

코틀린을 사용한 녹음앱 개발

Voice_Recorder

AINC LAB | 소민섭
sobh6498@hs.ac.kr

Voice_Recorder

목차

- 1 안드로이드 개발
 - > 안드로이드 작동원리 간단 설명
- 2 녹음앱 코드 리뷰
 - > MainActivity 주요기능 코드 리뷰
- 3 인터페이스 및 시연
 - > Voice_Recorder 인터페이스
- 4 향후 계획 및 피드백
 - > 향후 계획 및 피드백

01 | 안드로이드 개발

- 안드로이드 작동원리 간단 설명



List에서 chat 실행시 안드로이드 시스템에 의뢰해서 실행하는 구조

안드로이드 주요 컴포넌트 4가지

액티비티(Activity)
사용자가 앱과 상호작용하는 하나의 화면

서비스(Service)
백그라운드에서 어떠한 작업을 처리하기 위해서 사용

브로드 캐스터(Broadcast Receiver)
발생하는 각종 이벤트와 정보를 받아와 핸들링

콘텐츠 제공자(Content Provider)
용량이 큰 데이터들을 공유할 때 사용

02 | 녹음앱 코드 리뷰

- MainActivity 주요기능 코드 리뷰

```
class MainActivity : AppCompatActivity() {

    private var mediaRecorder: MediaRecorder? = null
    private var outputFilePath: String = ""

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val recordButton = findViewById<Button>(R.id.recordButton)
        val stopButton = findViewById<Button>(R.id.stopButton)
        val playButton = findViewById<Button>(R.id.playButton)

        // 녹음 버튼 클릭
        recordButton.setOnClickListener {
            if (checkPermissions()) {
                startRecording()
                recordButton.visibility = Button.GONE
                stopButton.visibility = Button.VISIBLE
            }
        }

        // 정지 버튼 클릭
        stopButton.setOnClickListener {
            stopRecording()
            recordButton.visibility = Button.VISIBLE
            stopButton.visibility = Button.GONE
        }
    }
}
```

1. 클래스 및 변수 선언

MainActivity는 AppCompatActivity를 상속받아 안드로이드의 기본 액티비티 기능을 사용

mediaRecorder는 음성을 녹음하는 데 사용되는 MediaRecorder 객체

outputFilePath는 녹음된 파일의 경로를 저장하는 변수

2. on Create 메서드

onCreate 메서드는 액티비티가 생성될 때 호출
여기서 레이아웃을 설정하고 버튼을 초기화를 한다.

3. 버튼 클릭 이벤트

녹음 버튼: 권한이 있는 경우 녹음을 시작하고
녹음 버튼을 숨기고 정지 버튼을 보이게 함
정지 버튼: 녹음을 정지하고 버튼의 상태를 변경합니다.

02 | 녹음앱 코드 리뷰

- MainActivity 주요기능 코드 리뷰

```
override fun onRequestPermissionsResult(  
    requestCode: Int,  
    permissions: Array<out String>,  
    grantResults: IntArray  
) {  
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)  
    when (requestCode) {  
        200 -> {  
            if (grantResults.isNotEmpty() && grantResults[0] == PackageManager.PERMISSION_GRANTED) {  
                // 권한이 승인  
                startRecording()  
            } else {  
                // 권한 거부  
                showPermissionExplanationDialog()  
            }  
        }  
    }  
}  
  
private fun showPermissionExplanationDialog() {  
    AlertDialog.Builder(context: this)  
        .setTitle("권한 필요")  
        .setMessage("음성 녹음을 위해 마이크 권한이 필요합니다. 설정에서 권한을 허용해주세요.")  
        .setPositiveButton(text: "설정으로 이동") { _, _ -> {  
            val intent = Intent(Settings.ACTION_APPLICATION_DETAILS_SETTINGS)  
            val uri = Uri.fromParts(scheme: "package", packageName, fragment: null)  
            intent.data = uri  
            startActivity(intent)  
        }  
        .setNegativeButton(text: "취소") { dialog, _ -> dialog.dismiss() }  
        .create()  
        .show()  
}
```

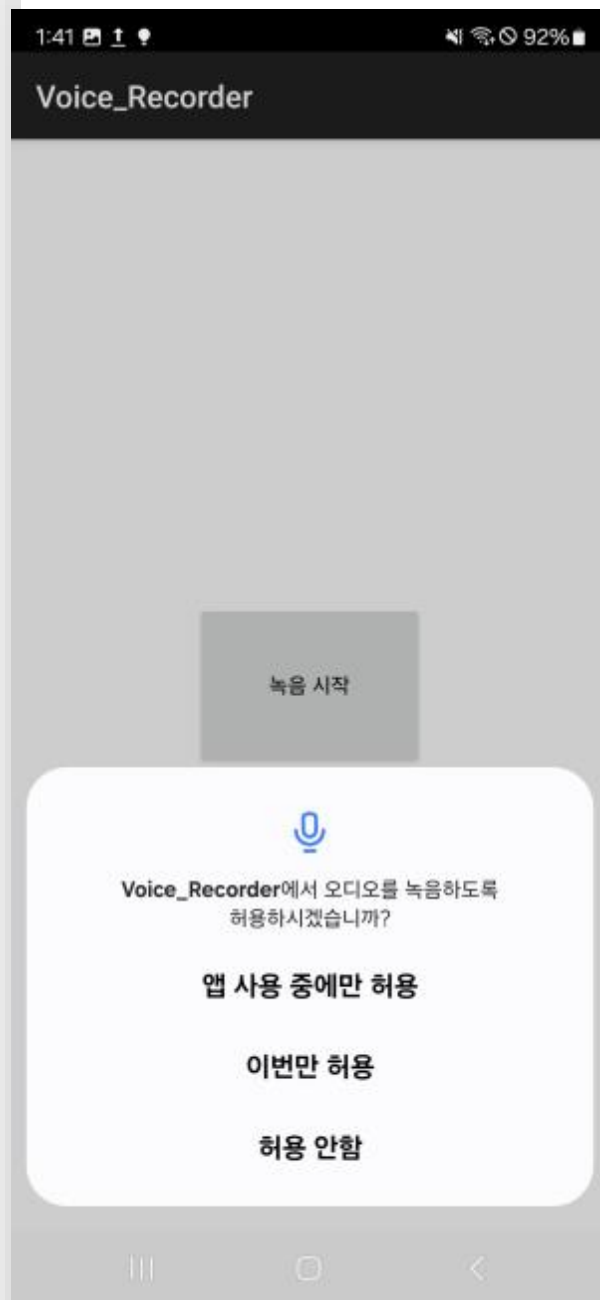
7.권한 요청 결과처리

권한 요청에 대한 결과를 처리합니다. 사용자가 권한을 승인하면 녹음을 시작하고, 거부하면 설명 다이얼로그를 띄웁니다.

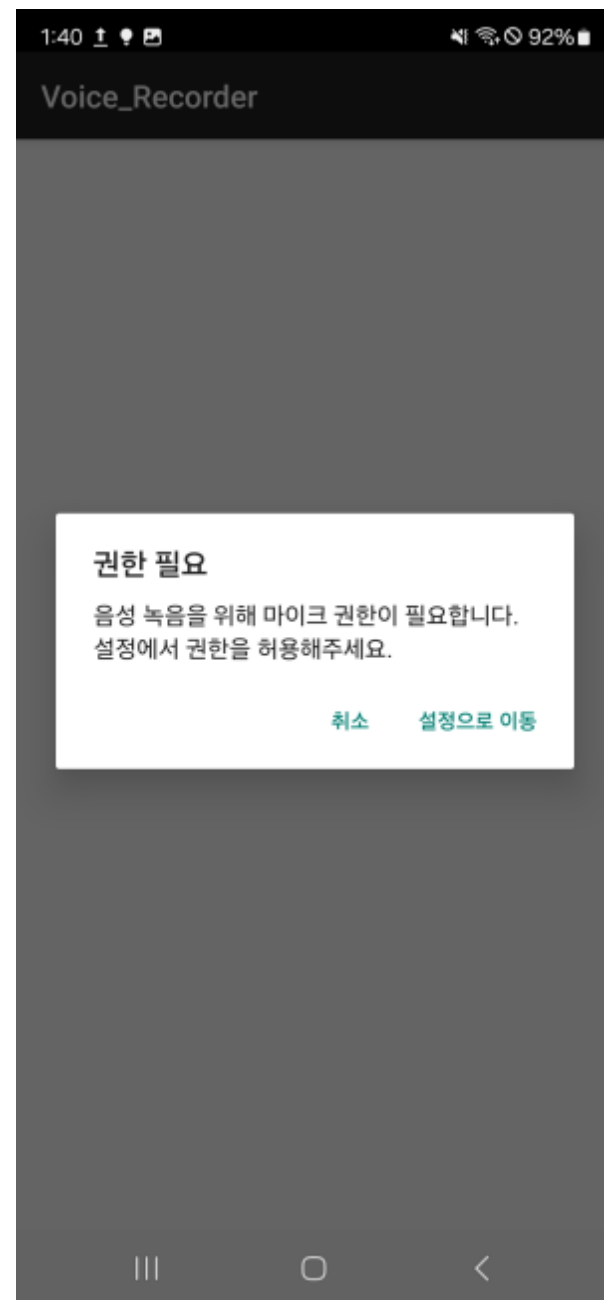
8.권한 설명 다이얼로그

사용자가 권한을 거부했을 때, 권한 필요성을 설명하는 다이얼로그를 보여줍니다. 설정으로 이동할 수 있는 버튼도 제공합니다.

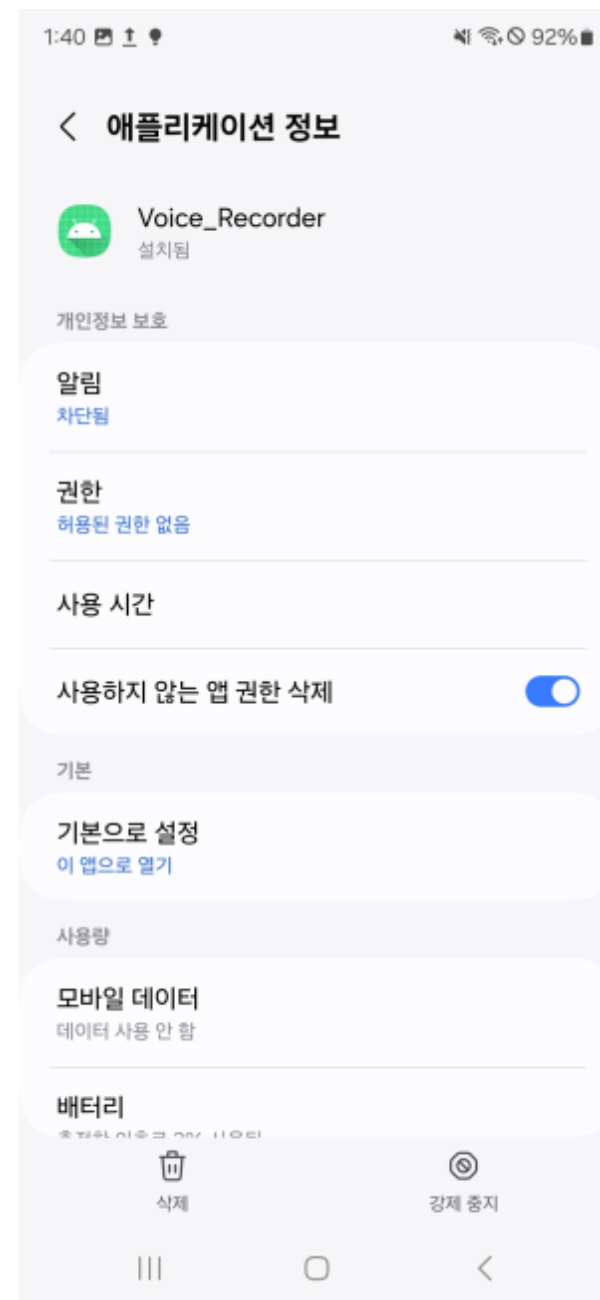
03 | 인터페이스 및 시연



[화면1]



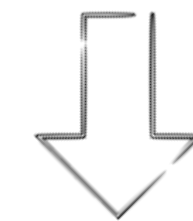
[화면2]



[화면3]

권한이 없는 경우

처음 앱 실행시 오디오 녹음 권한 부여
여부 확인



(권한 허용X)

사용자가 권한을 거부했을 때, 권한
필요성을 설명하는 다이얼로그
및 설정으로 이동할 수 있는 버튼 제공

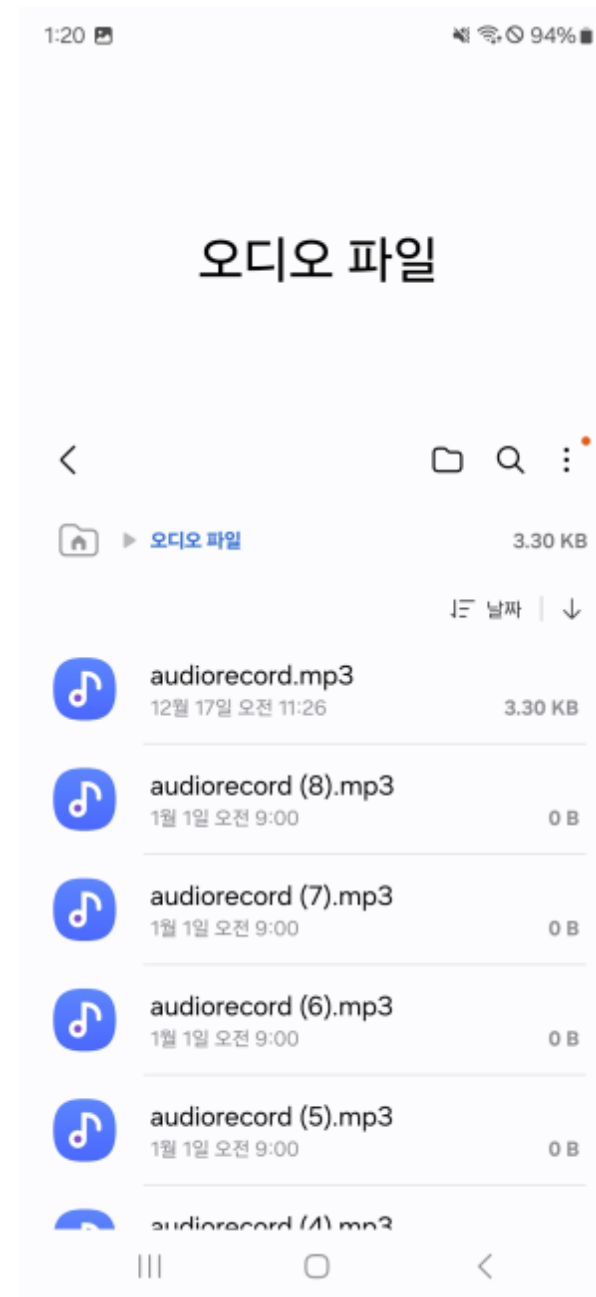
03 | 인터페이스 및 시연



[화면4]



[화면5]



[화면6]

권한이 있는 경우

녹음 시작 버튼 클릭시 파일에 대한 출력
경로 지정후 녹음 시작



녹화 시작 버튼 클릭시
시작 버튼 숨김 및 녹음 버튼 드러냄



녹음 중지 버튼 클릭시
안드로이드 오디오 폴더에 음성 저장
중지 버튼 숨김 및 시작 버튼 드러냄

04 | 향후 계획 및 피드백

향후 계획 및 피드백



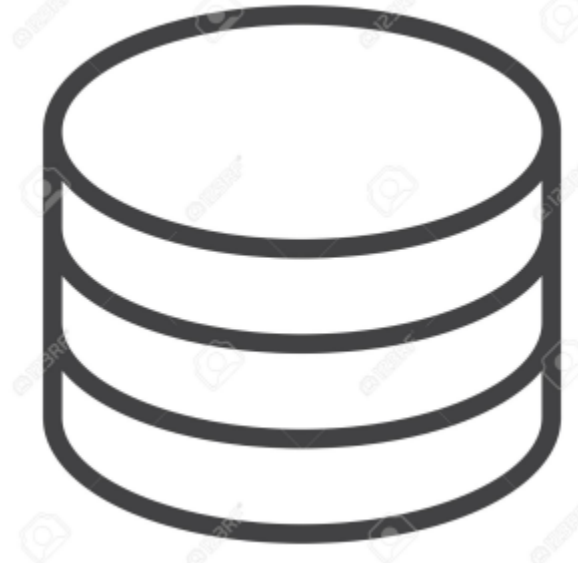
1. UI 개선

- 현재 앱에서는 Status 바 초록불을 제외하고 음성인식이 되고있는지 알수가 없음
음성인식 여부를 확인할 수 있는 UI 생성



2. 일시정지 기능 추가

- 일시정지 기능을 추가하여 중간에 멈췄다가 다시 녹음 할 수 있는 기능 추가



3. 외부저장소 접근 권한 추가

- 녹음 후 번거롭게 저장소를 들어갈 필요 없이 앱에 저장소 접근 권한을 부여받고 재생할 수 있는 기능 추가