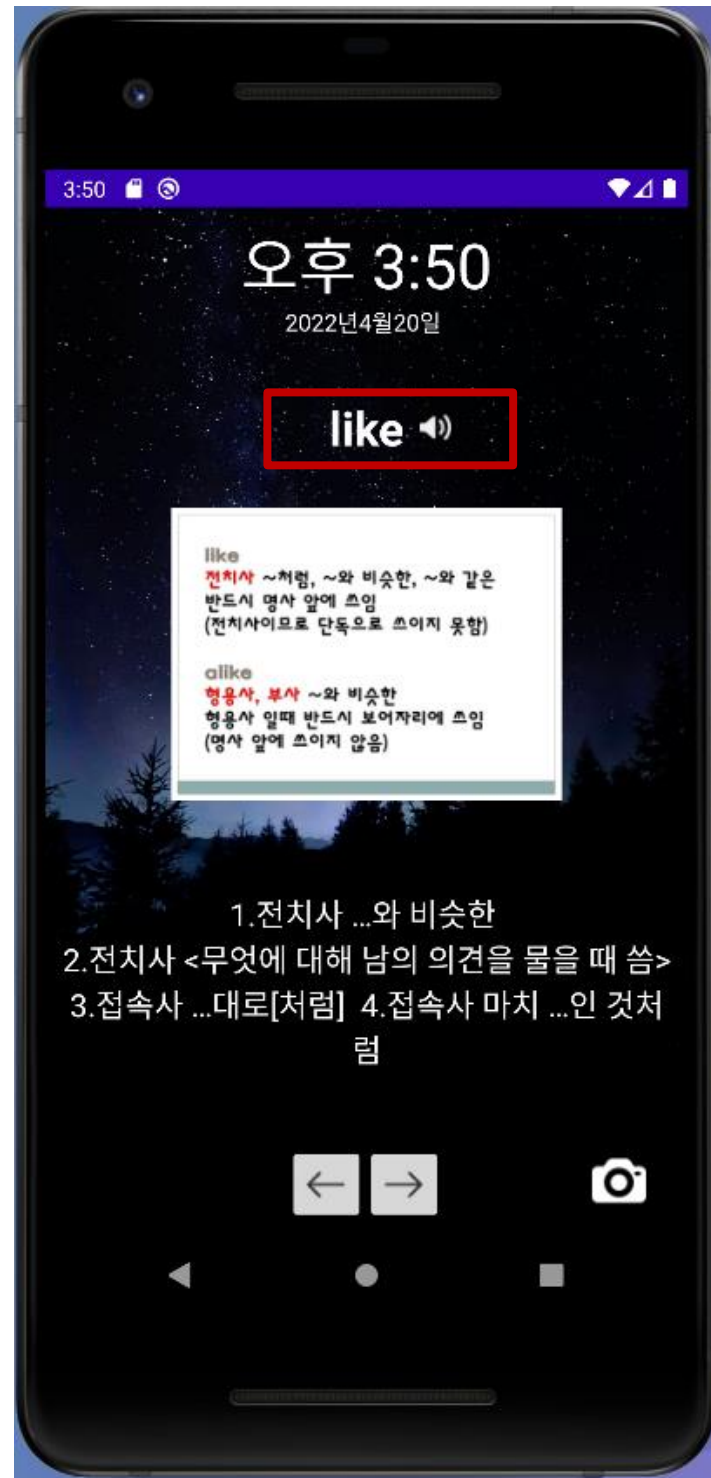
배열로 선언된 권한중 허용되지 않은 권한이 있는지 체크해줍니다.

```
//배열로 선언한 권한에 대해 사용자에게 허용 요청
public void requestPermission(){
    ActivityCompat.requestPermissions(activity,
        (String[]) permissionList.toArray(
            new String[permissionList.size()]),
        MULTIPLE_PERMISSIONS);
}
```

허용되지 않은 권한은 요청해줍니다.

05. 프로젝트 구현

- 잠금화면 구현



```
Intent intent = new Intent(getApplicationContext(), ScreenService.class);
startService(intent);
```

```
public void onCreate() {
    super.onCreate();
    mReceiver = new ScreenReceiver();
    IntentFilter filter = new IntentFilter(Intent.ACTION_SCREEN_OFF);
    registerReceiver(mReceiver, filter); // 브로드 캐스트 리시버를 등록한다.
}
```

```
if (intent.getAction().equals(Intent.ACTION_SCREEN_OFF)) { // 화면이 꺼졌다는 인텐트를 받으면
    Intent i = new Intent(context, LockScreenActivity.class); // LockScreenActivity를 띄운다.
    i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_SINGLE_TOP);
    context.startActivity(i);
}
```

```
@Override
public int onStartCommand(Intent intent, int flags, int startId){
    // 서비스를 강제로 종료했을 때 서비스를 어떤 방법으로 다시 시작시킬지 결정.
    super.onStartCommand(intent, flags, startId);
    if(intent != null){
        if(intent.getAction()==null){
            if(mReceiver==null){
                mReceiver = new ScreenReceiver();
                IntentFilter filter = new IntentFilter(Intent.ACTION_SCREEN_OFF);
                registerReceiver(mReceiver, filter);
            }
        }
    }
    return START_REDELIVER_INTENT;
    // 이후 서비스 재생성 가능, 강제로 종료되기 전에 전달된 마지막 인텐트를 다시 전달해주는 기능 포함.
}
```

• In LockScreenActivity

```
getWindow().addFlags(WindowManager.LayoutParams.FLAG_SHOW_WHEN_LOCKED
    | WindowManager.LayoutParams.FLAG_DISMISS_KEYGUARD);
// 안드로이드 잠금화면보다 위에 액티비티를 띄우게 해주는 코드
```

• 모든 권한이 승인되면 ScreenService를 실행해줍니다.

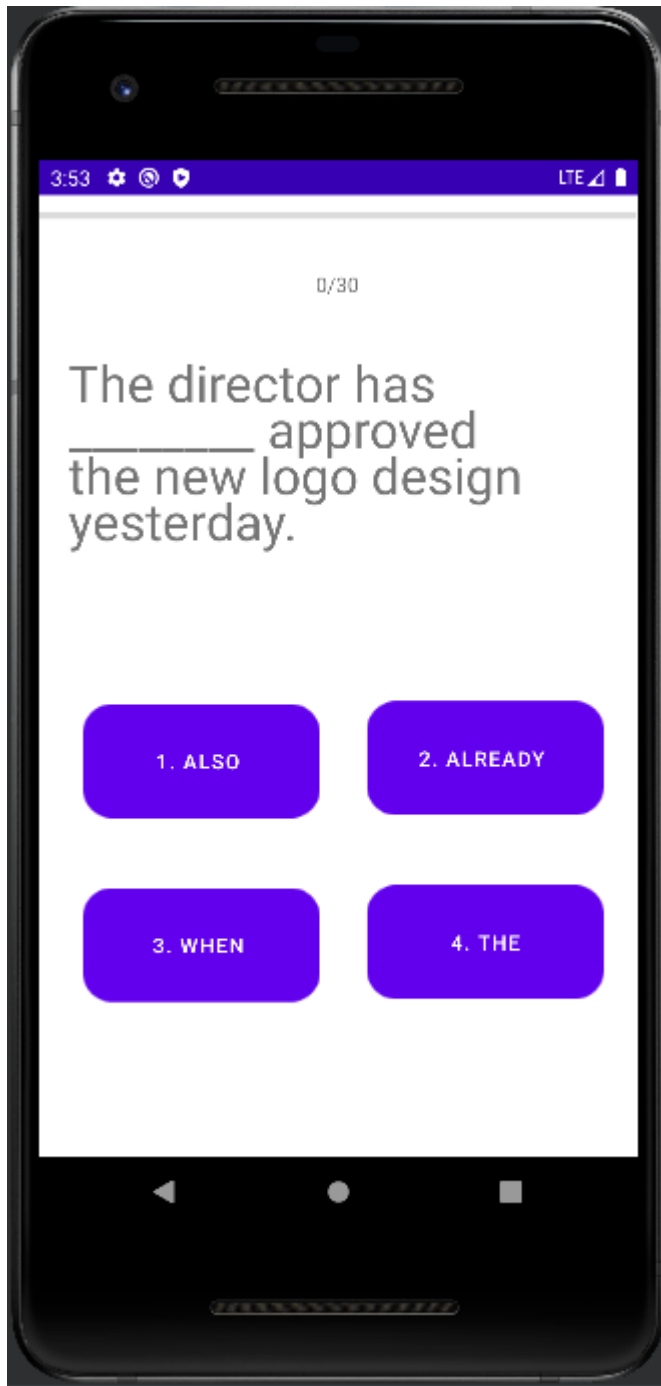
• 스크린 서비스가 start시 브로드캐스트 리시버를 등록시켜줍니다.

• 화면이 꺼졌다는 신호를 받으면 LockScreenActivity 실행시킵니다.

• 서비스를 강제로 종료했을때 다시 실행시켜줍니다.

05. 프로젝트 구현

-기능구현리뷰



▶ 퀴즈 화면 구현(Quiz Activity.java)

```
initLoadDB(); //DB로드
SetListener(); // 입력리스너 로드
SettingQuiz( num: 0);

progressBar.setMax(quizList.size());
QuizSequence quizth = new QuizSequence(); //진행바, UI 변경
han = new th1handler();
Thread th1 = new Thread(quizth);
th1.start();
```

- initLoadDB() 메서드에서 현재 퀴즈DB에 저장된 정보를 가져오는 역할을 합니다.
- 퀴즈, ProgressBar를 실행 시켜주는 것은 Thread로 만들었습니다.

```
public void SettingQuiz(int num){ // 퀴즈 세팅

    if(quizList.size()!=problem_Count) {
        String[] buts_Examples = quizList.get(num).getExample().split(regex: "\\n");

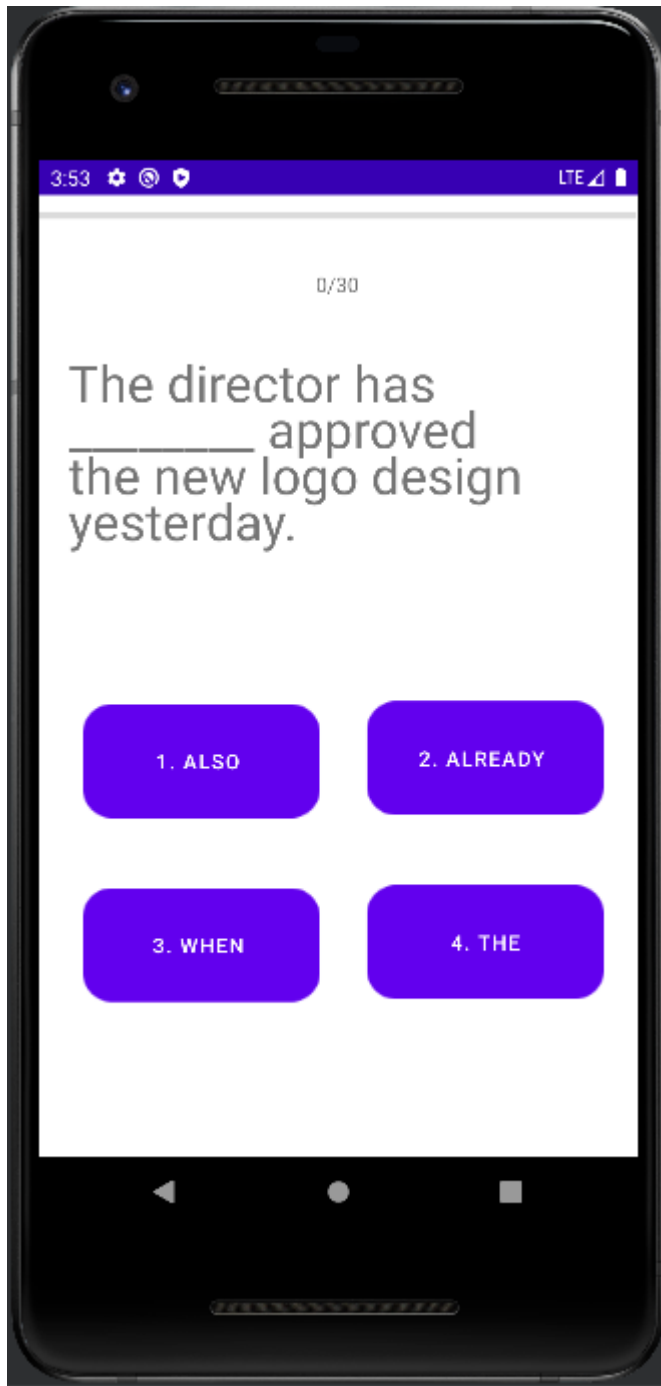
        Problem.setText(quizList.get(num).getProblem());

        for (int i = 0; i < buts_Examples.length; i++) {
            buts[i].setText(buts_Examples[i]);
        }
    }
}
```

- TextView 와 Button의 Text를 알맞은 번호의 퀴즈로 바꿔준다.
Button 의 Text에는 예시들이 들어가는데
ex) 1.this
2.is
- 이런 식으로 줄넘김으로 돼있어서 split으로 분리했다.

05. 프로젝트 구현

-기능구현리뷰



▶ 퀴즈 화면 구현 (quizResultCheck() Method)

```
public void quizResultCheck(){
    Intent intent_this = getIntent();
    String lang = intent_this.getStringExtra( name: "lang");
    if(quizList.size() != problem_Count) {
        if (quizList.get(problem_Count).getAnswer().equals(Answer)) {
            System.out.println("정답입니다" +problem_Count);

            problem_Count++;

            Answer = " ";
            SettingQuiz(problem_Count);

            if(problem_Count == quizList.size()){

                System.out.println("끝" + problem_Count);
                //팝업창
                Intent intent = new Intent( packageContext: this, ResultActivity.class);
                intent.putExtra( name: "wrong", Integer.toString(Wrong_Count));
                intent.putExtra( name: "problem",Integer.toString(problem_Count));
                intent.putExtra( name: "lang",lang);
                resultLauncher.launch(intent);

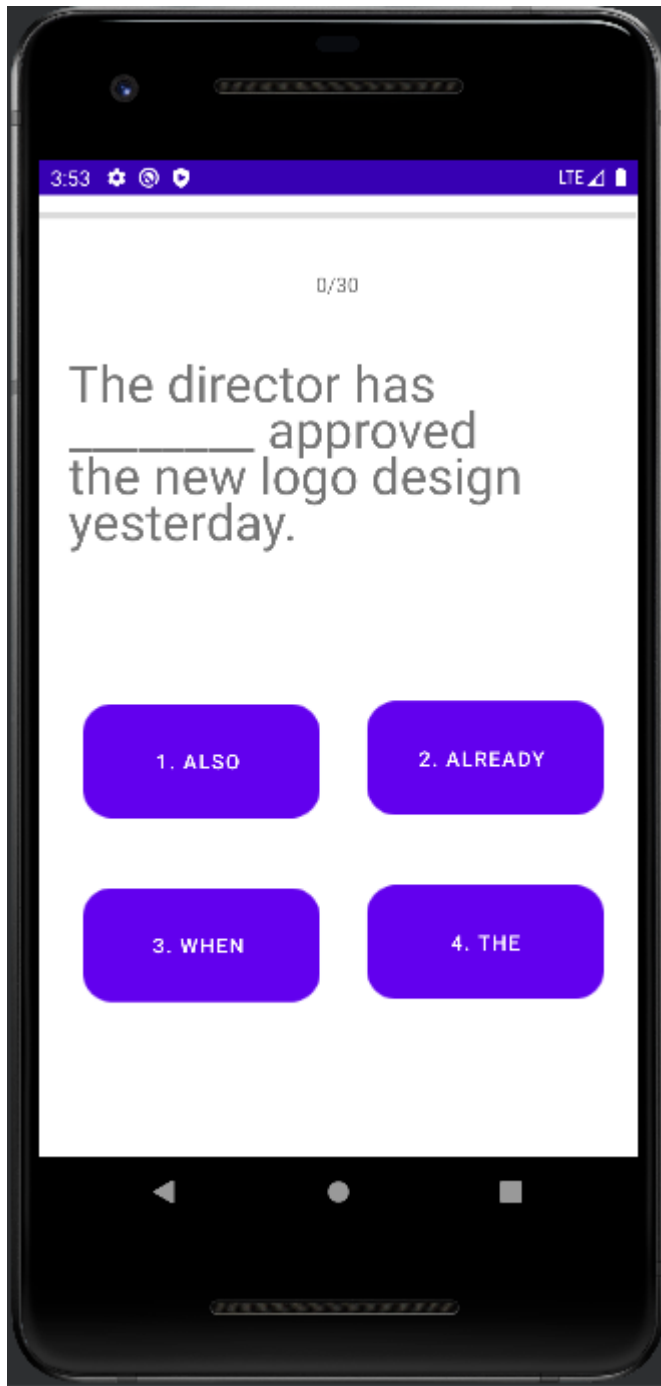
            }

        } else {
            Wrong_Count++;
            System.out.println("오답입니다" +Wrong_Count);
        }
    }else if(quizList.size() == problem_Count){
        Intent intent = new Intent( packageContext: QuizActivity.this , Btn_jp.class);
        startActivity(intent);
    }
}
```

- 퀴즈 정답을 체크해 주는 메서드
- 정답일 경우 Problem_Count를 1 증가시켜 다음 퀴즈로 넘어갈 수 있게 만들었다.
- 만약 퀴즈가 끝났을 경우 총 문제 수, 틀린 횟수의 정보를 팝업창을 만들어 결과를 전달하고 표시하게 만들었다.

05. 프로젝트 구현

-기능구현리뷰



▶ 퀴즈 화면 구현(QuizSequence)

```
class QuizSequence implements Runnable { //퀴즈 시퀀스
    int problem_Count = 0;
    @Override
    public void run() {

        while (true) {

            if(problem_Count == quizList.size()){
                problem_Count = 0;
                System.out.println(problem_Count);
                break;
            }else {
                settingProgress(data);
            }
            try {
                Thread.sleep( millis: 100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }

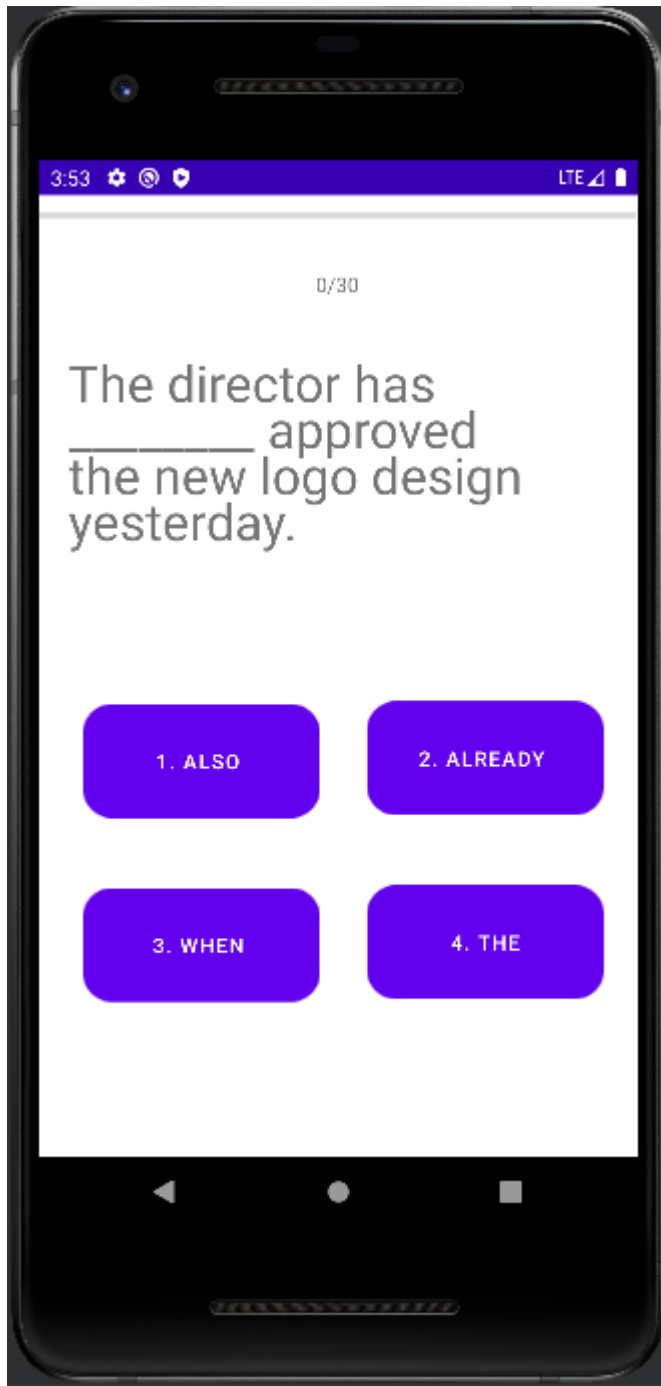
    class th1handler extends Handler{
        @Override
        public void handleMessage(Message msg){
            if(msg.what==0){
                super.handleMessage(msg);
                String progressbar_Num = msg.getData().getString( key: "data");
                String progress_Count = msg.getData().getString( key: "data2");

                ProgressCounter.setText(progressbar_Num + "/" + progress_Count);
                progressBar.setProgress(Integer.parseInt(progressbar_Num));
            }
        }
    }
}
```

- 퀴즈의 ProgressBar, Handler, Thread를 담당하는 쪽으로 만들었다.
- 위에 문제 수, 정답 수의 정보를 받아와 ProgressBar에 표시해준다.

05. 프로젝트 구현

-기능구현리뷰



▶ 퀴즈 화면 구현(RequestURLConnection.java)

```
URL url = new URL(_url);
URLConnection urlConn = (URLConnection) url.openConnection();

// [2-1]. urlConn 설정.
urlConn.setConnectTimeout(1000);
urlConn.setRequestMethod("POST"); // URL 요청에 대한 메소드 설정 : POST.
urlConn.setRequestProperty("Accept-Charset", "UTF-8"); // Accept-Charset 설정.
urlConn.setRequestProperty("Context_Type", "application/x-www-form-urlencoded;cahrset=UTF-8");

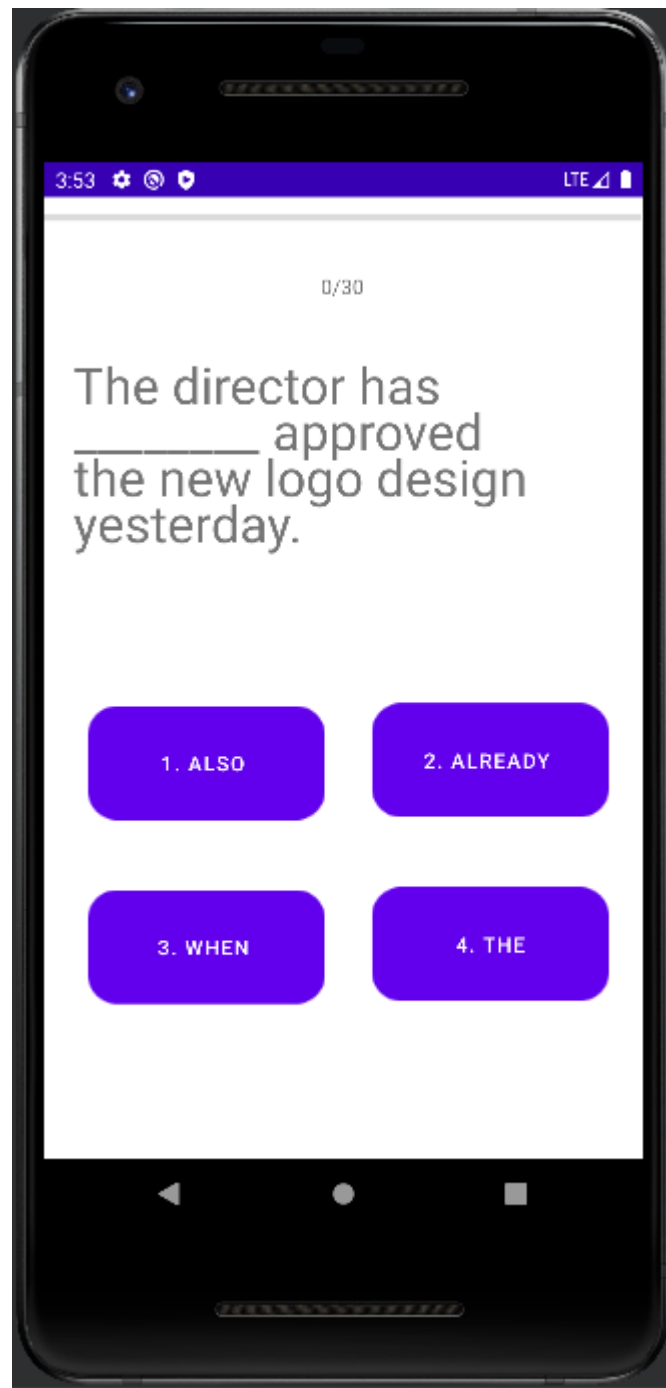
// [2-2]. parameter 전달 및 데이터 읽어오기.
String strParams = sbParams.toString(); //sbParams에 정리한 파라미터들을 스트링으로 저장. 예)id=id1&pw=123;
OutputStream os = urlConn.getOutputStream();
os.write(strParams.getBytes(charsetName: "UTF-8")); // 출력 스트림에 출력.
os.flush(); // 출력 스트림을 플러시(비운다)하고 버퍼링 된 모든 출력 바이트를 강제 실행.
os.close(); // 출력 스트림을 닫고 모든 시스템 자원을 해제.

// [2-3]. 연결 요청 확인.
// 실패 시 null을 리턴하고 메서드를 종료.
if (urlConn.getResponseCode() != HttpURLConnection.HTTP_OK) {
    System.out.println("null");
    return null;
}
```

- 지정된 서버의 URL 정보를 받아와 전달해 주는 역할을 하고
setConnectTimeout을 넣어서 서버가 켜져 있는지 꺼져 있는지에 대한 콜백을 받아옵니다

05. 프로젝트 구현

-기능구현리뷰



▶ 퀴즈 화면 구현(DataBaseHelper, AsyncTask.class)

```
public static class NetworkTask extends AsyncTask<Void, Void, String> {  
  
    private String url;  
    private ContentValues values;  
    private String lang;  
    public String json_file;  
    public JSONArray jsonArray;  
  
    public NetworkTask(String url, ContentValues values, String lang) {  
        this.url = url;  
        this.values = values;  
        this.lang = lang;  
    }  
  
    @Override  
    protected String doInBackground(Void... params) {  
  
        String result; // 요청 결과를 저장할 변수.  
        RequestHttpURLConnection requestHttpURLConnection = new RequestHttpURLConnection();  
        result = requestHttpURLConnection.request(url, values); // 해당 URL로 부터 결과물을 얻어온다.  
        System.out.println("result" +result);  
        return result;  
    }  
}
```

- RequestHttpURLConnection의 객체를 생성해서 얻어온 정보를 String으로 변환시켜주고 그 String을 Result로 전달해준다.

05. 프로젝트 구현

-오류 노트



▶ 문법 학습 부분

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="150dp"
    android:layout_marginBottom="1dp"
    android:background="#ffffff"
    android:gravity="center"
    android:orientation="vertical"
    android:textAppearance="?android:attr/textApp

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

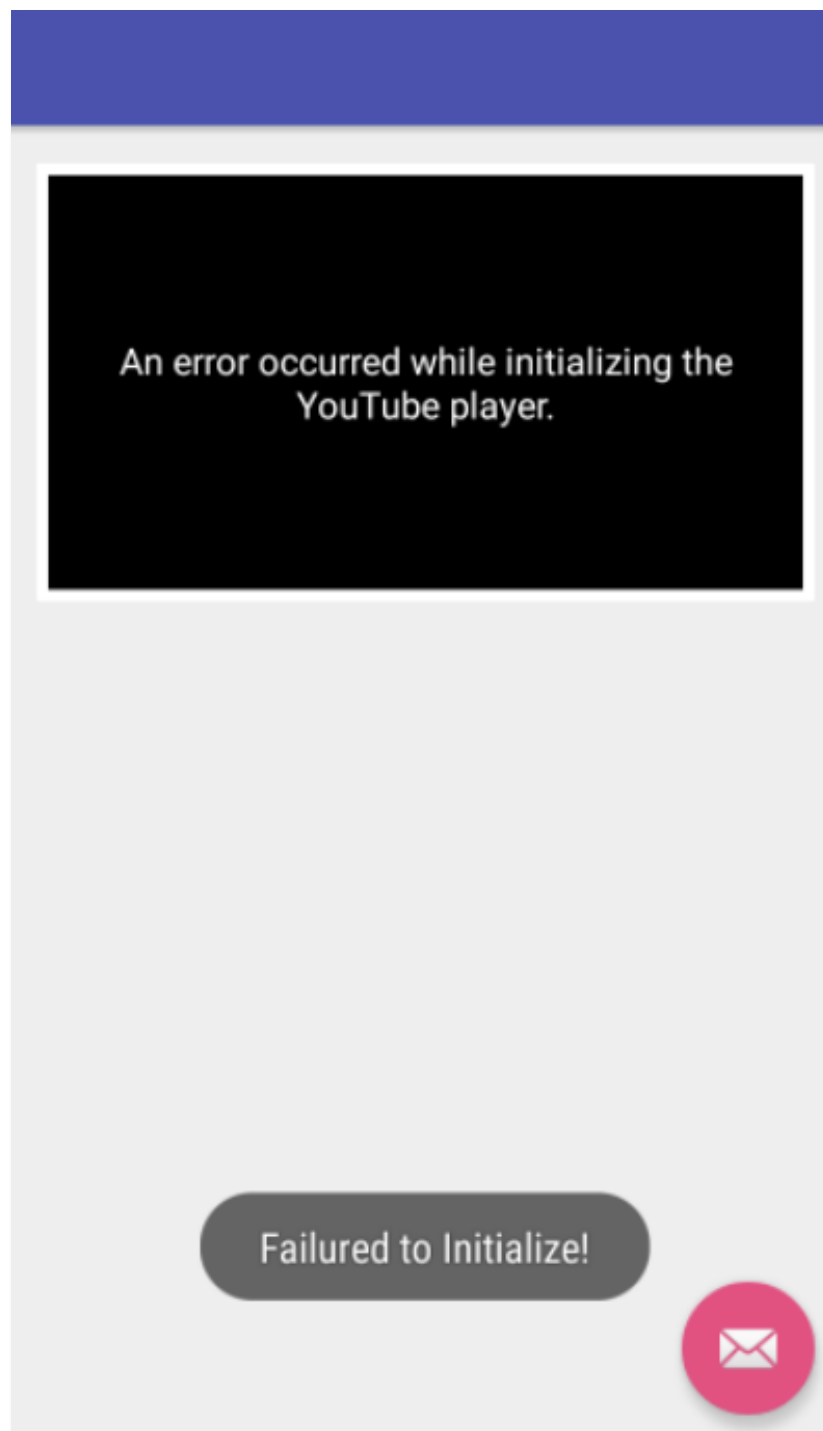
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="The durector has "
    android:textColor="#000000"
    android:textSize="20dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="already"
    android:textColor="#0000ff"
    android:textSize="20dp" />
```

- 텍스트로 강조해주며 색상을 바꿔야 하는데 vertical로 하면 줄바꿈이 되고 Horizontal로 Wn을 사용하면 줄바꿈이 밀려나므로 잠깐 고민이 있었지만 레아웃을 Vertical로 감싸주고, 한번더 Horizontal로 감싸 TextView로 각 단어 색상에 맞게 구현해주었다.

05. 프로젝트 구현

-오류 노트



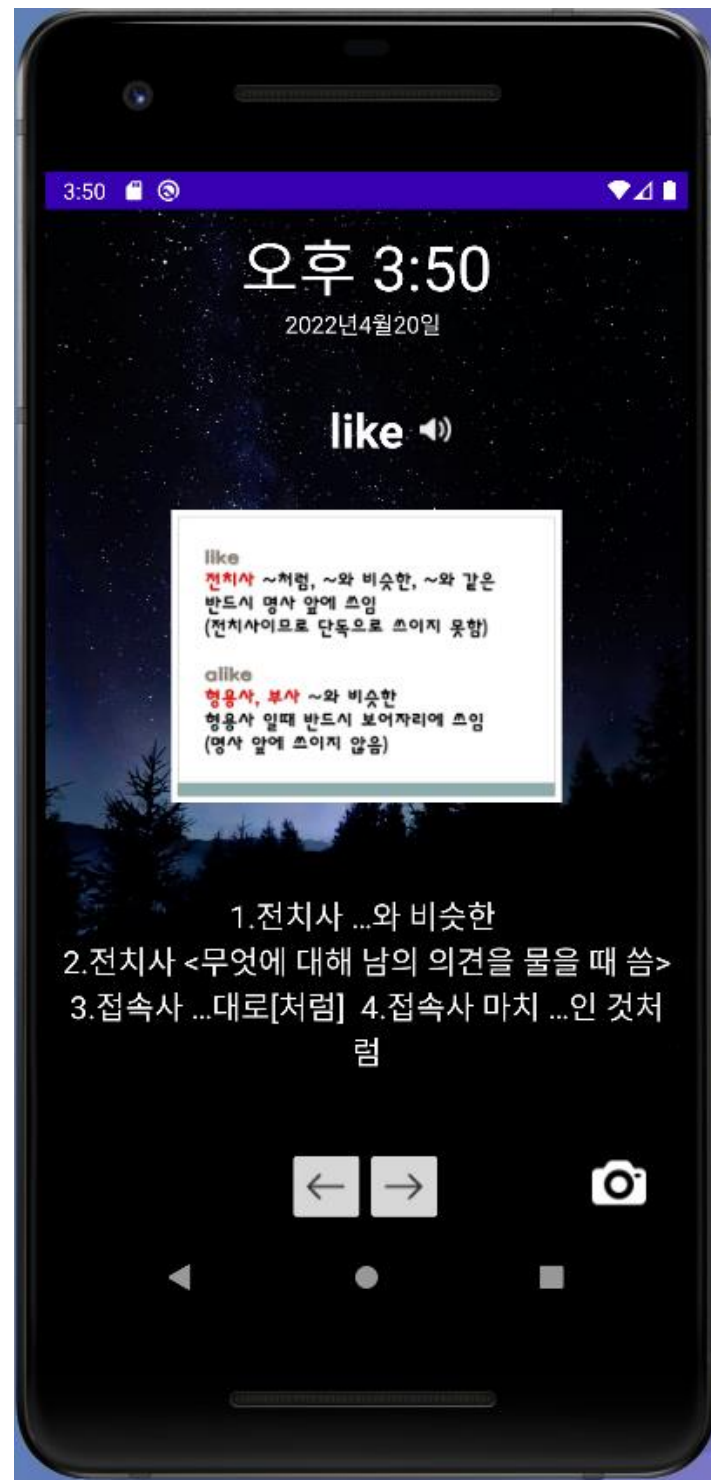
▶ 영상 학습 부분

```
<queries>
  <intent>
    <action android:name="com.google.android.youtube.api.service.START" />
  </intent>
</queries>
```

- 유튜브 플레이어를 재생 시켰을 때 유튜브 플레이어가 초기화되지 않아 오류가 발생했다는 문구가 떴었다.
- 앱에서 Android 11(API 수준 30) 이상을 타겟팅하고 기기에 설치된 다른 앱에 관한 정보를 쿼리하는 경우 시스템은 기본적으로 정보를 필터링하게 되는데 제한된 앱 공개 상태는 다른 앱에 관한 정보를 제공하는 메서드의 반환 결과에 영향을 미치게 되어 제한된 공개 상태는 다른 앱과의 명시적 상호 작용 에도 영향을 준다는 것을 확인했다. 그러므로 다른 패키지를 보려면 <queries> 요소를 사용하여 패키지 공개 상태 확대를 위해 앱 요구사항을 선언해야 하는 것을 확인하여 Manifest에 <queries>를 추가하였다.

05. 프로젝트 구현

-오류 노트

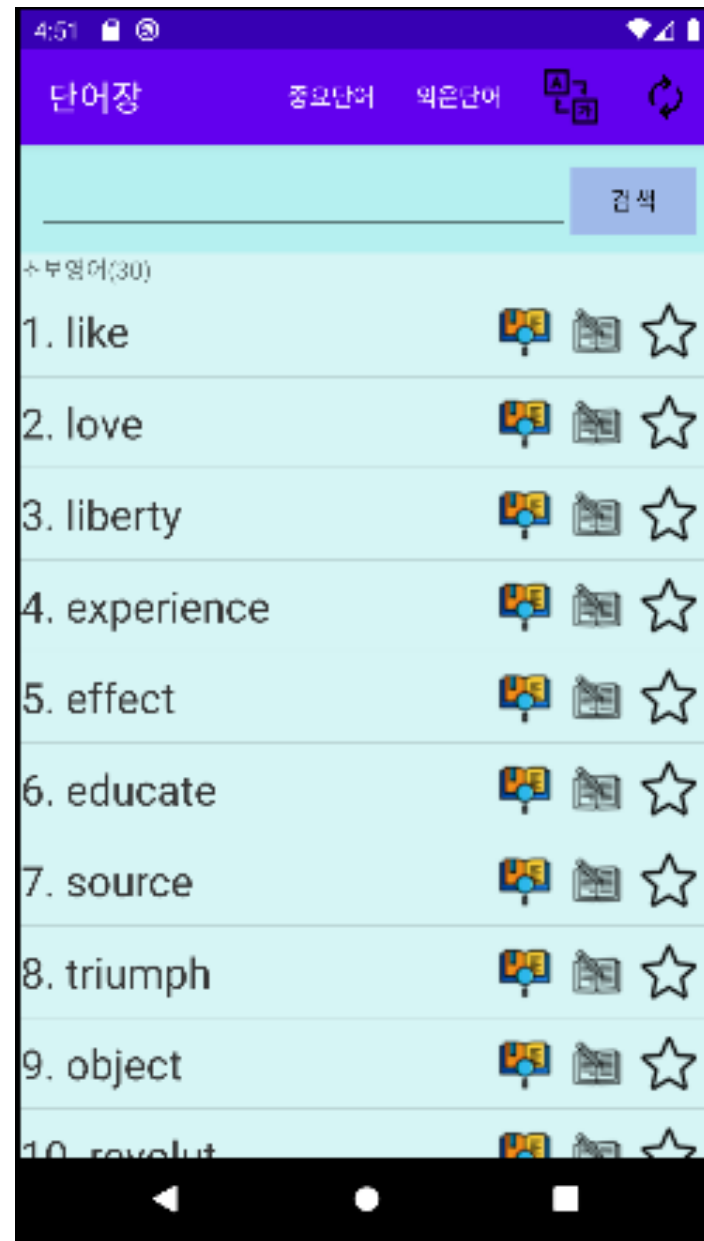


▶ 잠금 화면 구현

- 잠금 화면을 만드는 중 잠금 화면을 실행시킨 후 전원을 여러 번 On Off를 하게 되면 여러 개의 Activity가 계속 켜지는 현상이 있었는데 Manifest에서 `launchMode="singleTop"` 을 넣었더니 잠금 화면인 Activity가 여러 개가 생성되지 않고 한 개의 Activity만 생성되었고 추가적으로 개선한 점은 `AndroidManifest.xml`가 아닌 `ScreenReceiver`에 `i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_SINGLE_TOP);`를 넣어서 새로운 Activity를 만들면서 기존에 존재하는 모든 Activity를 종료하는 코드로 개선하여서 적용함.




05. 프로젝트 구현

-오류 노트



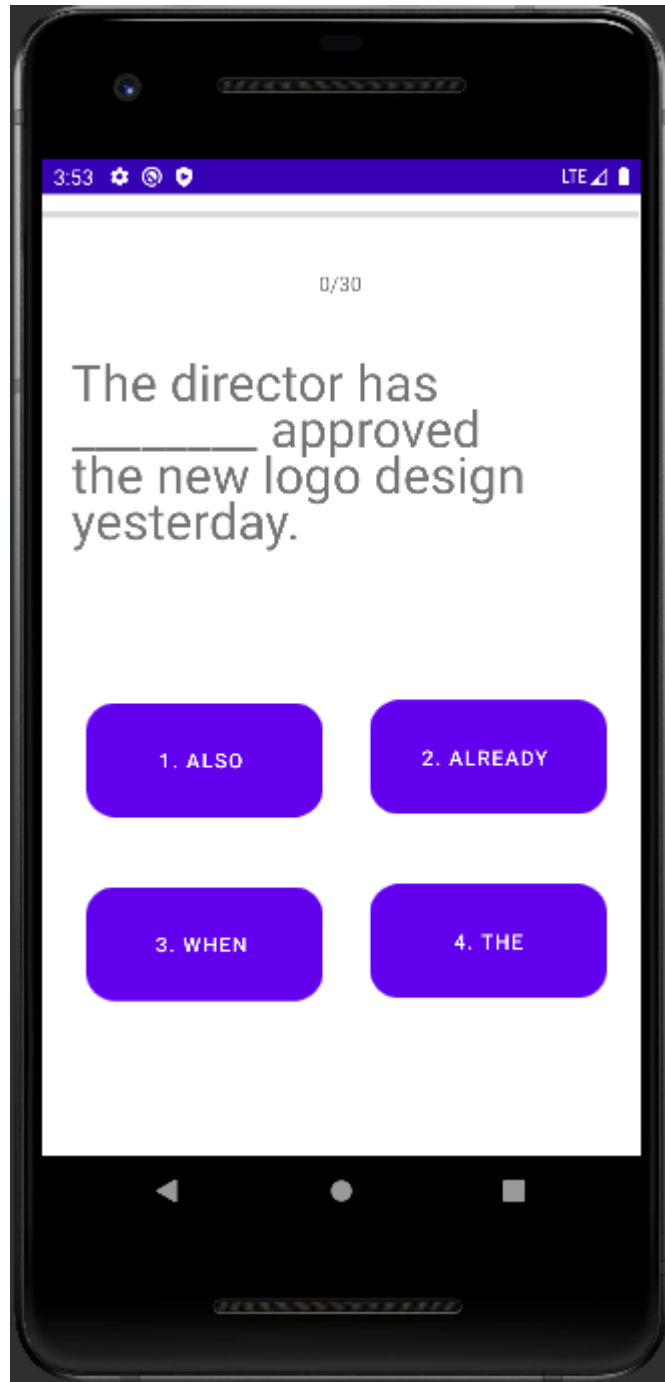
▶ 단어장 화면 구현

```
if(converView == null){
    LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    converView = inflater.inflate(this.getResourceId(*R.layout.listView_btn_item*/, parent, attachToRoot: false);
    holder = new ViewHolder();
    holder.word = (TextView) converView.findViewById(R.id.wordName);
    holder.imageClear = (ImageView) converView.findViewById(R.id.imageClear);
    holder.imageStar = (ImageView) converView.findViewById(R.id.imageStar);
    holder.imageDir = (ImageView) converView.findViewById(R.id.imageDirectory);
    converView.setTag(holder);
} else
    holder = (ViewHolder) converView.getTag();
```

- 리스트 뷰를 활용해서 프로그램을 작성할 때    좌측의 그림들이 이동하는 현상을 경험했습니다 이때 View 홀더를 사용해서 위의 현상을 해결했습니다.

05. 프로젝트 구현

-오류 노트



▶ 퀴즈 화면 구현

```
class QuizSequence implements Runnable { //퀴즈 시퀀스

    Bundle data = new Bundle();
    int problem_Count = 0;

    @Override
    public void run() {
        while (true) {
            SettingQuiz(data);
            if(problem_Count == quizList.size()){
                problem_Count = 0;

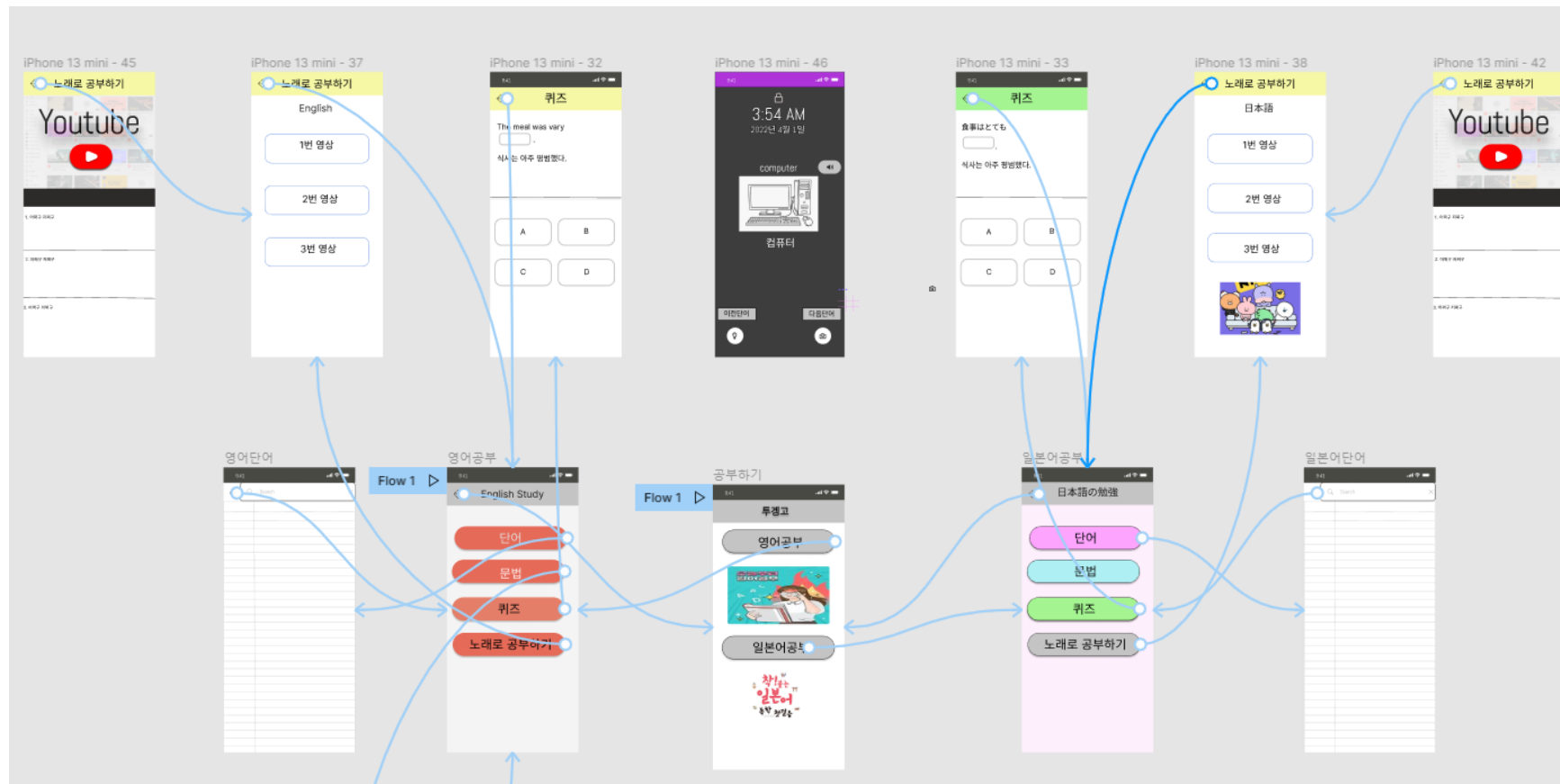
                break;
            }

            try {
                Thread.sleep( millis: 100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

- 퀴즈 시퀀스를 초반에는 while 문안에 settingQuiz()를 넣어서 최적화쪽으로 문제가 있었다.
- 하지만 이 다음에 따로 함수로 빼내어서 빠르게 고쳤다.

05. 프로젝트 구현

-오류 노트



```
word.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        Intent intent = new Intent( packageContext: Btn_eng.this , Dictionary_Activity.class);  
        startActivity(intent);  
    }  
});
```

```
<activity  
    android:name=".Dictionary.Dictionary_Activity"  
    android:label="사전 액티비티" />
```

▶ 프로토타입 구현

- XML을 활용하는데 있어 intent라는 클래스를 활용하게 되었습니다.
- 이 클래스는 activity를 실행하도록 도와주는 기능을 하는데 앱이 강제 종료가 되는 현상이 나와서 곤란하게 되었습니다. 이때 코드를 다시 한 번 살펴보면서 manifests <activity android:name=".class"/>가 빠졌다는 사실을 알게 되었습니다.

06. 프로젝트 후기

강유진

- 처음 접해보는 앱 프로젝트를 하면서 앱에 대한 지식을 쌓을 수 있었고, 예제pdf를 많이 복습하고 활용하여 계획했던 앱 화면을 구현할 수 있었습니다.

우효진

- API 라는 것을 처음 사용하여 약간의 어려움이 있었지만 안드로이드 프로젝트와 제공된 Google API와 연결하여 사용한다는 점이 재미있었다.

강대준

- 한번 구현하고 코드를 보니 더 좋은 코드가 계속 생각났는데 적용시키지 못해 아쉽다, 다음에는 더 좋은 효율의 코드를 빠르게 짤 수 있도록 기초 설계를 잘 해야겠다고 생각했습니다.

박규영

- 처음 접하는 앱에 대해서 많은 것을 알 수 있었고 앱 프로젝트를 하면서 예제를 통해 구현할 수 있는 기능이 많아서 좋았습니다.

06. 프로젝트 후기

김강규

- 팀원들과 아이디어를 공유하며 ‘어떤 기능을 넣고 어떤 자료조사를 하면 좋겠다.’ 라는 얘기를 하였습니다. 프로젝트에 대한 진행이 수월하게 진행됨으로써 팀워크가 얼마나 중요한지 알았습니다.

배광민

- 프로젝트를 진행하며 여러기능을 넣고싶다는 생겼습니다, 설계를 잘못해서 코드를 여러 번 옳으며 많은 것을 배웠습니다. 다들 늦게까지 고생 많으셨습니다 !

지금까지 4조였습니다.

감사합니다!