

음성인식 유튜브 영상 조절 애플리케이션

# 소프트웨어 상세 설계서 (Software Design Description)

2020년 4월 22일

인하대학교 컴퓨터공학과  
캡스톤설계 - 001반

교 수 신병석  
12163588 김예담  
12130397 박찬준  
12152984 전은호

# 목 차 (Table of Contents)

<b>목 차 (Table of Contents)</b>	<b>2</b>
<b>1. 서론 (Introduction)</b>	<b>3</b>
1.1 목적 (Purpose)	3
1.2 범위 (Scope)	3
1.3 개요 (Overview)	3
1.4 참고 자료 (Reference Material)	3
<b>2. 시스템 개요 (System Overview)</b>	<b>4</b>
<b>3. 시스템 아키텍처 (System Architecture)</b>	<b>5</b>
3.1 Architectural Design	5
3.2 Decomposition Description	6
3.2.1 음성인식 모듈	6
<b>4. 데이터 디자인 (Data Design)</b>	<b>8</b>
4.1 Data Description	8
<b>5. 컴포넌트 디자인 (Component Design)</b>	<b>10</b>
<b>6. Human Interface Design</b>	<b>12</b>
6.1 사용자 인터페이스 개요 (Overview of User Interface)	12
6.2 Screen Images	13
<b>7. 프로젝트 관리 (Milestone)</b>	<b>18</b>

# 1. 서론 (Introduction)

## 1.1 목적 (Purpose)

본 문서는 음성인식 유튜브 영상 조절 애플리케이션 ‘쿠키야! - 핸드프리 비디오 플레이어’(이후 쿠키야!)을 위한 설계서이다. 인하대학교 컴퓨터공학과 ‘컴퓨터공학 종합설계’ 001분반 ‘예찬호’ 팀의 프로젝트 설계 및 구현을 위한 것으로 시스템의 구조를 나타낸다.

본 문서는 인하대학교 ‘컴퓨터공학 종합설계’ 001분반 ‘예찬호’ 팀을 주요 독자로 한다. 부가적으로 담당 교수 신병석 교수님이 주요 독자가 될 수 있다. 추후 본 시스템을 상품으로 개발할 경우, 이와 관련된 모든 업체 직원들이 추가적인 독자가 될 수 있다.

## 1.2 범위 (Scope)

본 프로젝트는 음성을 통해서 유튜브 영상을 컨트롤하는 새로운 애플리케이션을 개발하는 것이다. 본 프로젝트의 개발 범위는 다음과 같다.

1. 특정 단어가 발화되었는지 인식할 수 있는 Keyword Spotting 기능
2. 음성을 통해서 ‘쿠키야!’ 애플리케이션을 동작하는 기능
3. 음성을 통해 컨트롤할 유튜브 영상을 애플리케이션 내부로 저장하는 기능
4. 음성을 통해서 영상의 특정 시간대에 북마크를 설정하고, 특정 북마크로 이동하는 기능
5. 음성을 통해서 영상을 멈추고, 다시 재생하는 기능
6. 음성을 통해서 영상의 플레이 시점을 조절하는 기능

## 1.3 개요 (Overview)

본 문서의 구성은 다음과 같다. 2-3장에서는 ‘쿠키야!’ 애플리케이션의 전반적인 시스템 개요와 구조를 서술하고 4장에서는 데이터스키마, 5장에서는 3장을 부연한다. 6장에서는 외부 인터페이스의 구성을 서술한다.

## 1.4 참고 자료 (Reference Material)

- 안드로이드 프로그래밍 정복1(김상형, 한빛미디어)
- <https://developer.android.com/docs> (안드로이드 개발 문서)
- <https://blog.fossasia.org/hotword-detection-in-susi-android-app-using-snowboy/> (SNOWBOY 사용 예제)
- <http://docs.kitt.ai/snowboy/> (SNOWBOY 개발 문서)

## 2. 시스템 개요 (System Overview)

음성인식 유튜브 영상 조절 애플리케이션 ‘쿠키야!’는 음성을 통해서 유튜브 영상을 조절할 수 있는 시스템이다. 사용자는 요리를 하고 있을 때, 또는 그림을 그리고 있을 때와 같이 양손을 사용할 수 없는 상황에서 유튜브 콘텐츠를 시청하는 중간에 영상의 특정 시점으로 이동과 같은 동작을 음성으로 명령할 수 있다.

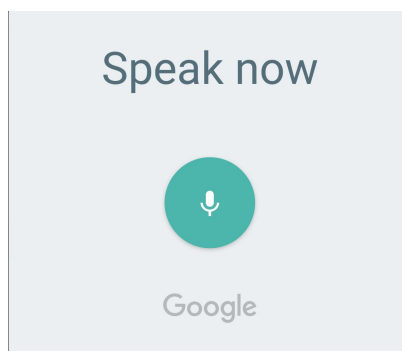
이 시스템은 안드로이드 운영체제 위에서 동작하는 안드로이드 애플리케이션이다. 안드로이드 애플리케이션은 다음 4가지 주요 구성요소로 이루어진다.

- **액티비티** : 사용자 인터페이스를 구성하는 기본단위.
- **서비스** : UI가 없어 사용자 눈에 직접적으로 보이지 않으며 백그라운드에서 무한히 실행되는 컴포넌트
- **Broadcast Receiver** : 시스템으로부터 전달되는 방송을 대기하고 신호 전달 시 수신하는 역할
- **Content Provider** : 다른 응용 프로그램을 위해 자신의 데이터를 제공

‘쿠키야!’ 앱의 주요 구성 성분은 **UI를 이루는 액티비티들과, Keyword Spotting 기능과 음성인식 기능을 수행하는 서비스**이다.

안드로이드 운영체제에서 실행되는 앱의 4가지 구성요소간 또는 다른 앱들간에는 **인텐트(Intent)**라는 객체를 통해서 상호작용할 수 있다. 안드로이드 운영체제는 앱의 구성요소들을 내부에 등록하여 관리하는데, 이 기능을 수행하는 것이 액티비티 매니저이다. 이 때 액티비티 매니저에게 다른 액티비티 실행과 같은 명령을 요청할 수 있는 포맷이 인텐트이다. 인텐트 객체에 원하는 동작을 설정하여 액티비티 매니저에게 전달하면, 안드로이드 운영체제가 해당 명령을 수행하게 된다.

음성인식 유튜브 영상 조절 애플리케이션 ‘쿠키야!’에 가장 핵심이 되는 음성인식기능과 Keyword Spotting 기능은 **구글의 STT 서비스와 KITT.AI사의 SNOWBOY Hotword Detection 엔진**을 이용한다.



**구글 STT 서비스**는 머신러닝 기반 음성 텍스트 변환기술이다. 구글 STT 서비스를 지원하는 안드로이드 단말기를 타겟으로 개발하는 경우 별도의 구성 없이 안드로이드 SDK 에 포함되어 있는 android.speech 패키지 구성 api들로 음성인식 기능을 구현할 수 있다. 본 시스템 역시 구글 STT 서비스를 지원하는 안드로이드 단말기를 타겟으로 개발을 진행한다.

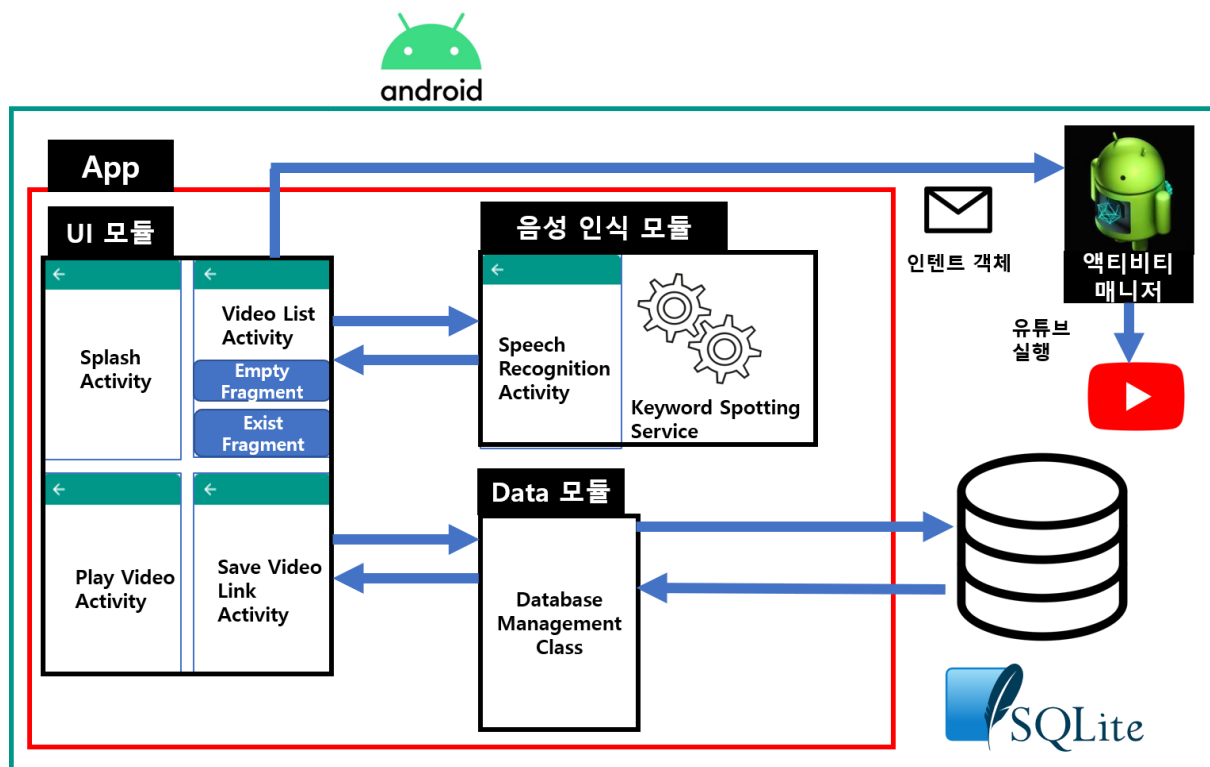


**KITT.AI사의 SNOWBOY Hotword Detection 엔진**은 커스터마이징 된 Hotword (Wake-up word)를 인식하는 api이다. ARMv7 계열의 CPU를 가지고 있는 android 기기를

지원한다. 최초에 커스터마이징 된 Hotword를 개발자 또는 사용자가 3번 녹음하여 snowboy 서버에 전송하면, Hotword를 인식할 수 있는 Deep neural network를 훈련시키고, 해당 Hotword 인식과 관련된 .mdl 형식의 파일을 내려받아 안드로이드에 설치된 Hotword detection engine에서 사용한다. 안드로이드 개발자는 snowboy 엔진을 앱 내부에 포함시키고, snowboy에서 제공하는 api를 통해 Hotword detection 기능을 구현할 수 있다.

### 3. 시스템 아키텍처 (System Architecture)

#### 3.1 Architectural Design



위 그림은 우리가 구현할 음성인식 애플리케이션의 구조를 나타낸다. 위 그림에서 본 애플리케이션을 나타내는 부분은 빨간색으로 구분된 영역이다. 본 애플리케이션은 **UI 모듈**, **음성 인식 모듈**, **Data 모듈**의 3가지 부분으로 나누어져있다.

**UI 모듈**은 크게 4가지 액티비티로 구성되어 있다. 각각의 액티비티는 초기화 화면(Splash Activity), 영상 리스트 화면(Video List Activity), 영상 재생화면(Play Video Activity), 유튜브 링크 공유 화면(Save Video Link Activity) 등 사용자에게 보여지는 화면을 구성한다. 또한 화면 내부에 존재하는 위젯(버튼, 동영상 화면 등)을 조작하는 기능을 수행할 수 있다.

**Data 모듈**은 UI 모듈에서 필요로 하는 데이터를 안드로이드 기기 내부의 SQLite 데이터베이스에서 가져오거나, 데이터를 저장하는 등의 역할을 한다. Data 모듈은 데이터를

관리하는 메소드를 가진 클래스 형태로 구현될 수 있으며, UI 모듈에서 이 메소드를 호출함으로써 상호작용할 수 있다.

**음성 인식 모듈**은 사용자가 Wake-up word를 발화하는지를 체크하고, Keyword spotting이 일어난 경우 사용자가 발화하는 문장을 인식하여 이에 적절한 동작을 수행하는 모듈이다. 이 모듈의 구성 중 하나인 **Keyword Spotting Service**는 백그라운드에서 계속 동작하여 Wake-up word가 발화되었는지를 체크한다. 다른 또 하나의 구성원은 **Speech Recognition Activity**로 Keyword Spotting이 일어난 경우 사용자에게 음성으로 명령어를 입력할 것을 제시하는 프롬프트를 띄우고, 사용자가 발화한 내용을 화면에 보여주며, 음성인식 기능을 수행하는 역할을 한다.

음성 인식 모듈과 UI 모듈은 서로 상호작용할 필요가 있다. 음성 인식 모듈에서 인식한 사용자의 명령에 따라 UI 모듈의 화면을 갱신하는 경우 등이 그 예이다. 이 두 모듈간에는 **인텐트**라는 객체를 통해서 서로 상호작용할 수 있다.

UI 모듈에서 Video List Activity는 안드로이드 시스템 내부에 존재하는 액티비티 매니저에게 인텐트 객체를 전송하여 유튜브 애플리케이션이 실행되게 할 수 있다.

## 3.2 Decomposition Description

이 절에서는 3.1절에서 설명한 각 모듈의 구성요소 중에서 가장 핵심적인 부분인 음성인식 모듈에 관해서 설명한다.

### 3.2.1 음성인식 모듈

RecordingThread
-detector : SnowboyDetect -handler : Handler -thread : Thread
+RecordingThread(handler : Handler, listener : AudioDataReceivedListener) +sendMessage(what : MsgEnum, obj : Object) : void +startRecording() : void +stopRecording() : void +record() : void

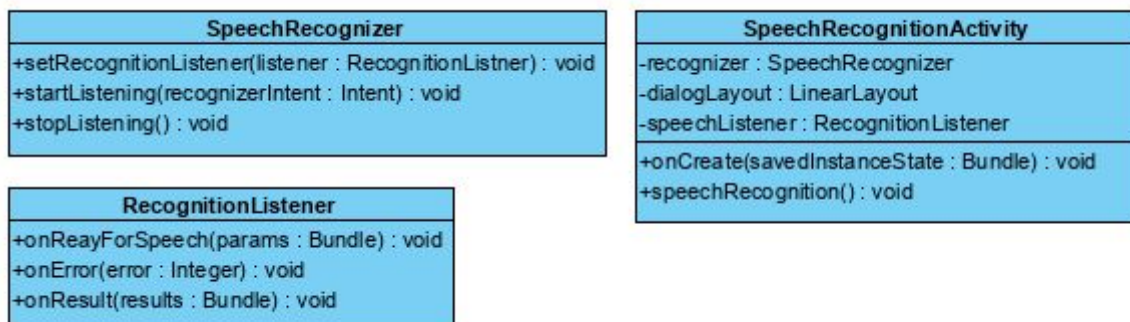
  

KeywordSpottingService
-recordingThread : RecordingThread -isDetectionOn : Boolean -handler : WakeUpHandler
+onCreate() : void +onDestroy() : void +onStartCommand(intent : Intent, flags : Integer, startId : Integer) : Integer +initKeywordSpotting() : void

먼저 KeywordSpottingService 클래스, 그리고 그 클래스의 멤버변수인 recordingThread의 클래스 정의를 살펴본다.

- **KeywordSpottingService** : 애플리케이션이 실행되는 순간에 백그라운드에서 계속적으로 “쿠키야” 라는 Wake-up word의 발화를 체크한다.
- **onCreate()** : Service 객체 생성시 초기화를 담당하는 콜백 메소드
- **onStartCommand()** : 내부적으로 Keyword Spotting을 시작하는 콜백 메소드
- **initKeywordSpotting()** : onCreate() 내부에서 실행되며 사용자로부터 오디오 사용권에 대한 퍼미션을 확인하고, RecordingThread 객체를 만들어 recordThread 멤버를 초기화한다.
- **RecordingThread** : Keyword Spotting 관련 opensource 라이브러리인 snowboy에서 제공하는 Thread 클래스이다. 기기의 마이크에서 들려오는 사운드를 인식하여 Wake-up word가 발화되었는지 체크하는 메소드들을 가지고 있다.

KeywordSpottingService 클래스의 recordingThread 멤버를 초기화할때 Myhandler 객체를 넘겨주는데, Myhandler 클래스는 handleMessage라는 콜백 메소드를 가지고 있다. recordingThread에서 전달받은 Myhandler 객체의 sendMessage 메소드를 호출하면 자동으로 handleMessage 메소드가 호출되며 sendMessage에서의 반환값을 넘겨줄 수 있다. 이를 통해서 thread 작업의 결과물을 전달할 수 있다. 여기서는 키워드가 발화되었는지 아닌지를 넘겨준다.



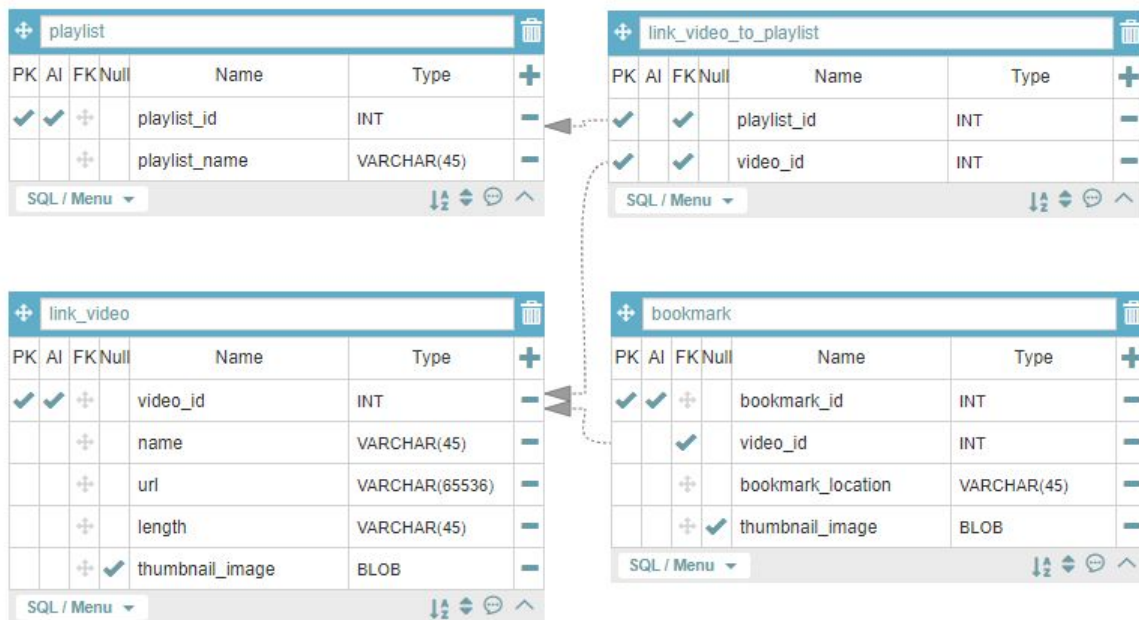
다음으로 SpeechRecognitionActivity 클래스와 관련 클래스 정의를 살펴본다.

- **SpeechRecognitionActivity** : KeywordSpotting이 일어난 후에 사용자로부터 음성 입력을 받기 위해서 보여주는 프롬프트 화면과 관련된 클래스이다.
- **speechRecognition()** : 음성인식을 수행하는 메소드이다. RecognitionListener 인터페이스 객체를 recognizer 멤버변수에 등록하고 해당 객체의 startListening() 메소드를 호출하여 음성인식을 시작한다.
- **SpeechRecognizer** : 구글의 android api 중 음성인식 관련 클래스이다.
- **startListening()** : 음성인식 기능을 시작한다. 오디오로 들려오는 음성을 구글 stt 서버로 보내고, 음성인식 결과를 받아온다.
- **RecognitionListener** : SpeechRecognizer에서 음성인식 완료 후 자동으로 수행하는 콜백 메소드들에 대한 인터페이스이다. SpeechRecognitionActivity에서 해당 인터페이스 구현한 뒤 객체를 생성하여 recognizer에 setRecognitionListener() 메소드를 통해 등록한다.

## 4. 데이터 디자인 (Data Design)

### 4.1 Data Description

본 절에서는 애플리케이션에서 생성되는 데이터가 어떤 구조를 형성하며, 관리되는지를 서술한다.



본 애플리케이션은 YouTube 영상마다 부여되어 있는 url 링크와 YouTube API를 이용하여 애플리케이션 내부에서 영상을 보여준다. 다음은 YouTube 영상의 url 저장 및 관련 기능을 위한 데이터베이스 테이블 스키마에 대한 설명이다.

테이블 이름 : link_video		
설명 : 저장된 YouTube 영상에 대한 정보를 나타냄		
column name	type	설명
video_id	INT	영상에 대한 id. 영상이 저장된 순서대로 1씩 증가하여 지정됨. primary key 속성을 가지고 있다.
name	VARCHAR	영상의 이름
url	VARCHAR	영상의 url
length	VARCHAR	영상의 길이
thumbnail_image	BLOB	영상의 썸네일 이미지 파일



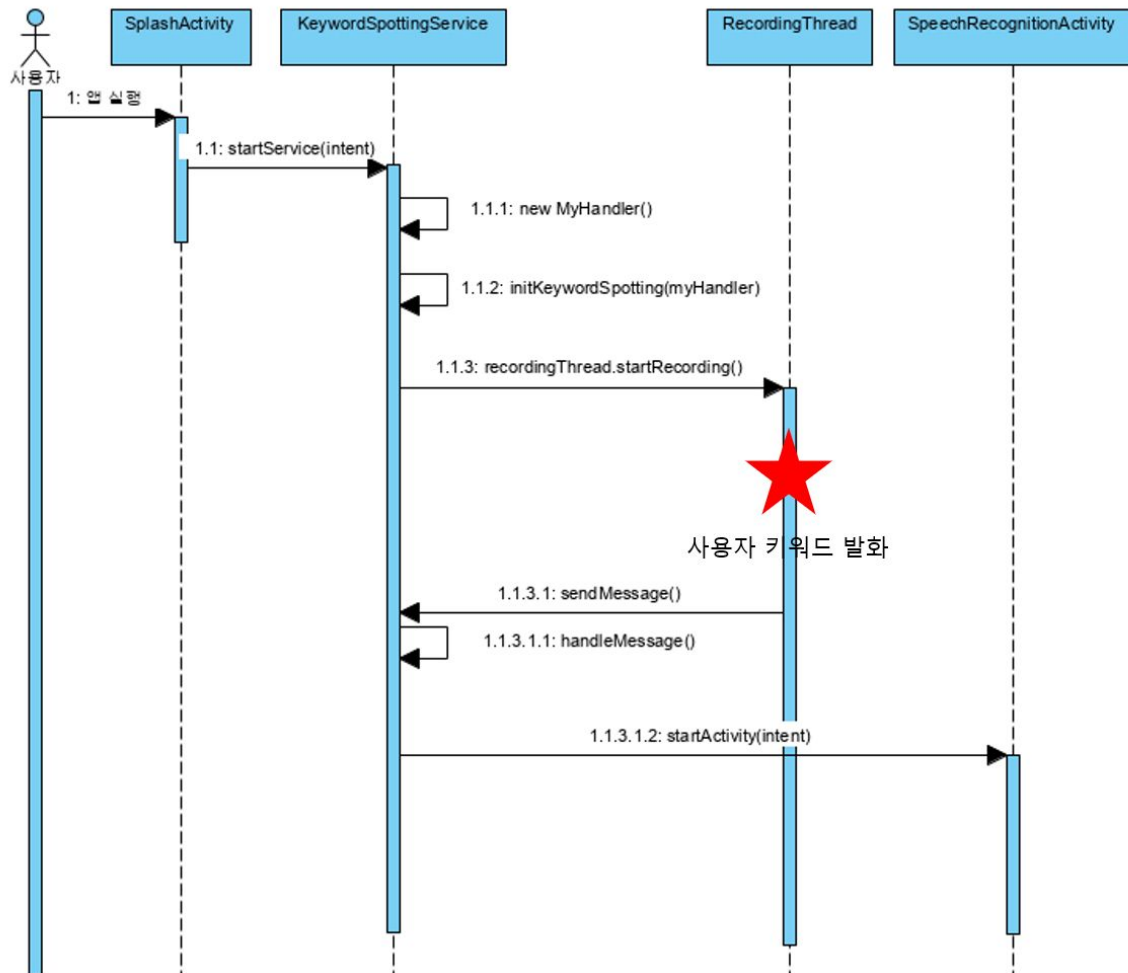
테이블 이름 : playlist 설명 : 사용자가 원하는 기준에 따라 영상을 분류하기 위한 재생목록 데이터를 나타냄		
column name	type	설명
playlist_id	INT	재생목록을 구분하는 id. 생성된 순서대로 부여되는 primary key이다.
playlist_name	VARCHAR	재생목록의 이름

테이블 이름 : link_video_to_playlist 설명 : YouTube 영상과 재생목록의 맵핑 관계를 기록한다.		
column name	type	설명
playlist_id	INT	영상이 저장된 재생목록의 아이디. playlist 테이블의 playlist_id 컬럼을 foreign key 속성으로 갖고 있다.
video_id	INT	재생목록에 저장된 영상의 아이디. link_video 테이블의 video_id 컬럼을 foreign key 속성으로 갖고 있다.

테이블 이름 : bookmark 설명 : 사용자가 시청했던 영상의 특정 시점에 부여한 북마크 정보		
column name	type	설명
bookmark_id	INT	영상의 특정 시점에 설정된 북마크에 대한 아이디. 생성된 순서대로 숫자를 부여받는 primary key 이다.
video_id	INT	해당 북마크가 설정된 영상의 아이디. link_video 테이블의 video_id를 foreign key 속성으로 갖는다.
bookmark_location	VARCHAR	북마크가 설정된 영상의 플레이 시점
bookmark_image	BLOB	북마크가 설정된 영상의 화면을 썸네일 이미지로 변환하여 저장

## 5. 컴포넌트 디자인 (Component Design)

이 장에서는 3.2절에서 보았던 음성인식모듈 관련 클래스들이 어떻게 동작하는지 조금 더 자세히 살펴본다.

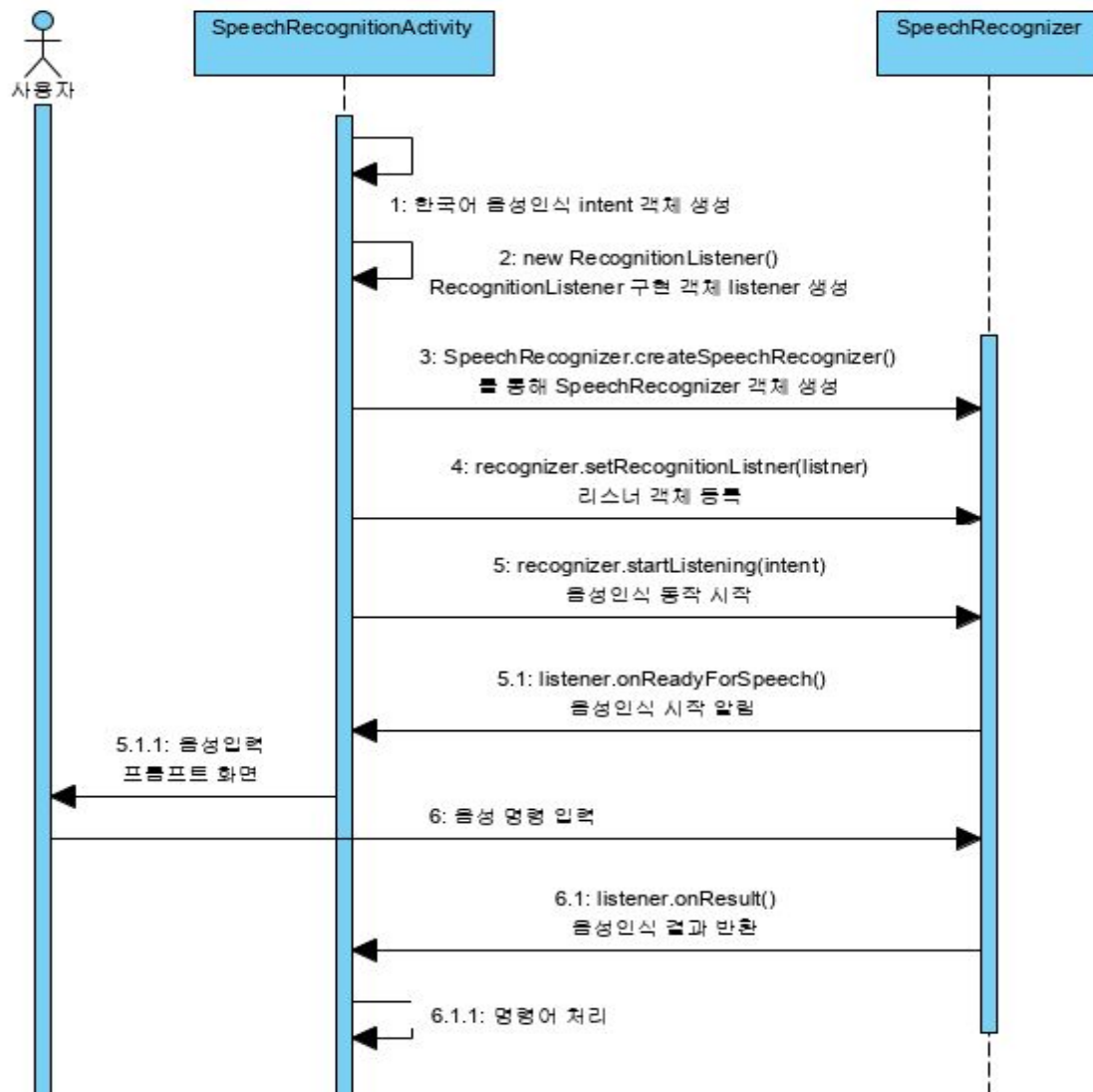


먼저 사용자가 **Wake-up word**를 발화하는지 체크하는 부분을 살펴본다.

- 맨 처음 사용자가 앱을 실행하면 SplashActivity가 실행된다.
- 이어서 KeywordSpottingService를 생성하기 위한 Intent를 초기화 시킨 후 startService(intent)를 통해서 서비스를 실행시킨다. KeywordSpottingService는 앱이 실행하는 동안 백그라운드에서 계속적으로 실행되는 것을 볼 수 있다.
- KeywordSpottingService는 new MyHandler() 함수를 통해서 MyHandler 인터페이스 객체를 구현한다. MyHandler는 RecordingThread에 넘겨져서 RecordingThread가 키워드를 인식할때 MyHandler에 구현된 인터페이스 메소드를 자동으로 실행하도록 한다. 이를 통해서 키워드 인식시 수행되어야 하는 동작들을 자동으로 실행되도록 할 수 있다.
- KeywordSpottingService는 initKeywordSpotting() 메소드를 실행한다. 클래스 멤버변수로 RecordingThread 타입의 recordingThread를 가지고 있는데, 이 RecordingThread 생성자를 이용하여 멤버변수를 초기화 한 다음, 앞에서 구현한 MyHandler 인터페이스 구현객체를 전달한다.

- recordingThread.startRecording() 메소드 호출로 키워드 인식을 구행하는 스레드를 생성하고 실행시킨다. 이 스레드는 앱이 실행되는 동안 계속적으로 동작하며 사용자가 키워드를 발화하면 전달받은 MyHandler 객체의 sendMessage()를 호출하여 키워드가 감지되었다는 것을 KeywordSpottingService 객체로 알려준다.
- sendMessage() 호출은 자동으로 handleMessage() 콜백 메소드를 호출하게 되며, 여기서 키워드 인식 후에 수행해야 하는 동작을 실행할 수 있다.
- 키워드가 감지되면 구글 stt 서비스를 이용하여 음성인식을 수행하는 Intent 객체를 초기화시키고, startActivity(intent)를 통해서 SpeechRecognitionActivity를 실행시킨다.

다음으로 음성인식 명령어를 처리하는 부분을 살펴본다.



- 키워드 발화가 감지되어 SpeechRecognitionActivity가 시작되면, 한국어 음성인식 관련 RecognizerIntent객체를 생성하여 멤버변수로 가진다.
- 그런 다음 음성인식의 각 단계에서 실행해줬으면 하는 동작들을 RecognitionListener 인터페이스로 구현하여 listener라는 객체를 생성한다.
- SpeechRecognizer.createSpeechRecognizer() 메소드 호출을 통해 SpeechReconizer 객체인 recognizer를 생성한다.

- recognizer.setRecognitionListener(listener)를 통해 앞에서 구현한 listener를 등록한다.
- recognizer.startListening(intent) 메소드 호출을 통해 한국어 음성인식을 시작한다.
- recognizer 내부에서 음성인식을 위한 준비를 마치면 listener.onReadyForSpeech() 콜백메소드를 호출하여 준비가 완료되었음을 SpeechRecognitionActivity에 알린다.
- SpeechRecognitionActivity에서는 사용자에게 프롬프트 화면을 띄워, 음성 명령어 입력을 요구한다.
- 사용자가 음성으로 명령어를 발화하면 SpeechRecognizer가 이를 인식한다.
- 음성인식이 완료되면 listener.onResult() 콜백 메소드를 호출하여 인식 결과를 SpeechRecognitionActivity에 전달한다.
- 인식결과를 바탕으로 명령어를 처리한다.

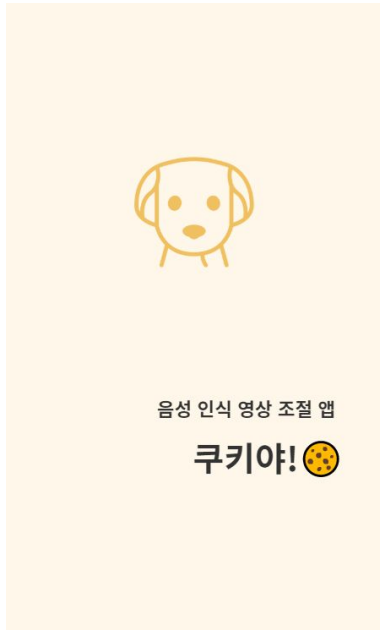
## 6. Human Interface Design

### 6.1 사용자 인터페이스 개요 (Overview of User Interface)

1. 특정 키워드 단어를 말하면, 해당 단어를 발화한 시점부터 음성인식 기능을 실행 -  
Keyword Spotting : 간단한 키워드를 인식하는 프로세스를 로컬에서 실행하고 있다가 해당 키워드를 인식한 시점부터 음성인식 기능을 실행하도록 한다.
2. 음성을 통해 유튜브 영상 재생, 멈춤 기능 : 사용자는 “쿠키야, 영상 재생해줘”, “쿠키야, 영상 멈춰줘”와 같은 문장을 발화하여 쿠키야 애플리케이션 내 영상을 재생하거나 멈출 수 있다.
3. 음성으로 영상 재생 시점을 조절 기능 : 사용자는 음성을 통해 애플리케이션 내 영상의 플레이 시점을 조절할 수 있다.
  - a. 영상 재생 시점을 특정 간격만큼 이동 : ex) “쿠키야, 10초 앞으로 이동”
  - b. 영상 재생 시점을 특정 시간대로 이동 : ex) “쿠키야, 3분 20초부터 재생”
4. 음성으로 영상에 북마크(Bookmark) 설정 및 특정 북마크로 이동 기능 : 사용자는 음성을 통해 영상의 특정 시점에 북마크를 설정할 수 있다. 여기서 북마크란, 영상의 특정 시점에 대한 일종의 레이블(Label)이다. 이 레이블은 숫자(1, 2, 3, ... 과 같은 자연수)로 표시되며, 사용자가 음성을 통해서 영상 플레이 시점을 조절할 수 있는 기준이 된다.
  - a. 영상이 재생되고 있는 현재의 재생 시점에 북마크 설정 : ex) “쿠키야, 여기에 북마크”
  - b. 영상 재생 시점을 특정 북마크 시점으로 이동 : ex) “쿠키야, 1번 북마크로 이동”
5. 유튜브 영상을 애플리케이션 내부로 저장 기능 : 사용자는 구글 사의 애플리케이션 ‘YouTube’내의 공유하기 기능을 이용하여 영상을 ‘쿠키야’ 애플리케이션에 저장할 수 있다.
6. 음성을 통해 ‘쿠키야’ 애플리케이션 조작 기능 : 사용자는 음성을 통해 본 애플리케이션 ‘쿠키야’를 조작할 수 있다.
  - a. 음성을 통해 특정 순서의 영상을 재생 : ex) “쿠키야, 1번 영상 재생”
  - b. 음성을 통해 앞 또는 뒤의 영상을 재생 : ex) “쿠키야, 다음 영상 재생”
  - c. 음성을 통해 영상 목록 화면으로 이동 : ex) “쿠키야, 영상 목록으로”
  - d. 음성을 통해 애플리케이션 종료 : ex) “쿠키야, 프로그램 종료”

## 6.2 Screen Images

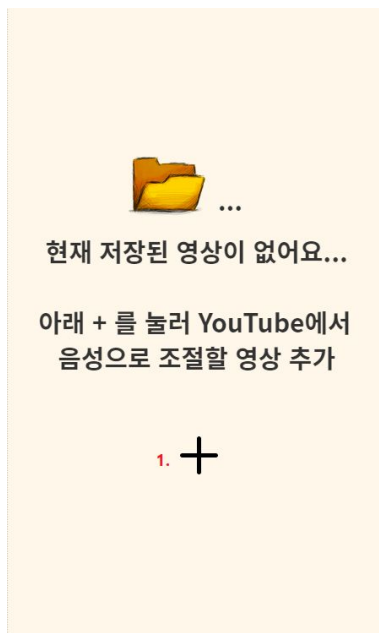
### 1. 로딩화면



- 사용자가 애플리케이션을 실행시키면 '쿠키야!'가 시작하는 중임을 나타낸다.

- 이 화면이 나타나는 동안 애플리케이션이 동작하기 위해서 필요한 작업을 내부적으로 수행할 수 있다.

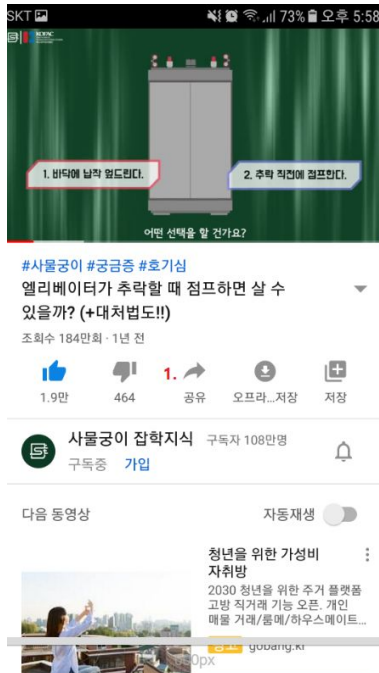
### 2. 영상목록 화면(애플리케이션에 저장된 영상이 없는 경우):



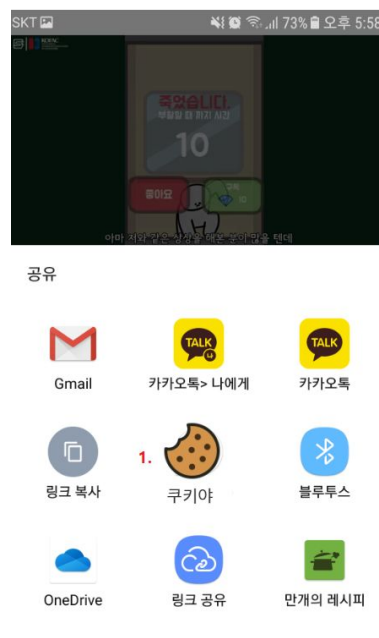
- '쿠키야!' 애플리케이션을 설치한 후 처음 실행하거나, 애플리케이션에 저장한 YouTube 영상이 없는 경우에 보여지는 화면이다. 본 애플리케이션('쿠키야!')은 YouTube 애플리케이션과 함께 동시 동작하는 애플리케이션이 아니다. 사용자는 YouTube에서 본인이 음성으로 조절하고 싶은 영상의 링크를 본 애플리케이션에 저장한 뒤, 본 애플리케이션이 독자적으로 동작하면서 영상을 조절하게 된다. 이 화면은 다음과 같은 기능이 있다.

- 숫자 1의 위치 오른쪽에 위치한 플러스(+) 기호 모양의 버튼이 존재한다. 이 버튼을 터치하는 경우 YouTube 애플리케이션이 실행된다. 만약 YouTube 애플리케이션이 존재하지 않는다면, 웹 브라우저가 실행된 뒤에 YouTube 웹 사이트로 이동하게 된다.

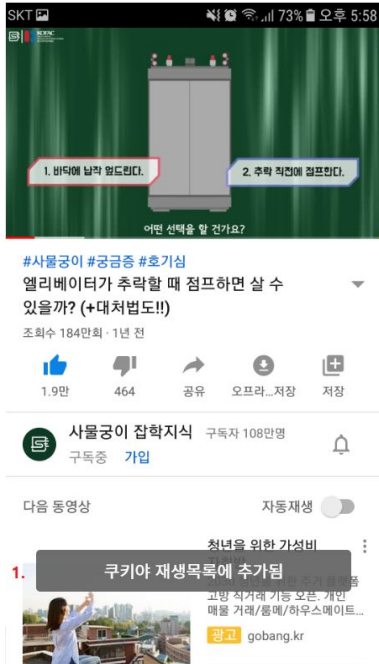
사용자는 YouTube 애플리케이션에 존재하는 임의의 영상을 다음과 같은 방법으로 본 애플리케이션('쿠키야!')에 저장할 수 있다.



- YouTube의 영상 재생화면에서 숫자 1의 오른쪽에 위치한 '공유' 버튼을 클릭한다.

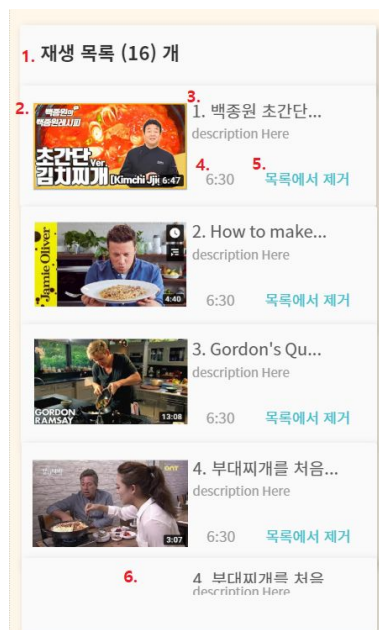


- 공유 기능에 대한 대화상자가 등장했을때, 숫자 1 위치 오른쪽에 나타난 것처럼 본 애플리케이션('쿠키야!')에 대한 아이콘이 나타난다. 해당 아이콘을 터치하면 '쿠키야!' 애플리케이션 내부에 현재 보고 있는 영상의 링크 (URL 형태의 Text)가 저장된다.



- 영상이 성공적으로 저장되면, 숫자 1 위치 오른쪽에 나타난 것처럼 안드로이드의 토스트 메시지를 통해서 사용자에게 영상이 성공적으로 저장되었음을 나타낸다.

### 3. 영상목록 화면(애플리케이션에 저장된 영상이 있는 경우)



- YouTube에서 음성으로 조절할 영상을 본 애플리케이션('쿠키야!') 저장한 후에는 왼쪽과 같은 화면이 나타난다. 이 화면은 다음과 같은 기능이 있다.

- 숫자 1 오른쪽에 나타난 것처럼 현재 애플리케이션에 저장된 영상의 링크 개수를 보여준다.

- 각 영상은 카드형태의 구조로 보여진다. 임의의 영상에 대한 카드를 나타내는 영역을 터치하는 경우 해당 영상을 재생하는 화면으로 이동한다.

- 숫자 2 오른쪽에 나타난 것처럼 영상의 썸네일이 카드 구조 왼쪽에 나타난다.

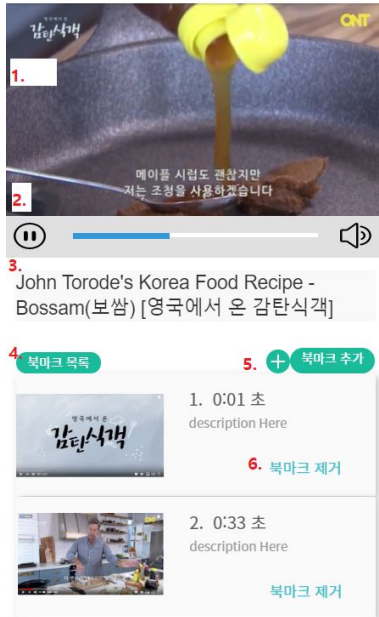
- 숫자 3 아래쪽에 나타난 것처럼 영상의 제목이 나타난다. 영상 제목 왼쪽에는 숫자가 부여되어 있는데, 해당 영상이 '쿠키야!' 애플리케이션에 저장된 순서대로 부여된 숫자이며, '이전 영상'과 '다음 영상'을 판단하는 기준이 된다.

- 숫자 4 아래쪽에 나타난 것처럼 영상의 길이를 나타낸다.

- 숫자 5 아래쪽에 나타난 것처럼 '목록에서 제거' 버튼이 있어 사용자가 더이상 저장해두고 싶지 않는 영상을 해당 버튼을 눌러서 제거할 수 있다.

- 숫자 6이 나타나는 위치처럼 스마트기기의 화면에 모두 나타낼 수 없을 만큼 저장된 영상이 많다면 이 후 영상은 보이지 않고, 사용자가 손가락을 위로 스윙할 경우 다음 영상에 대한 카드들이 보여지는 '스크롤뷰' 형태이다.

#### 4. 영상 재생화면

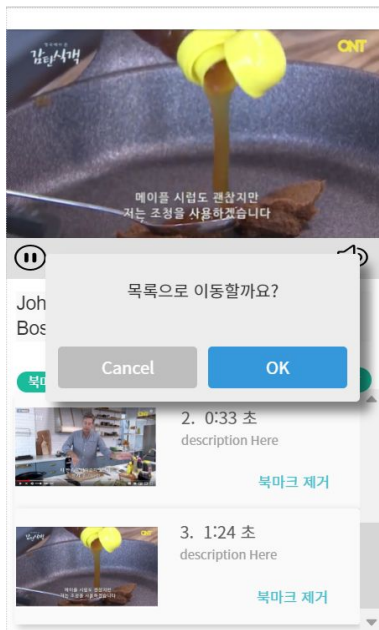


영상 목록화면에서 재생을 원하는 영상을 터치하면 나타나는 화면이다. 다음과 같은 기능이 있다.

- 숫자 1 위치에 나타난 것처럼 비디오 플레이어를 이용해서 해당 영상을 플레이한다.
- 숫자 2 아래쪽에 나타난 것처럼 비디오 플레이어의 컨트롤러가 나타난다.
- 숫자 3 오른쪽에 나타난 것처럼 영상의 제목이 나타난다.
- 숫자 4 아래쪽에 나타난 것처럼 사용자가 지정한 북마크의 목록이 나타난다. 각 북마크는 카드형태로 나타나며, 썸네일, 북마크 시간의 위치, 북마크 제거 버튼이 나타난다. 여러 개의 북마크가 존재하여 기기 화면에 모두 나타나지 않는 경우 아래쪽에 위치한 북마크는 가려지게 되며, 위로 스윙핑 했을 때 나머지 북마크가 나타나는 '스크롤뷰' 구조다.

- 숫자 5 오른쪽에 나타난 것처럼 '북마크 추가' 버튼이 있으며 해당 버튼을 터치하는 경우 플레이 되는 영상의 현재 시점에 대한 북마크가 생성된다.
- 숫자 6 오른쪽에 나타난 것처럼 '북마크 제거' 버튼이 있으며 해당 버튼을 터치하는 경우 북마크가 제거된다.

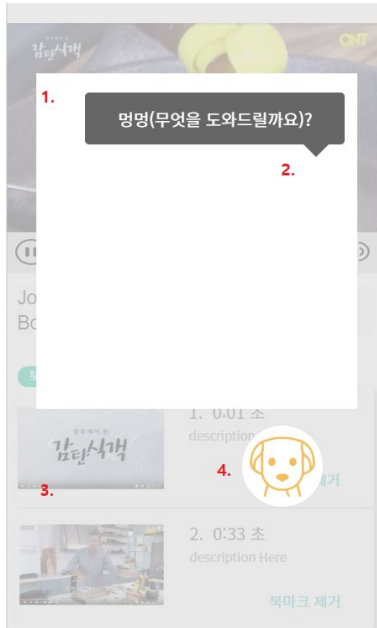
#### 5. 목록으로 이동하기 확인 대화상자



영상 재생화면에서 안드로이드 기기의 '뒤로가기' 버튼을 누른 경우 나타나는 확인 대화상자이다. OK 버튼을 누르는 경우 재생목록으로 이동하며, Cancel 버튼을 누르는 경우 원래 화면으로 복귀한다. 사용자가 영상을 보다가 실수로 뒤로가기 버튼을 누른 경우 의도하지 않게 목록 화면으로 넘어가지 않도록 하기 위해 대화상자가 나타난다.



## 6. Keyword Spotting이 일어났을 경우



본 애플리케이션(‘쿠키야!’)을 실행하는 도중에 “쿠키야”라고 발화하면 애플리케이션의 백그라운드에서 실행되고 있던 Keyword Spotting 서비스는 Wake up word인 “쿠키야”를 인식한다. 그 다음 왼쪽과 같은 화면이 나타난다. 세부사항은 다음과 같다.

- 영상이 재생되고 있었다면, 영상을 일시정지한다.
- 숫자 1이 위치한 영역과 같이 대화상자가 나타난다.
- 숫자 2 위쪽에 나타난 것과 같이 사용자에게 어떤 명령을 내릴 것인지에 대한 프롬프트 메시지를 보여준다. 또한 메시지에 나타난 문장은 TTS(Text-To-Speech) 기능을 통해서 소리의 형태로 사용자에게 전달한다.
- 숫자 3이 위치한 영역과 같이 이전에 보고 있었던 화면은 흐려진다.
- 숫자 4 오른쪽에 나타난 것처럼 강아지 모양이 사용자에게 현재 음성인식 기능이 실행 중임을 나타낸다. 이 이미지는 음성인식 기능이 활성화된 경우에 모두 나타난다.

## 7. 음성을 통해 영상 멈춤



Wake up word를 통해서 음성인식 기능을 활성화 시킨 뒤에 “영상 일시정지”와 같은 문장을 발화하는 경우 나타나는 화면이다. 세부사항은 다음과 같다.

- 숫자 1 위쪽에 나타난 것처럼 사용자가 발화한 문장을 텍스트 형태로 보여줘서 사용자 본인이 발화한 내용과 일치하는지 확인할 수 있게 한다.
- 숫자 2 위쪽에 나타난 것처럼 유효한 명령어를 인식한 경우 “명명(알겠어요!)”와 같은 응답이 메시지 형태로 보여진다.
- 응답 메시지가 보여진 후에는 원래 화면으로 돌아가 영상을 일시정지 한다.
- 이와 같은 방법으로 음성을 통해 영상 다시 재생, 영상간격 이동, 북마크 생성, 북마크로 이동, 이전 및 다음 영상 재생, 특정 영상 재생, 재생목록으로 이동, 프로그램 종료 등의 기능을 수행할 수 있다.

6.1절의 3,4,6의 화면이미지는 위 7번의 이미지와 동일하다.

## 7. 프로젝트 관리 (Milestone)

	7주 차	8주 차	9주 차	10 주 차	11 주 차	12 주 차	13 주 차	14 주 차	비고
구글 stt 사용법 정리 및 샘플 코드 작성									
snowboy 사용법 정리 및 샘플 코드 작성									
YouTube android api 사용법 정리 및 샘플 코드 작성									
KeywordSpottingService 구현									
SpeechRecognitionActivity 구현									
애플리케이션에 사용할 이미지 디자인									
영상 재생화면 구현									
음성인식 프롬프트 화면 구현									
북마크 기능 구현									
초기 화면, 영상 목록화면 구현									