TeStar

Team TeStar
2016150019 박재홍
2018158031 장의수

Contents

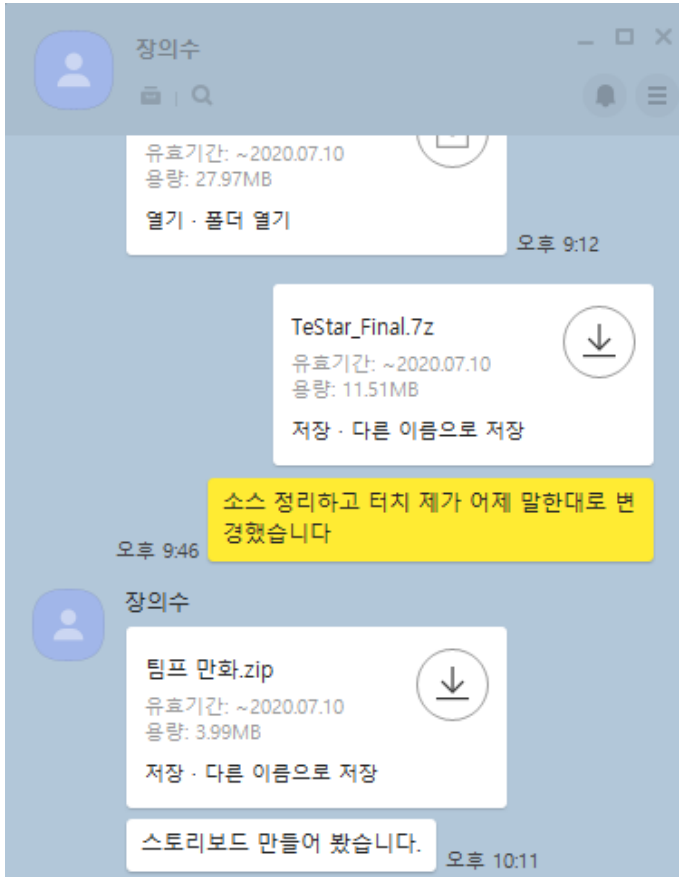
Team TeStar

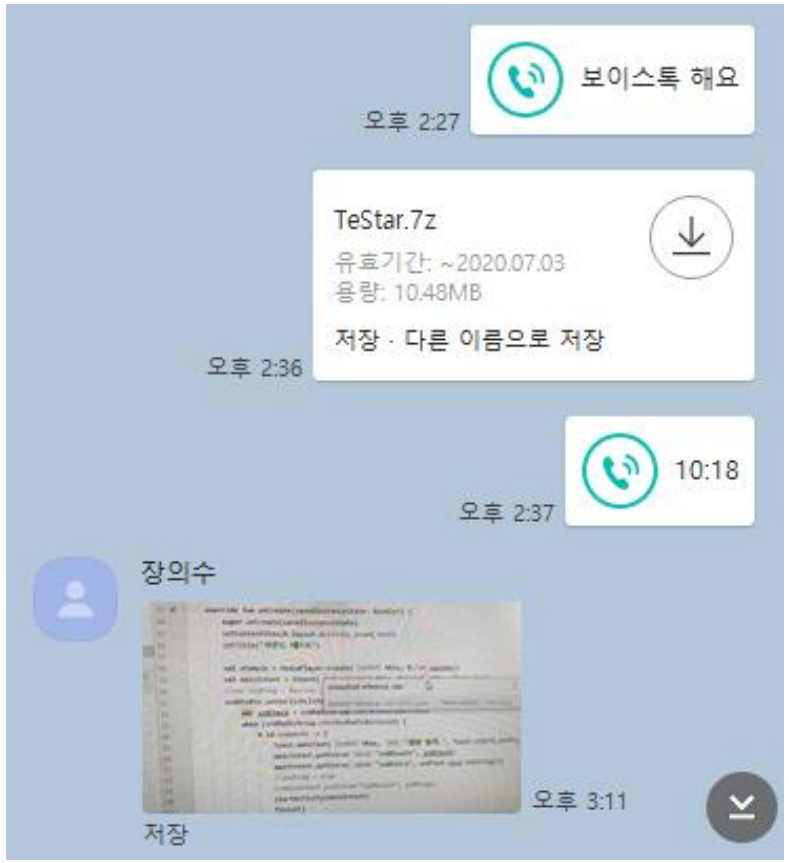
- 박 재 홍
- 역할: 아이디어 구상 및 인텐트 분담 구현(메인, 사운드, 카메라, 네트워크)

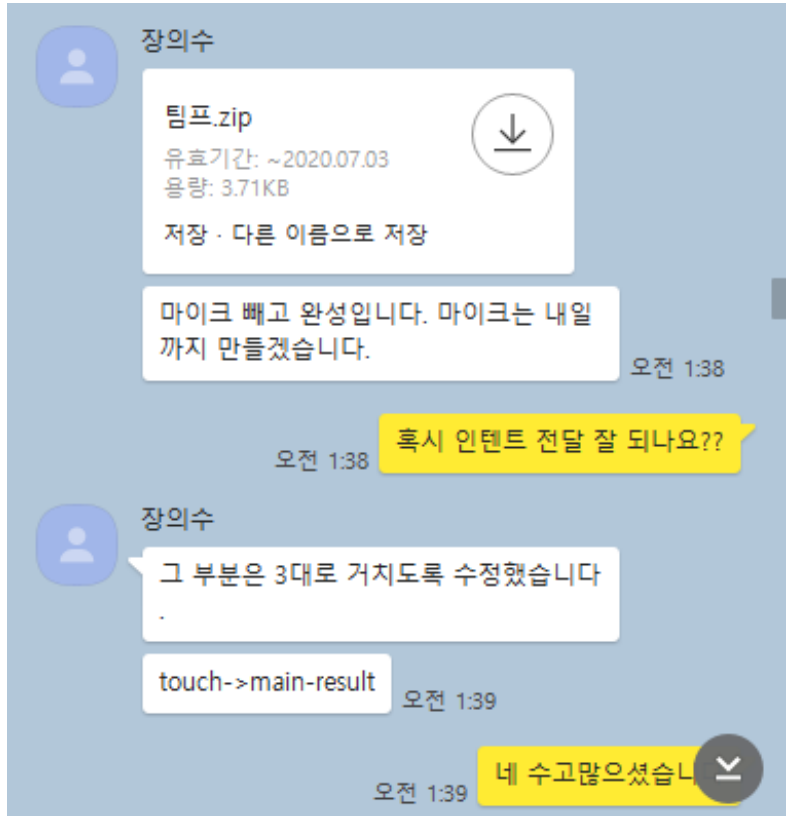
- 장 의 수
- 역할: 인텐트 분담 구현(터치, 마이크, 결과) 및 테스트

활동 내용

카카오톡을 적극적으로 활용하여
팀 프로젝트 진행







TeStar

- 휴대폰 종합 테스트 앱
- 문제 발생(의심) 시 개인의 자가 진단 및 A/S 문의 등 고객지원, 휴대폰 중고 거래 등의 경우까지 유용하게 이용 가능
- 현재 사운드(스피커), 네트워크, 마이크, 카메라, 터치 동작 여부까지 테스트 가능

앱 개발 배경

- 개인적 사유

➢고장 난 스피커

→ 아이디어의 전환

- 앱이 만들어진다면?

→ 다양한 곳에 사용될 수 있을 것(자가 진단, 중고 거래, A/S시…)

앱 기능 소개

- 수업에서 배운 기능
- 다양한 UI 관련 기능들
- 이벤트 처리
- 인텐트 데이터 전달
- 권한 설정

- 새로 공부한 기능
- 소리 재생
- 소리 녹음(마이크 이용)
- 카메라 이용
- 풀스크린 액티비티(Fullscreen Activity) 이용
- 파일 다루기(아주 약간)
- 아이콘 만들기

공부한 기능

- 소리 녹음
- 파일 다루기

```
private var output: String? = null
private var mediaRecorder: MediaRecorder? = null
private var state: Boolean = false
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_mic_test)
    val resultIntent = Intent( packageContext: this, ResultActivity::class.java)
    if (ContextCompat.checkSelfPermission( context: this,
            Manifest.permission.RECORD_AUDIO) != PackageManager.PERMISSION_GRANTED && ContextCompat.checkSelfPermission( context: this,
            Manifest.permission.WRITE_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {
        val permissions : Array<String> = arrayOf(android.Manifest.permission.RECORD_AUDIO, android.Manifest.permission.WRITE_EXTERNAL_STORAGE, android.Manifest.permission.READ_EXTERNAL_STORAGE)
        ActivityCompat.requestPermissions( activity: this, permissions, requestCode: 0)
    }
    output = Environment.getExternalStorageDirectory().absolutePath + "/recording.mp3"
    mediaRecorder = MediaRecorder()

    mediaRecorder?.setAudioSource(MediaRecorder.AudioSource.MIC)
    mediaRecorder?.setOutputFormat(MediaRecorder.OutputFormat.MPEG_4)
    mediaRecorder?.setAudioEncoder(MediaRecorder.AudioEncoder.AAC)
    mediaRecorder?.setOutputFile(output)
    start.setOnClickListener { it: View!
        if (ContextCompat.checkSelfPermission( context: this,
                Manifest.permission.RECORD_AUDIO) != PackageManager.PERMISSION_GRANTED && ContextCompat.checkSelfPermission( context: this,
                Manifest.permission.WRITE_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {
            val permissions : Array<String> = arrayOf(android.Manifest.permission.RECORD_AUDIO, android.Manifest.permission.WRITE_EXTERNAL_STORAGE, android.Manifest.permission.READ_EXTERNAL_STORAGE)
            ActivityCompat.requestPermissions( activity: this, permissions, requestCode: 0)
        } else {
            startRecording()
        }
```

공부한 기능(cont.)

- 소리 재생

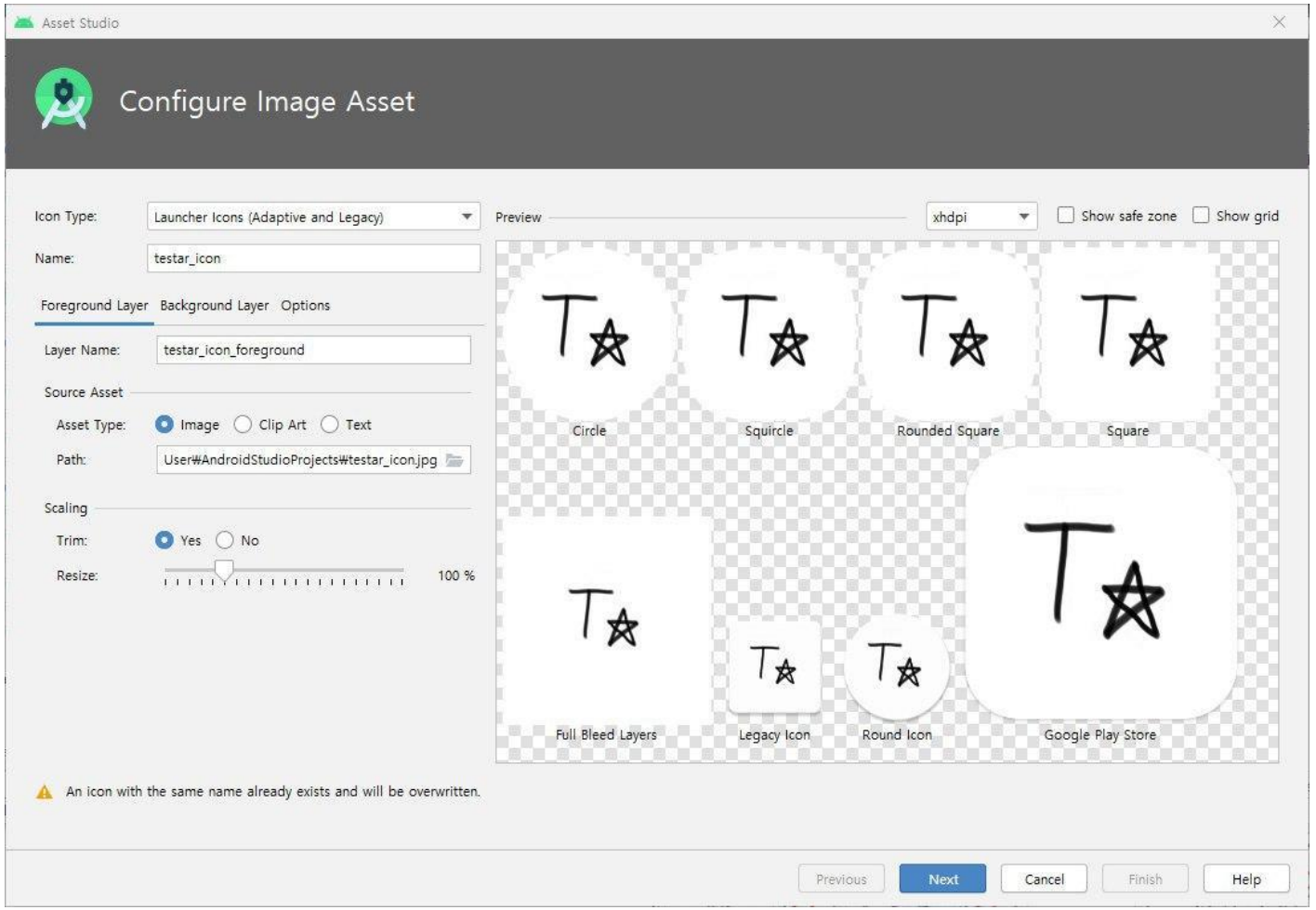
```
val aSample : MediaPlayer! = MediaPlayer.create( context: this, R.raw.speaker)
val resultIntent = Intent( packageContext: this, ResultActivity::class.java)
sndRtnBtn.setOnClickListener() { it: View!
    var sndCheck : Int = sndRadioGroup.checkedRadioButtonId
    when (sndCheck) {
        R.id.sndWorks -> {
            resultIntent.putExtra( name: "kind", value: "sound")
            resultIntent.putExtra( name: "sndResult", value: true)
            resultIntent.putExtra( name: "sndExtra", sndText.text.toString())
            startActivity(resultIntent)
            finish()
        }
        R.id.sndDont -> {
            resultIntent.putExtra( name: "kind", value: "sound")
            resultIntent.putExtra( name: "sndResult", value: false)
            resultIntent.putExtra( name: "sndExtra", sndText.text.toString())
            startActivity(resultIntent)
            finish()
        }
        -1 -> Toast.makeText( context: this, text: "아무 것도 선택하지 않았습니다.", Toast.LENGTH_SHORT).show()
    }
}
playBtn.setOnClickListener() { it: View!
    aSample.start()
}
```

공부한 기능(cont.)

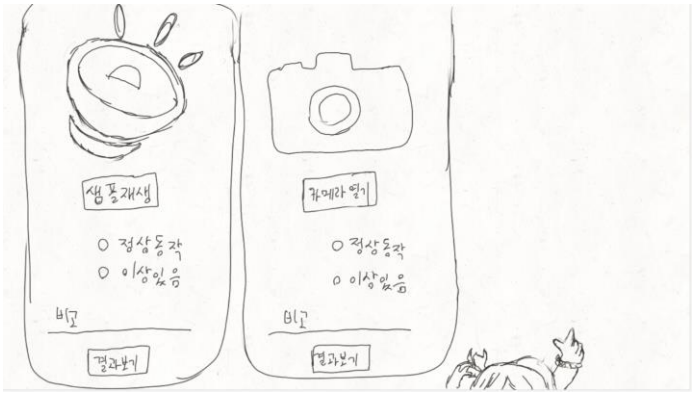
- 카메라 이용

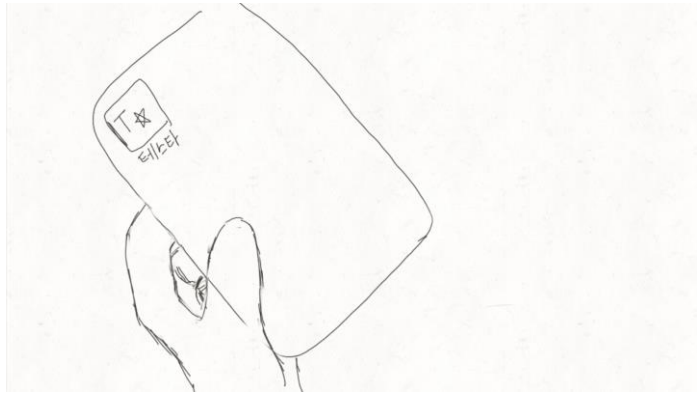
```
private val PERMISSION_CODE = 1000;
private val IMAGE_CAPTURE_CODE = 1001
var image_uri: Uri? = null
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)

    setTitle("카메라 테스트")
    setContentView(R.layout.activity_camera_test)

    val resultIntent = Intent( packageContext: this, ResultActivity::class.java)

        camBtn.setOnClickListener() { it: View!
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
                if (checkSelfPermission(Manifest.permission.CAMERA)
                    == PackageManager.PERMISSION_DENIED ||
                    checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE)
                    == PackageManager.PERMISSION_DENIED){
                    //permission was not enabled
                    val permission : Array<String> = arrayOf(Manifest.permission.CAMERA, Manifest.permission.WRITE_EXTERNAL_STORAGE)
                    //show popup to request permission
                    requestPermissions(permission, PERMISSION_CODE)
                }
                else{
                    //permission already granted
                    openCamera()
                }
            }
            else{
                //system os is < marshmallow
                openCamera()
            }
```

공부한 기능(cont.)

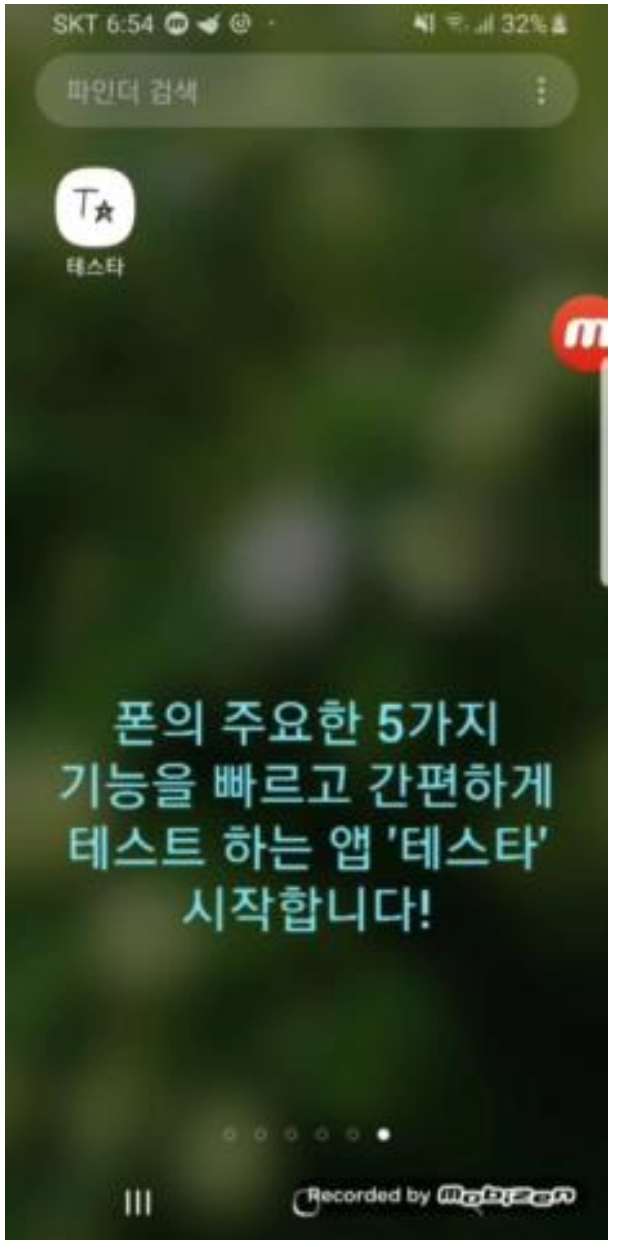

Asset Studio
Configure Image Asset
Icon Type:
Launcher Icons (Adaptive and Legacy)
Name:
testar_icon
Foreground Layer
Background Layer
Options
Layer Name:
testar_icon_foreground
Source Asset
Asset Type:
Image
Clip Art
Text
Path:
User#AndroidStudioProjects#testar_icon.jpg
Scaling
Trim:
Yes
No
Resize:
100 %
Preview
xhdpi
Show safe zone
Show grid
Circle
Squircle
Rounded Square
Square
Full Bleed Layers
Legacy Icon
Round Icon
Google Play Store
An icon with the same name already exists and will be overwritten.
Previous
Next
Cancel
Finish
Help

사용 시나리오 스토리보드





시연 동영상


SKT 6:54
32%
파인더 검색
테스타
폰의 주요한 5가지
기능을 빠르고 간편하게
테스트 하는 앱 '테스타'
시작합니다!
Recorded by

소감

- 박재홍: 이번 팀프로젝트를 진행하면서 처음으로 휴대폰에 앱을 설치하여 테스트해봤는데 처음 구현해 본 센서 동작이 실제 플랫폼에서 동작하는 것을 보니 흥미로웠고, 프로젝트 자체도 수월하게 진행되어 과정도 만족스러웠다. 새로운 기능도 이용해 보고, 공부가 많이 되었다.

- 장의수: 센서를 다루는 부분이라는 점에서 처음 보는 함수가 많았고, 따라서 warning이 한번 나면 어느 부분이 문제인지 한참을 헤매야 했다. 그러나 분명한 것은, 그 많은 시간이 헛되지 않을 만큼 유용한 기능을 배웠다는 점이다.

감사합니다