

# 캡스톤 디자인

## 2020학년 1학기 과제 보고서

(제목 : Realtime STT모듈을 활용한  
키워드 블랙박싱)

제출일자 : 2020. 06. 30

참여자구분	성 명	학번	학년	역할	비고
팀원	나선엽	201432018	4	코더, DB	
팀원	조성진	201432067	4	코더, DB, 앱 디자인	

## 요 약

우리가 일상생활에서 놓치고 있는 대화내용을 키워드 형식으로 저장하여 지나간 대화들을 찾고 싶은 사용자들에게 보여주는 어플입니다. 먼저 사용자는 액티비티와 백그라운드에서 녹음을 진행하고, 알고리즘을 통해 대화내용을 키워드화 하고, 빈도수가 높으면 글자가 커지는 워드 클라우드 형태, 단어의 빈도수와 키워드를 보여주는 리스트뷰로 구현하였다.

## 1. 서론

지금 사회는 정보화 산업 시대이다. 자그마한 것 하나까지 데이터화 하여 이용하며 우리의 삶과 데이터는 떼 놓을 수 없는 사이가 되었다. 국가에서는 국민들의 정보를, 회사에서는 사원들의 정보를, 은행에서는 고객들의 정보를, 개개인은 사사로운 정보들을 모두 정보 기기에 저장·관리 한다. 스마트폰은 물론 이제는 사물인터넷까지 등장하여 생활의 모든 면에서 정보와 매우 밀접한 관련이 있는 가운데 우리들이 놓치고 있는 데이터들도 분명히 존재한다고 생각했고, 사소하지만 소중한 데이터를 놓치지 않기 위해 이 연구를 진행했다.

### 1.1 주제 선정 이유

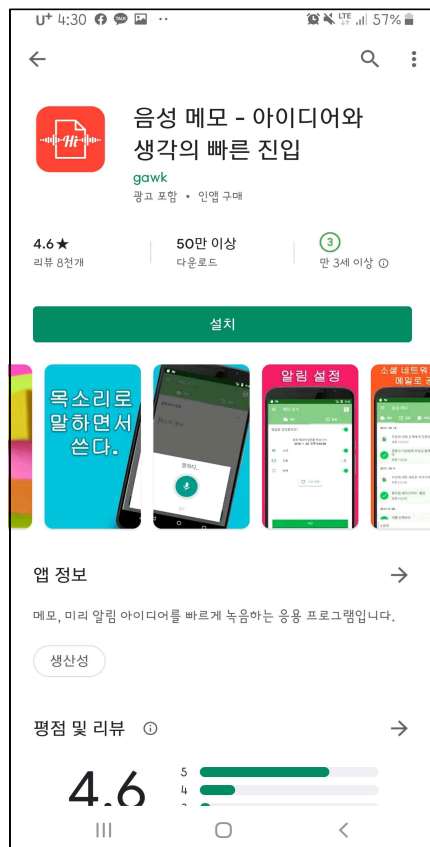
최근 정보를 이용한 수많은 앱이나 프로그램이 개발되고 있다. 대중적으로는 타임피드를 이용해 사람들과 소통하는 SNS(Social Network Service)어플, 단순 카메라에서 여러 필터링과 기능을 넣어 사람들에게 사진 데이터를 예쁘게 남겨주는 카메라 어플, 플레이 데이터를 이용하여 여러 정보를 제공하는 무수한 게임 어플 등등. 이미 우리는 여러 데이터를 다루고 있다. 하지만 분명 우리가 놓치고 있는 데이터 또한 존재한다고 생각했다. 사람들은 지나간 대화내용을 기억하지 못하는 상황을 종종 겪는다. 친구와의 대화나 동료들과의 회의시간에서 했던 대화의 내용들을 따로 기록을 해 놓았다면 당연히 기록을 보고 떠올릴 수 있겠지만, 기록을 할 수 없는 상황이거나 당시의 불필요성으로 인해 기록을 해놓지 않았다가 나중에 떠올리려면 당연히 가물가물하여 기억하기 쉽지 않을 것이다. ‘그렇다면 우리가 기록하지 않는 상황에서라도 누군가 기억하고 있다면 어떨까? 이후에 지나간 대화내용의 키워드라도 하나 던져준다면 그 때의 내용을 기억하기 훨씬 수월하지 않을까?’ 라는 생각들로 스마트폰으로 하여금 우리의 대화 내용을 기억하게 하는 목적으로 이 연구를 시작했다.

## 2. 관련 과제 또는 연구

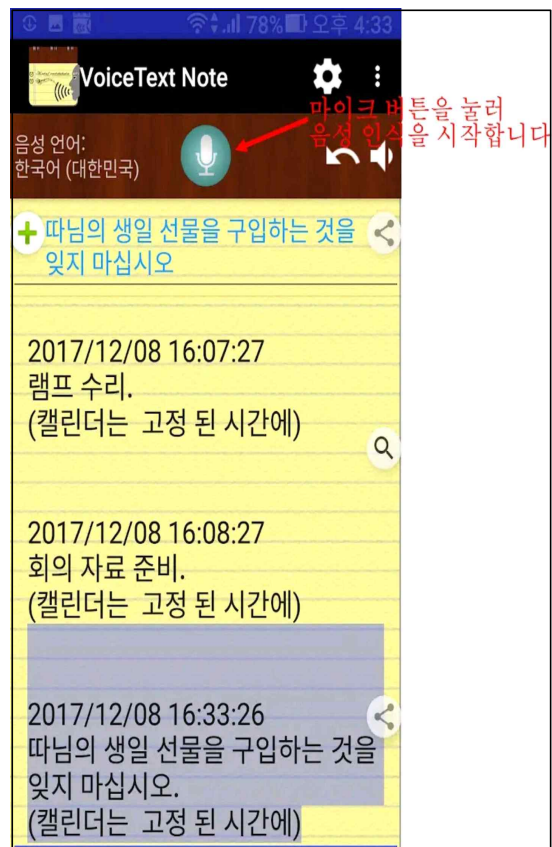
### 2.1 관련 어플 조사

음성을 텍스트로 변환하여 메모해주는 어플을 찾아본 결과 메모장+녹음 기능을 활용하여 원하는 메모를 음성으로 기록하는 어플이 존재했다. 추가적으로 메모들을 편집할 수 있고, 시간대 별로 메모들이 정리되었다.

[그림 1]과 [그림 2]은 유사 어플리케이션이다.



[그림 1] 스피치 메모



[그림 2] 음성 입력 공책

## 2.2 관련 기능 조사

실시간으로 음성을 계속 입력받을 수 있는 방법과, 해당 기능을 서비스로 바꿔서 백그라운드에서 음성 입력을 받을 수 있는 방법을 연구했다. 그리고 안드로이드에서 제공중인 STT 엔진과 접목하여 해당 음성 데이터를 Text화 하여 다루는 방법을 연구했다. 이후에 Text로부터 Keyword를 추출해내는 알고리즘에 대한 조사를 진행했다.(Keyword 추출 기능 관련 알고리즘은 직접 개발하기로 결정)

## 3. 요구사항 분석

### 3.1 개발 시스템의 목표

스마트폰을 조작할 수 없거나 불필요성에 의해 기록하지 않는 상황에서도 스마트폰의 STT 엔진을 통해 실시간 대화들을 Text화 하고 Keyword들을 뽑아내어 사용자에게 각종 방식으로 제공하는 어플

## 3.2 기능 요구사항

### 3.2.1 Background Listening

안드로이드 내장 Voice Recorder(Audio)기능을 이용해 음성 데이터를 다룸. 이를 Service화 하여 Background에서도 Running가능한 Audio 엔진을 구성(이후 STT엔진과 결합)

### 3.2.2 Resting Period

음성 데이터가 입력되면 호출되는 문장 휴지기 판단 기능(음성이 들어오고 있는지 판단하는 기능). 음성 데이터가 입력된 후에 1초 동안 음성 데이터가 입력되지 않는 구간이 발견되면 STT를 호출하여 지금까지 변환된 Text들을 처리하고 다시 음성 데이터 입력을 기다리는 역할을 하는 기능

### 3.2.3 Realtime STT(Speech-To-Text)

Resting Period 기능에 의존하여, 음성 데이터의 입력이 끝났다고 판단되면 호출되는 기능으로 현재까지 입력되어 버퍼에 저장되었던 데이터들을 시스템에 Text화 하여 제공하는 기능

### 3.2.4 Extract Keyword

STT엔진으로부터 받아온 Text들을 분석하여 대화 내용중 중요 키워드를 도출해내는 알고리즘(완벽하지 않은 관계로 이후에 지속적인 업데이트 필요)

### 3.2.5 Database Management

안드로이드 내장 Database(이하 'DB')를 다루는 기능으로 도출된 Keyword나 빈도수 도출된 시간(Time Stamp)등을 DB에 저장하고, 사용자로 하여금 DB를 백업하거나 초기화 하는 작업을 실행하는 기능

### 3.2.6 Wordcloud(External Service)

본 어플(시스템)과 연동된 외부 서비스로, 입력받은 단어(Keyword)들을 빈도수에 따라 WordCloud 형식으로 만들어 제공하는 기능

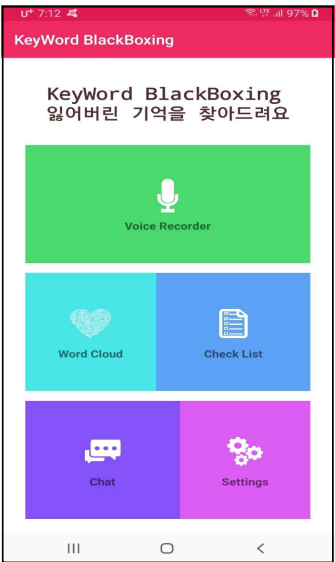
### 3.2.7 Widget Service

어플을 키지 않은 상태에서도 블랙박스 서비스를 ON/OFF할수 있는 스위치를 가진 위젯 기능(6월30일 까지 개발이었으나, 현재는 구상만 해두고 구현 하지 못한 상태입니다.학기 방학 중 구현예정)

요구사항 분석의 결과 위와 같은 기능들을 도출해내고 유스케이스와 그림들을 그렸다.

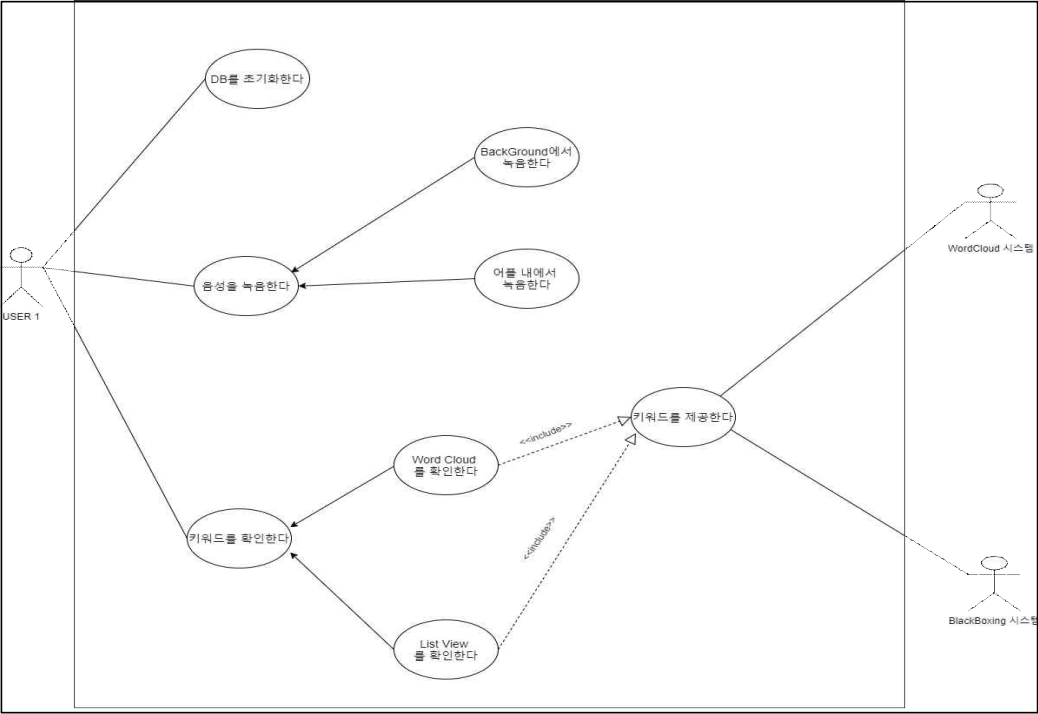
#### 4. 설계

[그림 3]은 요구사항 분석을 기초로 하여 설계한 메인 화면이다. Keyword Blackboxing 기능을 시작할 수 있는 Voice Recorder 화면과 키워드를 확인하는 방법인 Word Cloud, Check List 화면 그리고 대화 기록을 볼 수 있는 Chat 화면과 Setting 화면등이 구성되어있다.



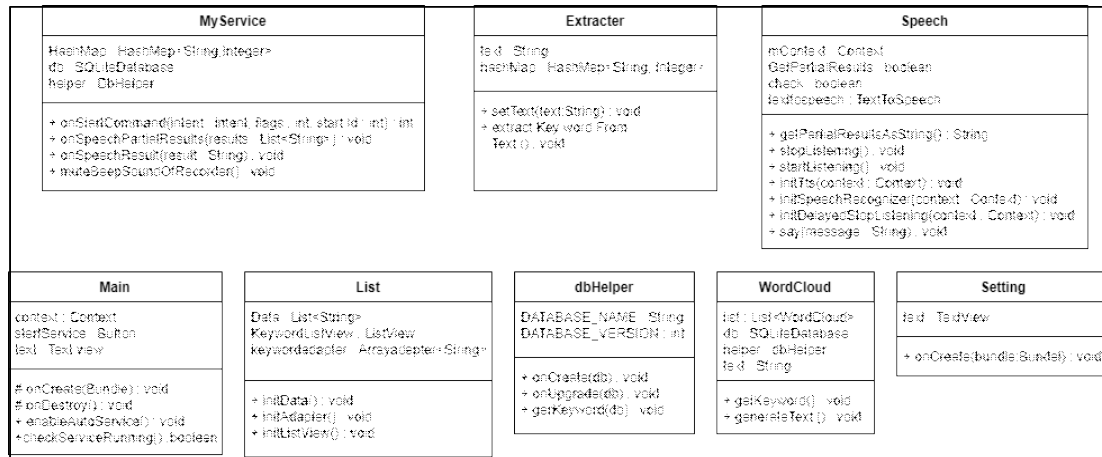
[그림 3] 메인 화면

[그림 4]는 사용자 입장에서 시스템의 동작을 나타낸 유스케이스 다이어그램이다.



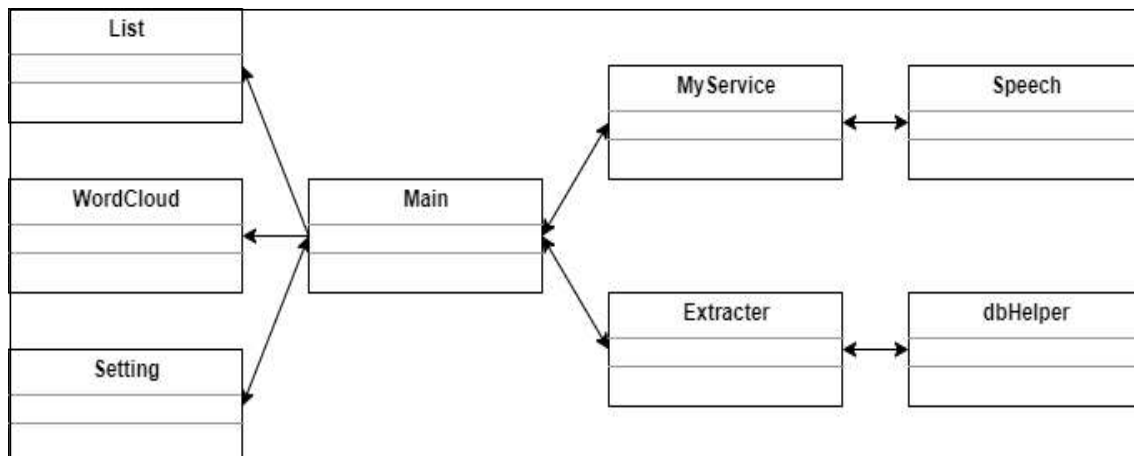
[그림 4] 유스케이스 다이어그램

[그림 5]는 객체 설계 그림이다. 요구사항 분석 결과를 기반으로 Service나 DB관련 객체, 추출 알고리즘객체 등 여러 객체를 구상했다.



[그림 5] 객체 설계

[그림 6]은 [그림 5]에서 설계된 객체들의 관계를 나타낸 객체 관계 그림이다.



[그림 6] 객체 관계

## 5. 구현 결과

### 5.1 개발환경

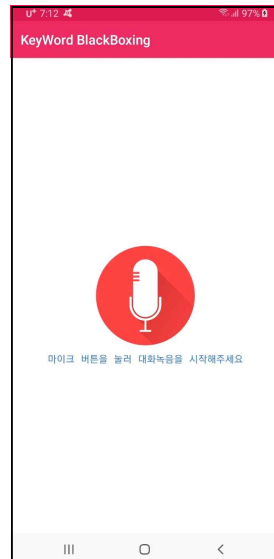
본 앱은 JDK를 기반으로 Android Studio를 이용하여 Android 플랫폼에서 실행 가능한 어플로 만들어졌다. 앱의 개발은 Window환경에서 Java언어를 사용하여 개발하였다.

### 5.2. 구현

본 앱은 Audio모듈과 STT를 결합하여 만든 실시간 음성 변환 시스템과 내장 알고리즘으로 추출한 Keyword를 Viewing하는 기능으로 구성되어 있다.

다음의 [그림 7]과 [그림 8]은 음성 녹음 화면으로 각각 녹음 시작 전/후를 나타낸다.  
[그림 7]을 보면 입력받은 음성을 \*Toast 메시지로 띄워주는 것을 볼 수 있다.

\* Toast 메시지 : 안드로이드 화면에 메시지 형태로 팝업을 띄워주는 기능



[그림 7] 녹음 시작



[그림 8] 일시 중지

[그림 9]는 Background상태(바탕 화면)의 서비스 실행 결과를 보여준다. 어플 외부의 상황에서도 [그림 8]과 마찬가지로 Toast 메시지를 이용한 팝업이 정상적으로 띄워지는 것을 볼 수 있다.



[그림 9] 바탕 화면

[그림 10] Wordcloud 화면

[그림 11] Listview 화면

이번 과제를 진행하면서 안드로이드 기반의 어플을 제작할 때 실시간으로 데이터들을 다루고 서비스를 제공하는 방법에 대해 많은 고민과 어려움이 있었지만 비슷한 연구들의 결과나 논문을 참고하여 시행착오 끝에 실시간으로 키워드를 제공하는 서비스를 개발할 수 있었다. 현재까지는 모듈이나 알고리즘에 대한 부분이 집중적으로 개발되었고 지속적으로 모듈 및 알고리즘의 업데이트가 진행 될 예정이다. 향후 과제 수행자들은 시스템 방면에서 보다는 Keyword 추출법에 대한 연구가 필요하고, 서비스 방면에서 앱의 사용자들에게 보기 좋게 서비스(데이터 제공)하는 방법의 연구가 필요하다.

[1] 이성직 · 김한준, TF-IDF의 변형을 이용한 전자뉴스에서의 키워드 추출 기법, 한국전자거래학회지, 제14권, 4호, 2009년, pp. 59-72

[2] 사이트: <https://github.com/natu10/Speech-Recognizer>

[3] 사이트: <https://github.com/alhazmy13/AndroidWordCloud>