

# 비대면 환경에서의 효과적인 교육 및 비즈니스를 위한 개인 맞춤형 융합 콘텐츠 생성 기반 기술

---

최종발표

김이홍조

2016112166 김운호

2017112085 이유경

2017112099 조민지

2017111718 홍은주

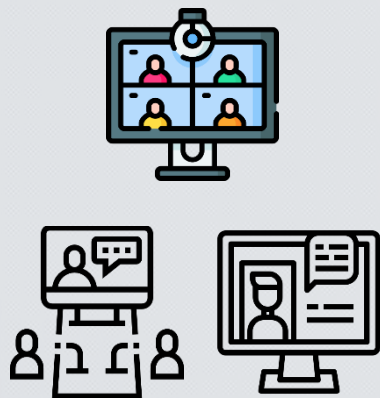
# 목차

---

1. 프로젝트 개요
2. 프로젝트 요구사항
3. 진행상황
4. 앞으로의 계획

# 프로젝트 개요

비대면 환경에서의 효과적인 교육 및 비즈니스를 위한  
개인 맞춤형 융합 콘텐츠 생성 기반 기술



## 비대면 환경에서의 영상

- 소통의 핵심
- 다양한 정보를 포함

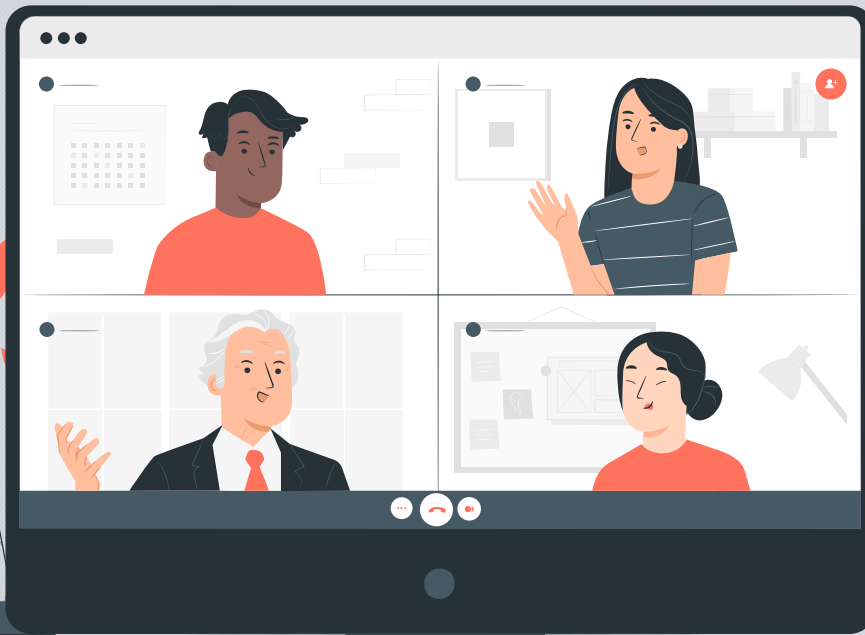


자막기술



# 프로젝트 개요

keyword : **교육**



동영상



수업자료

# 프로젝트 개요

## 1. 사용자가 자막을 관리할 수 있는 통합적인 기능

- 생성, 수정, 삭제, 저장, 불러오기

## 2. 실시간 및 로컬 영상의 자동 자막 생성

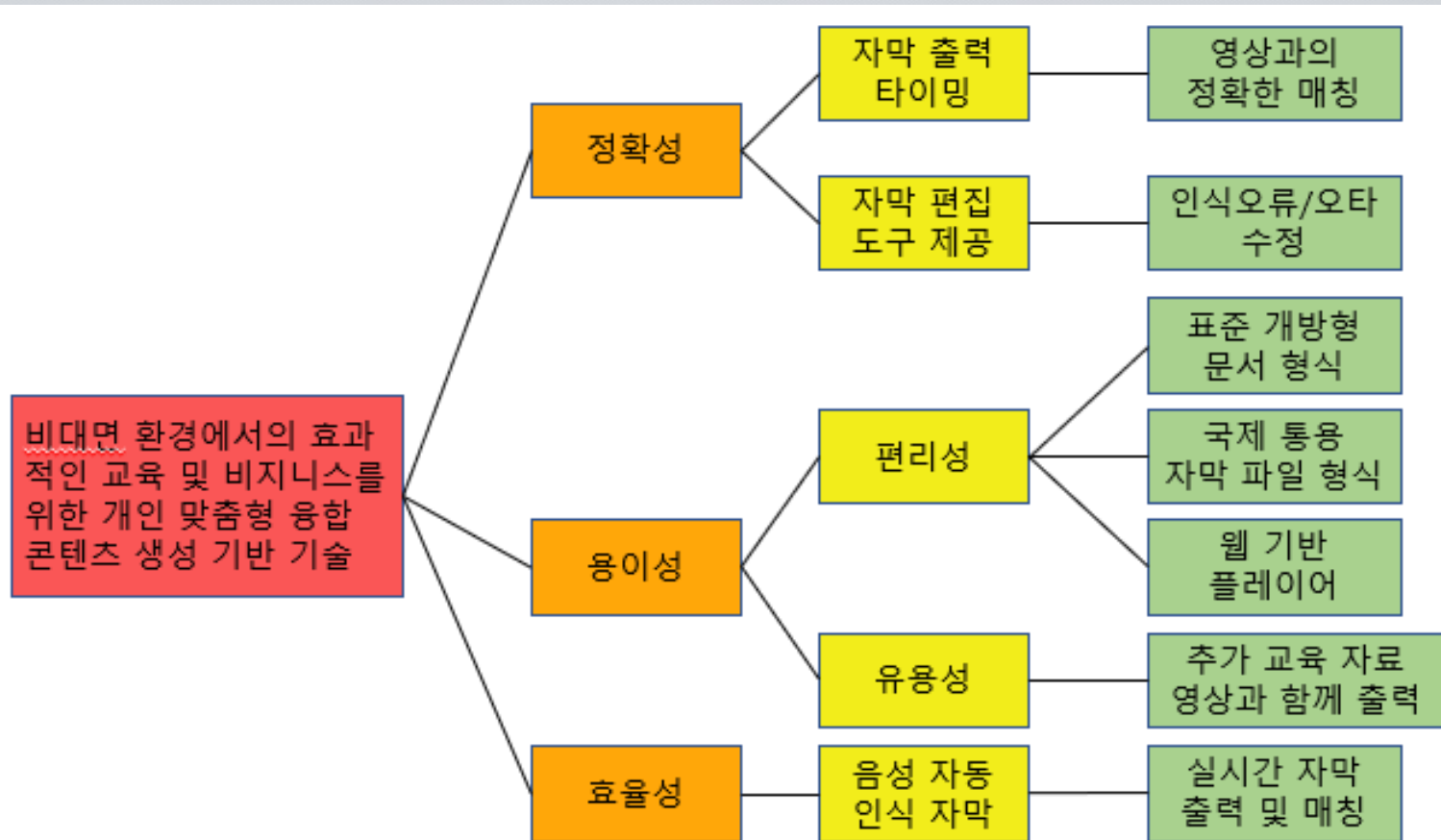
## 3. 영상에 참고할 수 있는 파일 자막화

## 4. 생성된/저장된 자막에서 키워드를 검색

## 5. 자막 클릭 시 매칭 되는 영상을 부분을 재생하는 기능(책갈피)

# 프로젝트 요구사항

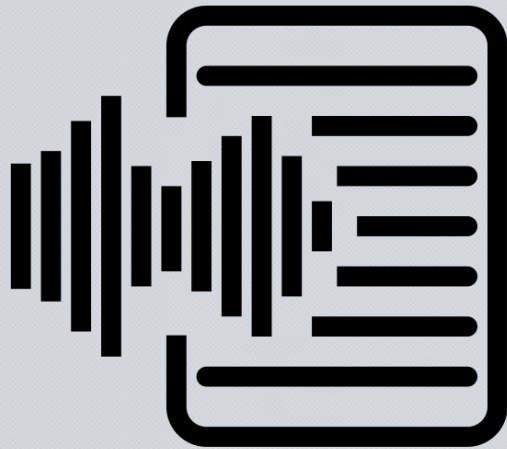
## 요구 사항 정의를 위한 목적 나무





# 프로젝트 요구사항

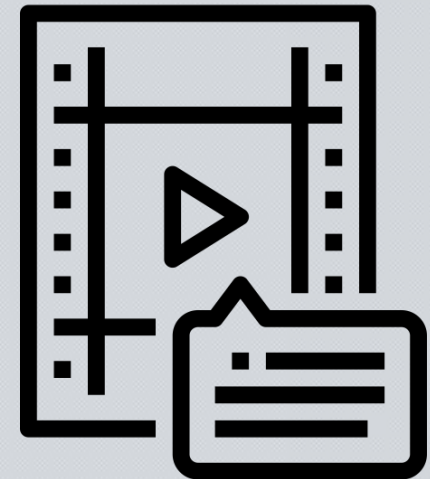
## 자막 종류 세분화



AI 음성인식



ODF 파일

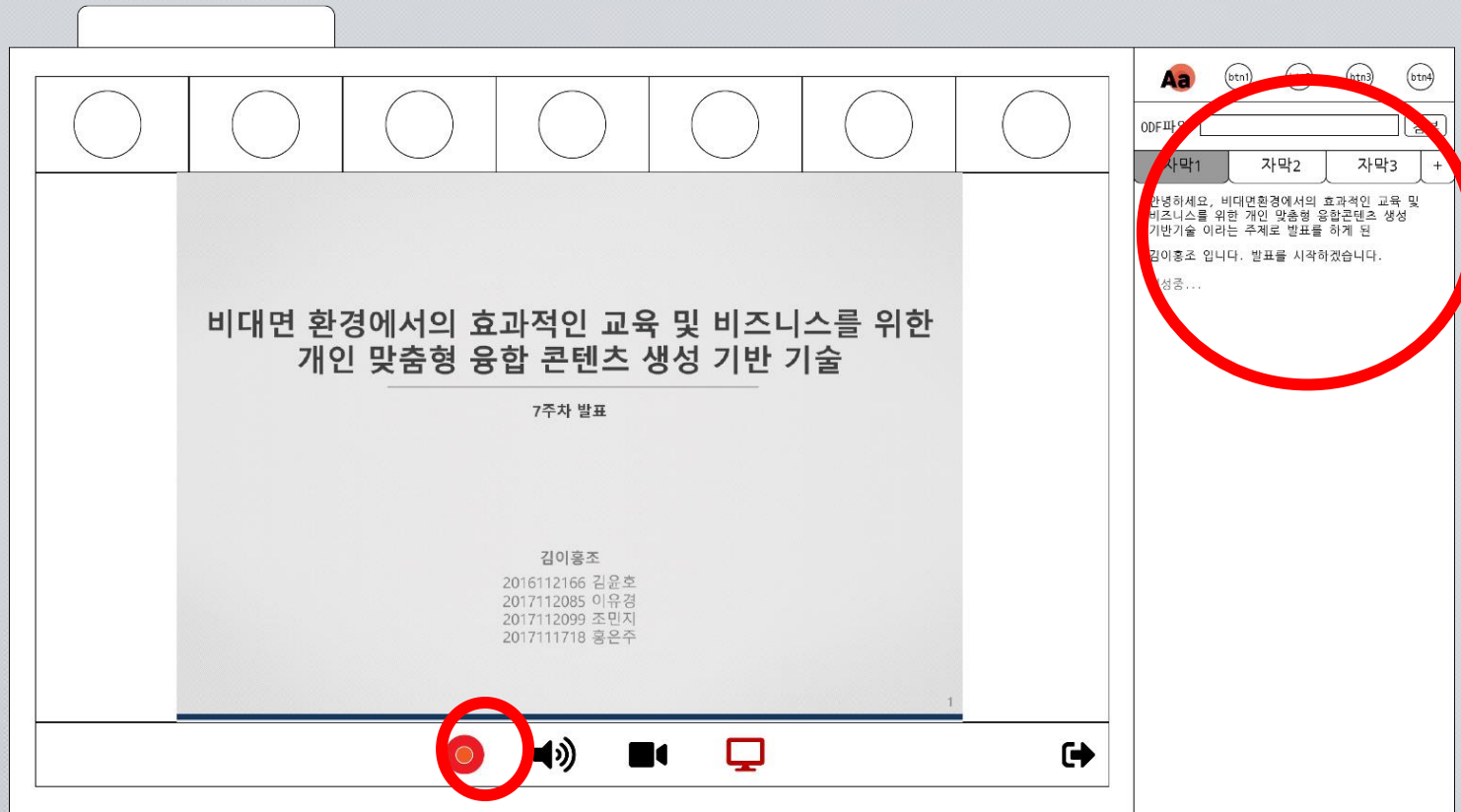


사용자 생성

# 프로젝트 요구사항\_UI/UX

## 자막 종류 세분화

### - AI 음성인식

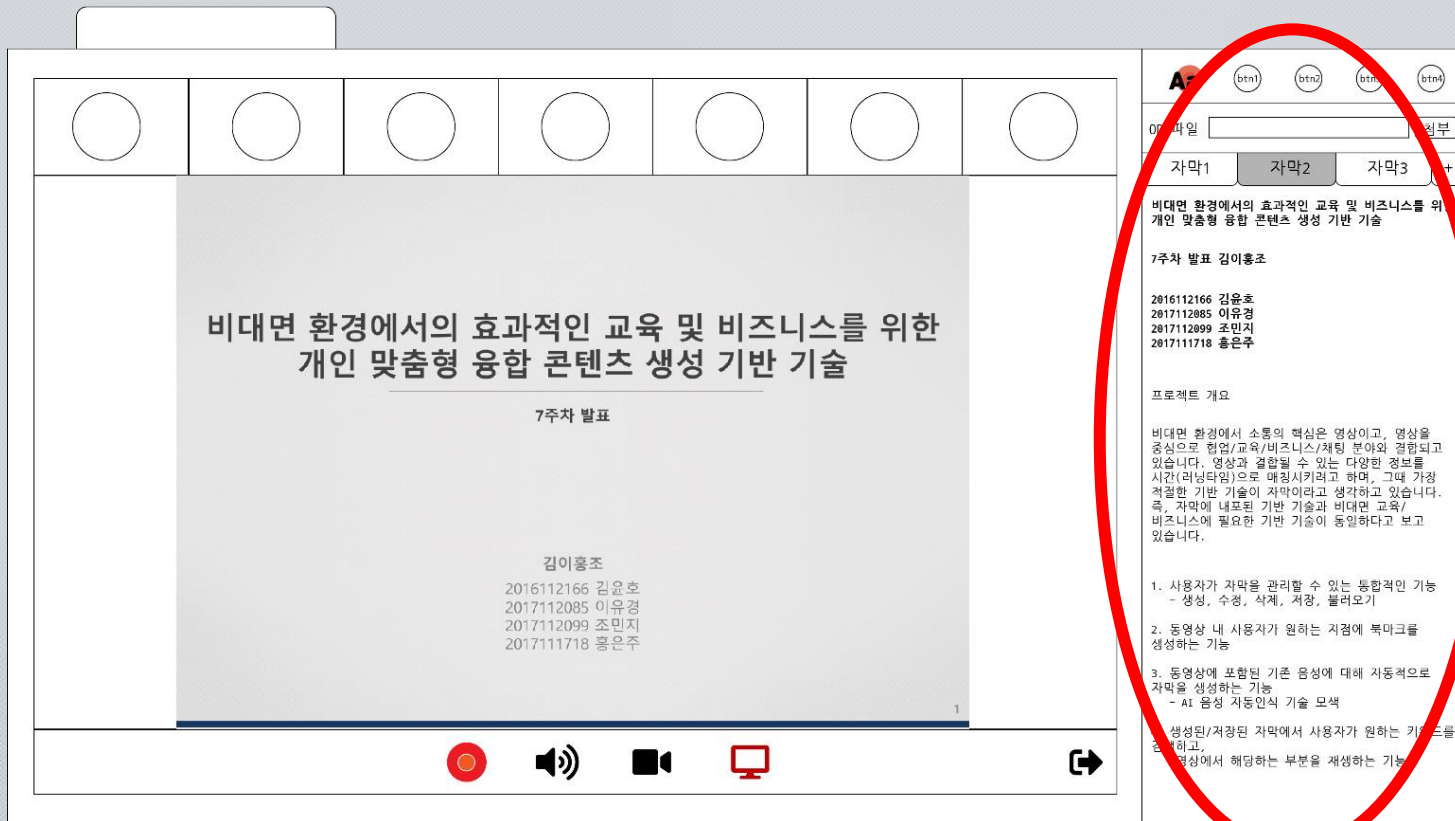


음성 인식 엔진을 통해서  
영상의 화자 인식 및  
자동 자막 생성



# 프로젝트 요구사항\_UI/UX

## 자막 종류 세분화 - ODF 파일



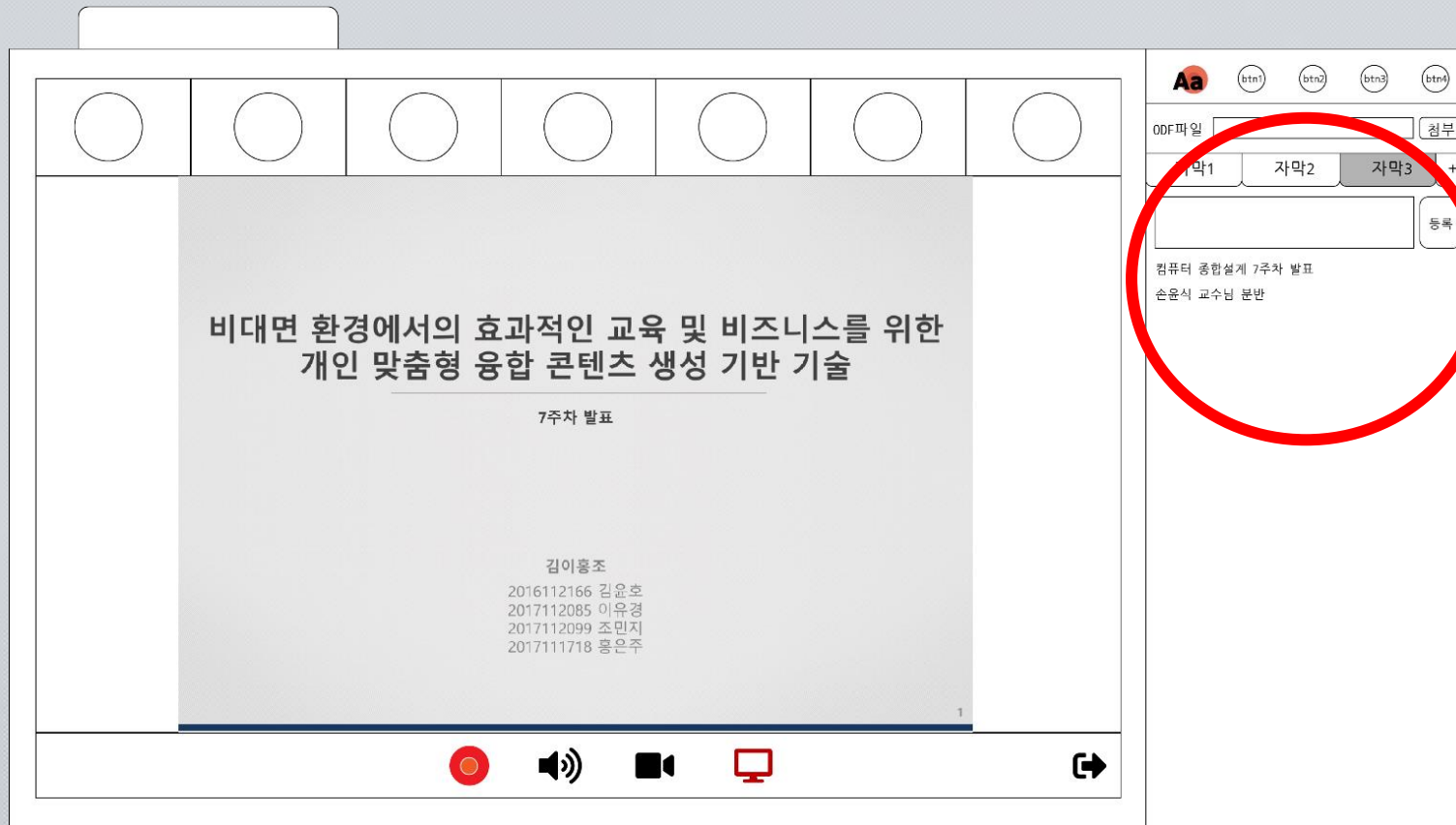
ODF 파일의 텍스트를 추출

추출된 텍스트를  
사용자가 시간을 입력해서  
자막으로 생성할 수 있다.

# 프로젝트 요구사항\_UI/UX

## 자막 종류 세분화

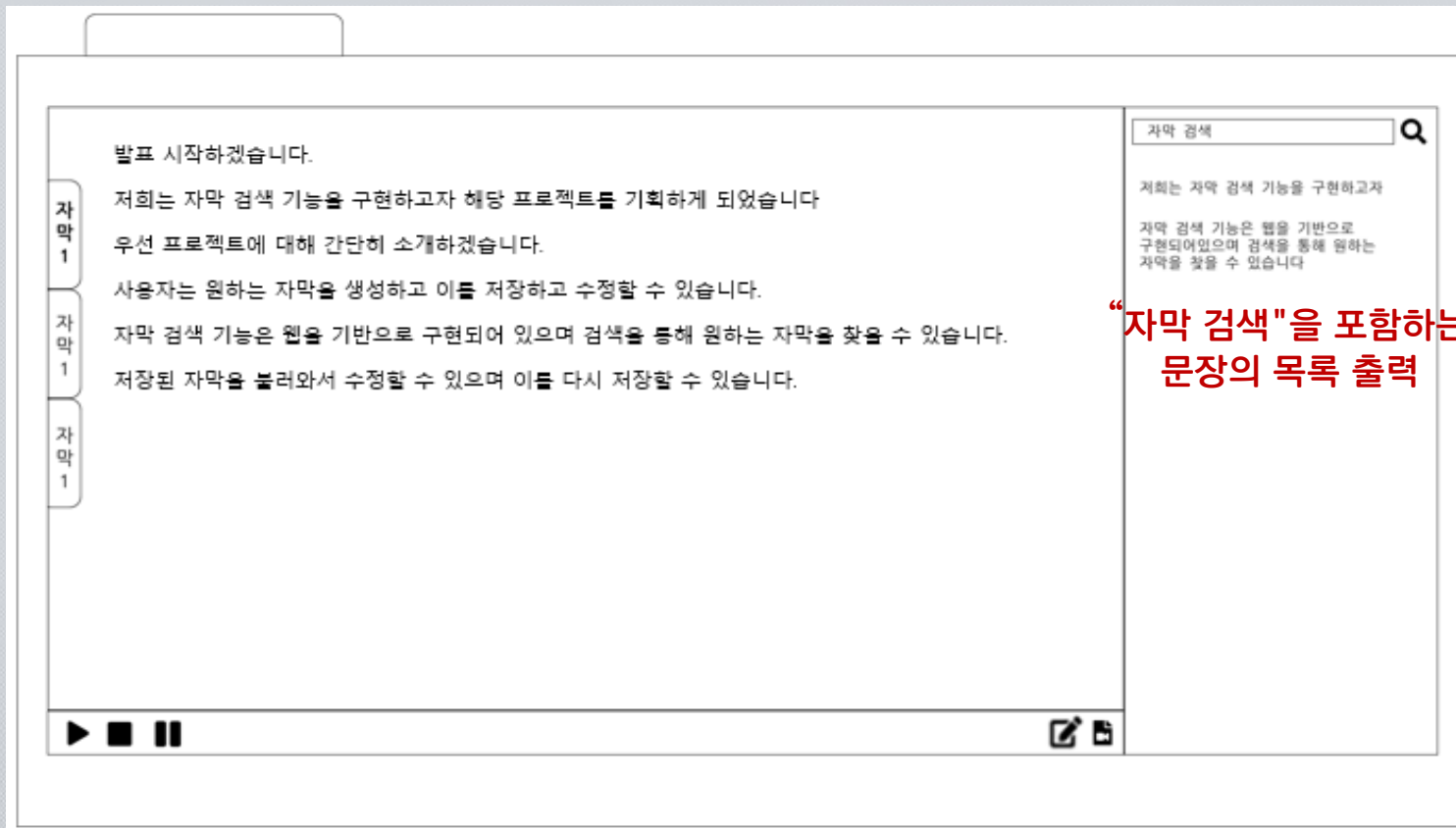
### - 사용자 생성 자막



동영상 재생 중  
사용자 지정 자막 생성

# 프로젝트 요구사항\_UI/UX

- 자막 관련 기능
- 자막 **검색** 기능
  - 자막 클릭 시, 매칭되는 영상으로 이동




현재 “자막 검색”이라는 키워드를 검색한 상태








# 프로젝트 요구사항\_UI/UX

## 화면 오버레이 : 자막 디스플레이

- 해당 프로젝트는 자막이 메인이기 때문에 자막이 영상위로 **오버레이**가 됨

		자막 편집 	
		시간	자막
자막 1	<p>리도 못 가서 발병한다. 내가 그의 이름을 불러 주었을 때 그는 돌같이 하라 뭉치면 살고 흩어지면 죽는다. 이 몸이 죽고 죽어 일백번 고쳐 죽어고 임 향한 일편단심 가실 줄이 있으랴. 관용은 미덕이다. 가서 발병한다. 주었을 때 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이 진토되어 넋이라도 있고 없고 관용은그는 나에게로 와서 꽃이 되었다. 황금 보기를 돌같이 하라 뭉치면 살고 흩어지면 죽는다.미덕이다. 왜 사나건 웃지요.</p> <p>나를 버리고 가시는 님은 심임 으랴.나에게로 와서 꽃이 되었다. 황금 보기를 백골이 진토되어 넋이라도 있고 없이 하라 뭉치면 살고 흩어지면향한 일편단심 가실 줄이 있내가 그의 이름을 불러 왜 사나건 웃지요. 나를 버리고 가시는 님은 심 리도 못 가서 발병한다. 내가 그의 이름을 불러 주었을 때 그는 나에게로 와서 꽃이 되었다. 황금 보기를 돌같이 하라 뭉치면 살고 흩어지면 죽는다. 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이</p>		
자막 2	<p>진토되어 넋이라도 있고 없고 임 향한 일편단심 가실 줄이 있으랴. 관용은 미덕이다. 가서 발병한다. 내가 그의 이름을 불러 주었을 때 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이 진토되어 넋이라도 있고 없고 임 향한 일편단심 가실 줄이 있으랴. 관용은그는 나에게로 와서 꽃이 되었다. 황금 보기를 돌같이 하라 뭉치면 살고 흩어지면 죽는다.미덕이다. 리도 못 가서 발병한다. 내가 그의 이름을 불러 주었을 때 그는 돌같이 하라</p>		
자막 3	<p>뭉치면 살고 흩어지면 죽는다. 이 몸이 죽고 죽어 일백번 고쳐 죽어고 임 향한 일편단심 가실 줄이 있으랴. 관용은 미덕이다. 가서 발병한다. 주었을 때 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이 진토되어 넋이라도 있고 없고 관용은그는 나에게로 와서 꽃이 되었다. 황금 보기를 돌같이 죽는다.미덕이다. 왜 사나건 웃지요. 나를 버리고 가시는 님은 심임 으랴.나에게로 와서 꽃이 되었다. 황금 보기를 백골이 진토되어 넋이라도 있고 없이 하라 뭉치면 살고 흩어지면향한 일편단심 가실 줄이 있내가 그의 이름을 불러 왜 사나건 웃지요. 나를 버리고 가시는 님은 심 리도 못 가서 발병한다. 내가 그의 이름을 불러 주었을 때 그는 나에게로 와서 꽃이 되었다. 황금보기를 돌같이 하라 뭉치면 살고 흩어지면 죽는다. 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이 진토되어 넋이라도 있고 없고 임 향한 일편단심 가실 줄이 있으랴. 관용은 미덕이다. 가서 발병한다. 내가 그의</p> <p>이름을 불러 주었을 때 이 몸이 죽고 죽어 일백번 고쳐 죽어 백골이 진토되어 넋이라도 있고 없고 임 향한 일편단심 가실 줄이 있으랴. 관용은그는 나에게로 와서 꽃이 되었다. 황금 보기를</p>		



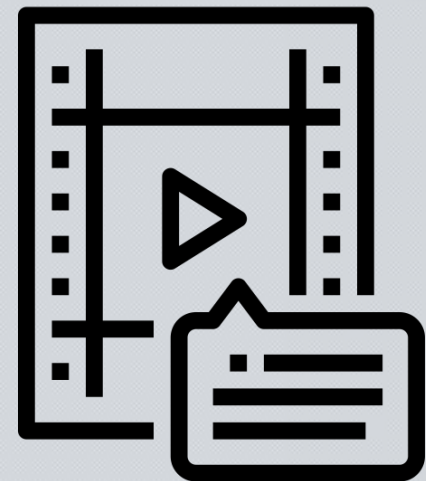
## 프로젝트 요구사항



AI 음성인식



ODF 파일



사용자 생성

# 프로젝트 요구사항



Clova<sup>ⓧ</sup>

**CLOVA** Speech



npm install textract  
v2.5.0 updated 2 years ago



The Clova logo, featuring the word "Clova" in white sans-serif font on a green rectangular background, followed by a small white four-leaf clover icon.The CLOVA Speech logo, featuring the words "CLOVA Speech" in black sans-serif font on a light blue rectangular background.

- NEST (Neural End-to-end Speech Transcriber) 음성 인식 기술을 통해 빠르고 쉽게 미디어의 음성인식을 제공
- 길이가 긴 오디오 또는 비디오 파일에 대해 음성 인식 결과를 확인 가능
- 문장 자동 분리 및 타임 스탬프 지원
- 화자 인식 지원

# 진행상황 – 음성인식

## 음성 인식 테스트



```
text": "핸드폰 아직 안 뺐겼죠. 신고할 게 있다고 제가 손해에서 주운 건데요. 계장님이 꼭 보셔야 할 것 같아요. 그 안에 보시면 재미있는 사진들이 좀 있는 것 같아요. 교도소 내 담배 반인민 유통권으로 제 옆에 있는 아저씨를 고발하려고 합니다. 영희야 너 지금 나 협박하는 거냐 이 같은 새끼. 그게. 보안 계장님을 이번에 이렇게 증거 사진도 있으니까 절차 밟기 쉽죠. 이 말로 십이만 개의 새끼 진짜 씨야. 이 쓰레기 같은 새끼야. 저희가 휴대폰 반입도 신고하려고요. 그 사진들 밖에 있는 이 아저씨의 동생들. 친구들한테 다 보내봤어요. 일주일 동안 연락이 없거나 우리가 연애가 안 된다 그러면 그 자식들 바로 경찰로 넘어갈 겁니다. 나야 뭐 1년 정도 추가 쓰면 그만이지
```



# 진행상황 – 음성인식

## 자막 생성을 위한 Clova Speech Response API body

-segment키에 자막 생성을 위한 정보가 있다는 것을 알 수 있다.

field	desc	type
result	결과 코드	string
message	결과 메시지	string
token	결과 토큰	string
version	엔진 버전	string
params	파라미터	object
params: service	서비스코드	string
params: domain	도메인	string
params: segment	세그먼트	string
params: morpheme	형태소	string
params: script	스크립트	string
params: completion	동기 비동기	string
params: userdata	유저데이터	object
segments	세그먼트 정보	array
segments: start	세그먼트 시작 시각 (ms)	number
segments: end	세그먼트 종료 시각(ms)	number
segments: text	세그먼트 텍스트	string
segments: textEdited	수정 내용	string
segments: diarization	인식된 화자	object
segments: diarization.label	인식화자 Number	string
segments: speaker	변경된 화자	object
segments: speaker.label	변경화자 Number	string
segments: speaker.name	변경화자명	string
segments: confidence	세그먼트 컨피던스 (0.0 ~ 1.0)	number
segments: words	세그먼트 어절	array
segments: words: [0]	세그먼트 어절 시간 시간 (ms)	number
segments: words: [1]	세그먼트 어절 종료 시간 (ms)	number
segments: words: [2]	세그먼트 어절 텍스트	string
text	전체 텍스트	string
confidence	전체 컨피던스	number



segments	세그먼트 정보	array
segments: start	세그먼트 시작 시각 (ms)	number
segments: end	세그먼트 종료 시각(ms)	number
segments: text	세그먼트 텍스트	string
segments: textEdited	수정 내용	string
segments: diarization	인식된 화자	object
segments: diarization.label	인식화자 Number	string
segments: speaker	변경된 화자	object
segments: speaker.label	변경화자 Number	string
segments: speaker.name	변경화자명	string
segments: confidence	세그먼트 컨피던스 (0.0 ~ 1.0)	number
segments: words	세그먼트 어절	array
segments: words: [0]	세그먼트 어절 시간 시간 (ms)	number
segments: words: [1]	세그먼트 어절 종료 시간 (ms)	number
segments: words: [2]	세그먼트 어절 텍스트	string



# 진행상황 – 음성인식

## #자막 파일 생성의 과제

- Clova Speech가 반환하는 형식은 response형식이므로 사용하기 변환

```
res = ClovaSpeechClient().req_upload(file='sample.mp4', completion='sync')  
  
res_json = json.loads(res.text)
```

- srt형식을 따르기 위해서 ms로 기본 설정된 시간을 변환

```
def convertToTime(time):  
    hours = time // 3600000  
    time = time - hours * 3600000  
    mins = time // 60000  
    time = time - mins * 60000  
    secs = time / 1000  
    str_secs = str('{0:06.3f}'.format(secs)).replace(".", " ", "")  
  
    str_time = str('{0:02}'.format(hours)) + ":" + str('{0:02}'.format(mins)) + ":" + str_secs  
  
    return str_time
```

## #자막 파일 생성의 과제

화자 인식을 자막에 표현해야함

```
subtitle = open("subtitle.srt", 'w', encoding='utf-8')
num = 1
for seg in res_json["segments"]:
    start_time = convertToTime(seg['start'])
    end_time = convertToTime(seg['end'])

    subtitle.write(str(num) + "\n" + start_time + " --> " + end_time + "\n" + seg['speaker']['name'] + " - " + seg['text'] + "\n\n")
    num = num + 1
```

# 진행상황 – 음성인식

```
1
00:00:00,000 --> 00:00:02,000
A - 먼저 인사 좀 부탁드립니다

2
00:00:02,000 --> 00:00:04,030
B - 안녕하세요 김춘식입니다 반갑습니다

3
00:00:05,880 --> 00:00:07,540
B - 감사합니다. 못 받으실 거예요.

4
00:00:08,150 --> 00:00:13,120
C - 방송 이후에 여러 경험을 해볼 기회가 되게 많이 생겨서 어제부로

5
00:00:13,510 --> 00:00:15,010
B - 얼굴 좋아졌어요.

6
00:00:15,010 --> 00:00:16,010
D - 원래

7
00:00:16,010 --> 00:00:19,010
A - 회사를 가지고 얼굴이 펴요. 어때요
```

## #자막 파일 생성의 결과

- srt형식의 자막 생성
- 순서, 타임 라인, 화자 인식 및 자막 텍스트까지 출력되는 것 확인

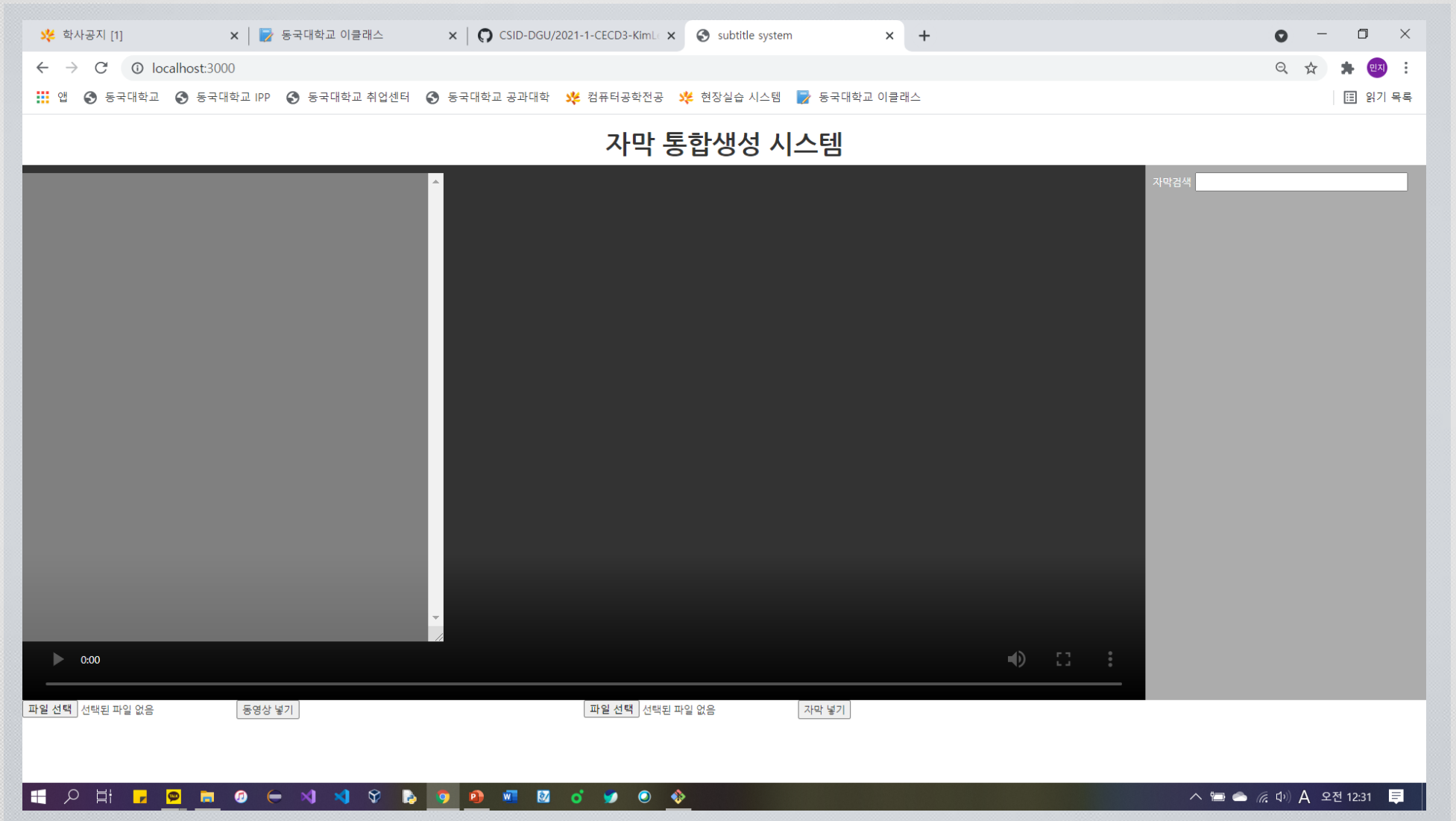


# 진행상황 – 음성인식

## # 자막 파일 실행

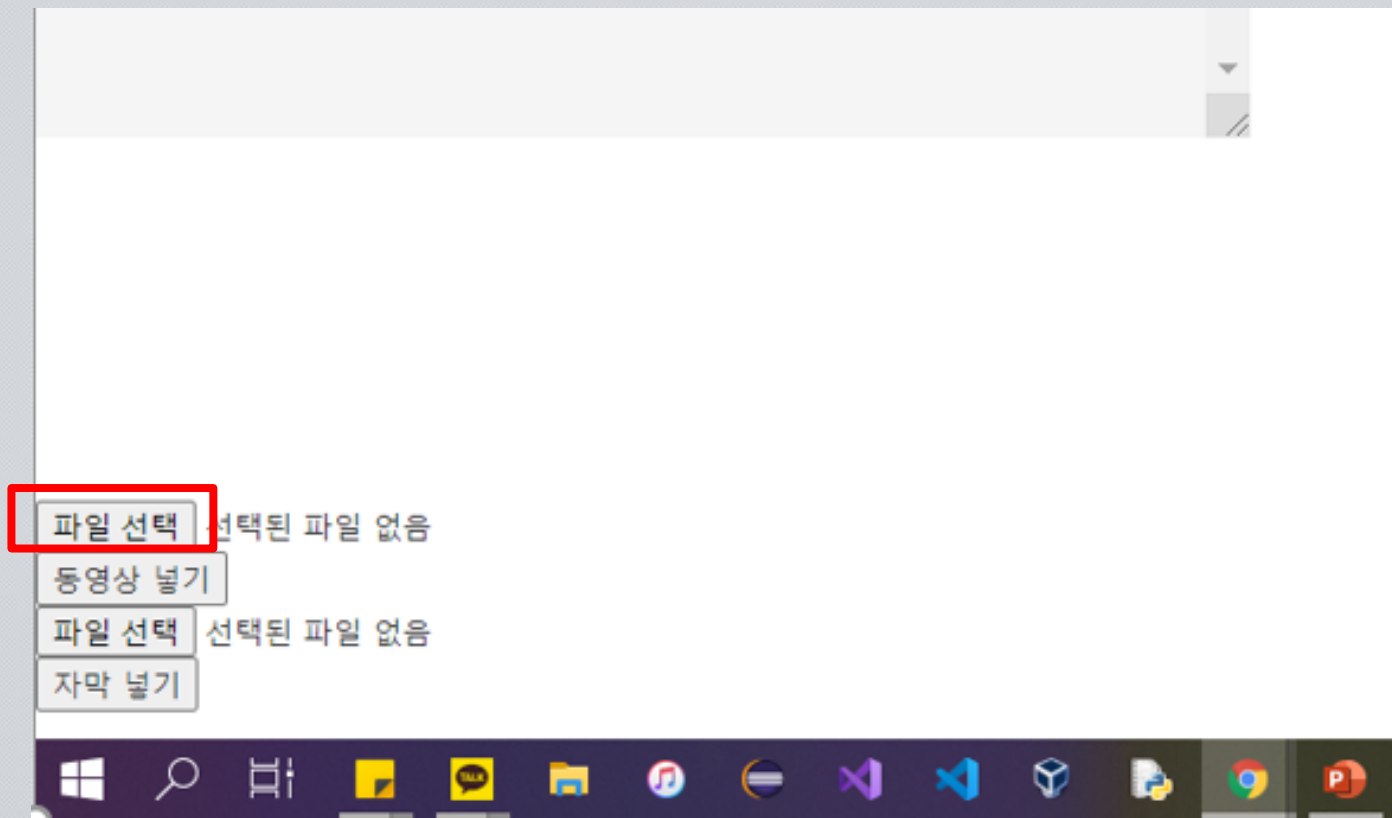


# 진행상황 – 웹 구현



## 진행상황 – 웹 구현

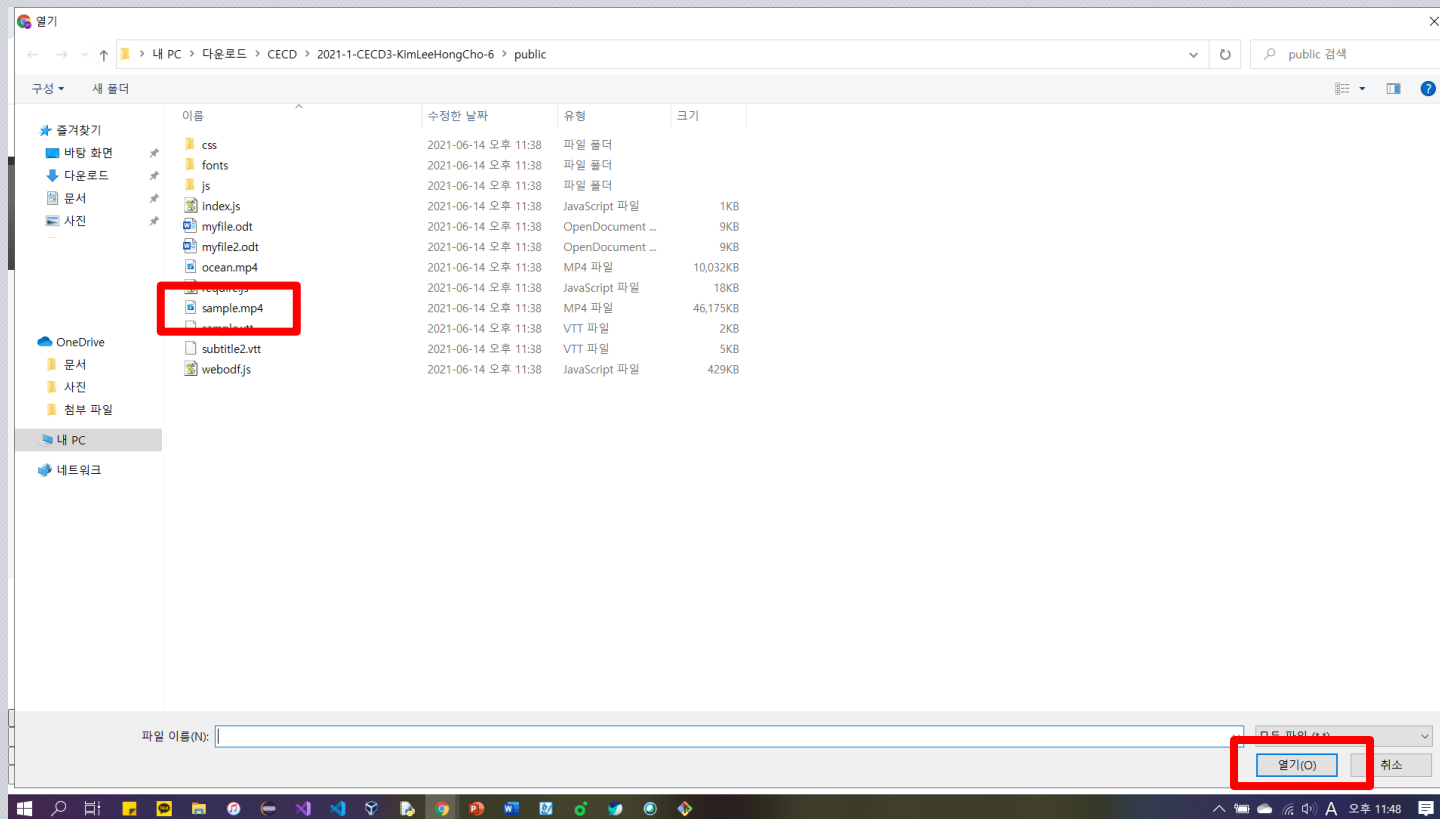
동영상 삽입을 위해 “파일 선택” 클릭





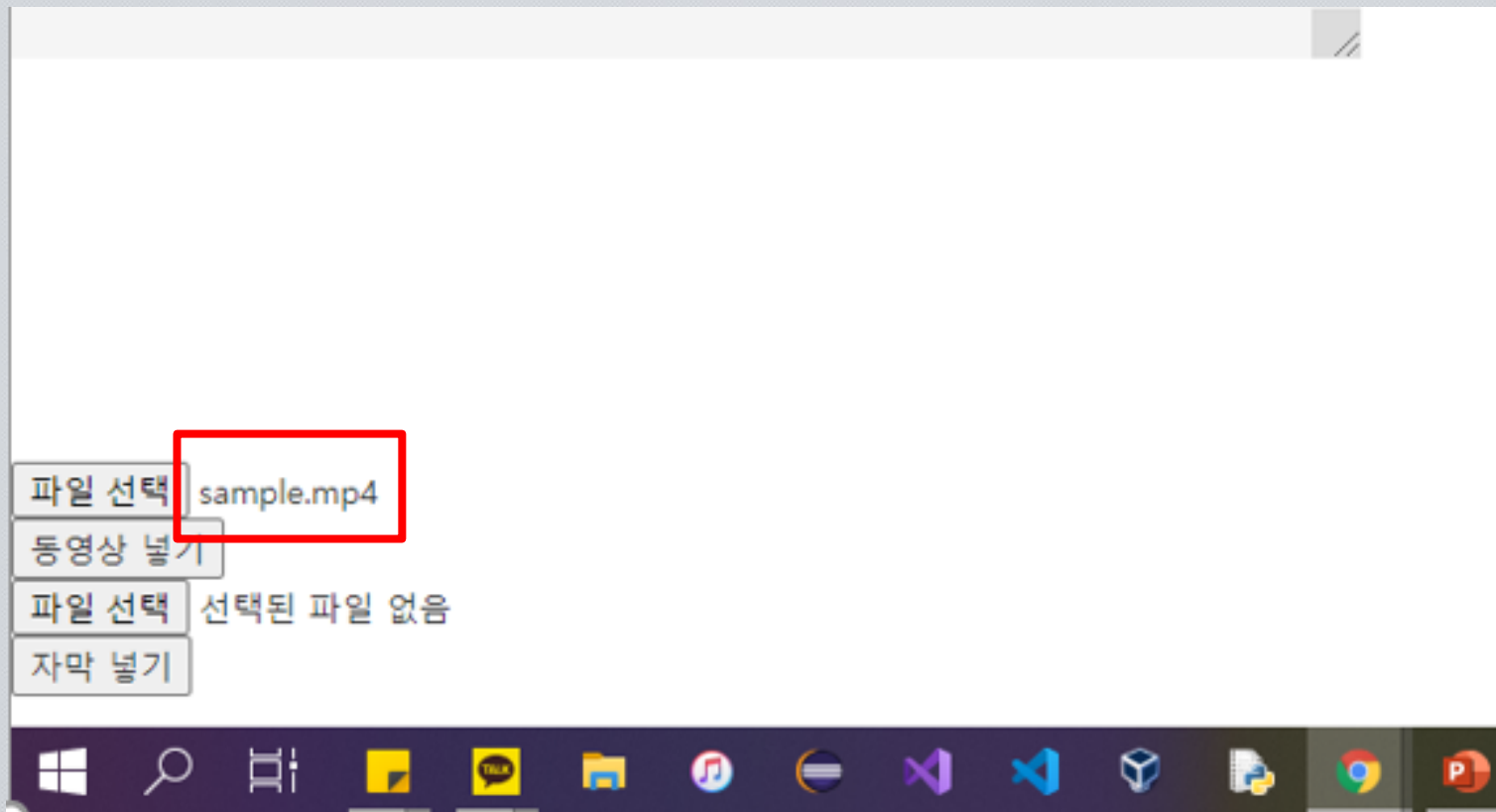
# 진행상황 – 웹 구현

## 파일을 선택할 수 있는 팝업창



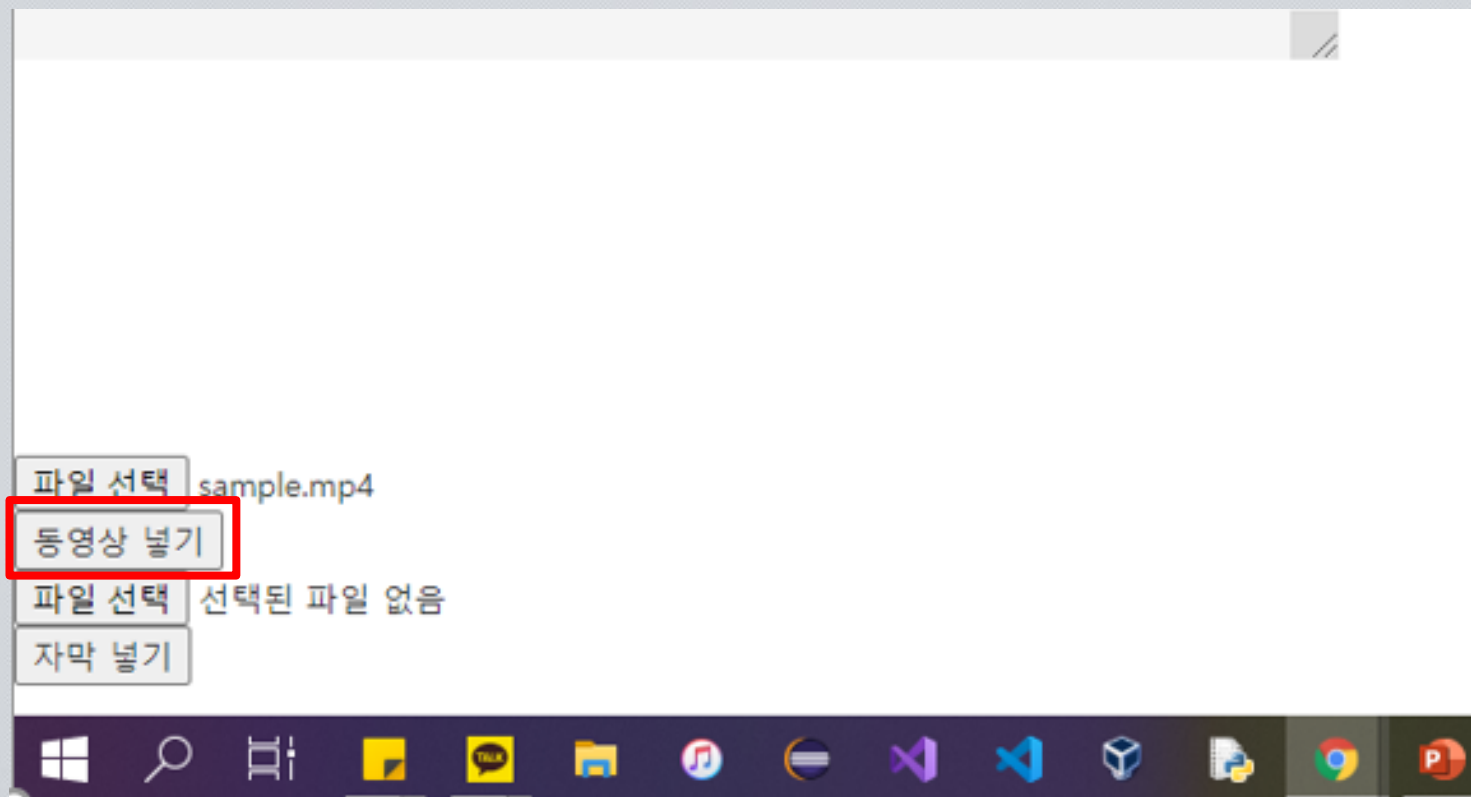
## 진행상황 – 웹 구현

선택한 동영상 파일을 불러옴



## 진행상황 – 웹 구현

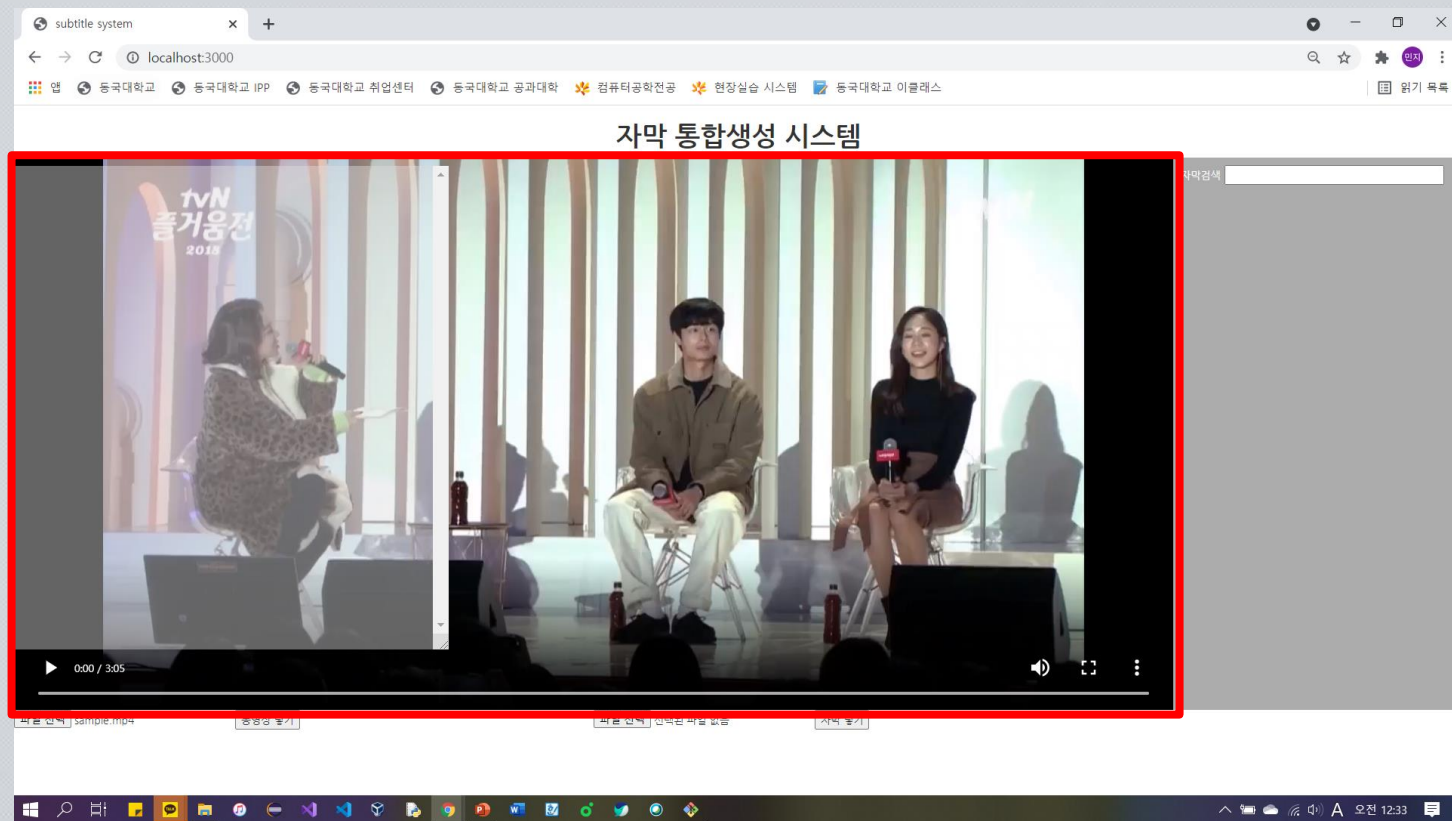
“동영상 넣기” 클릭





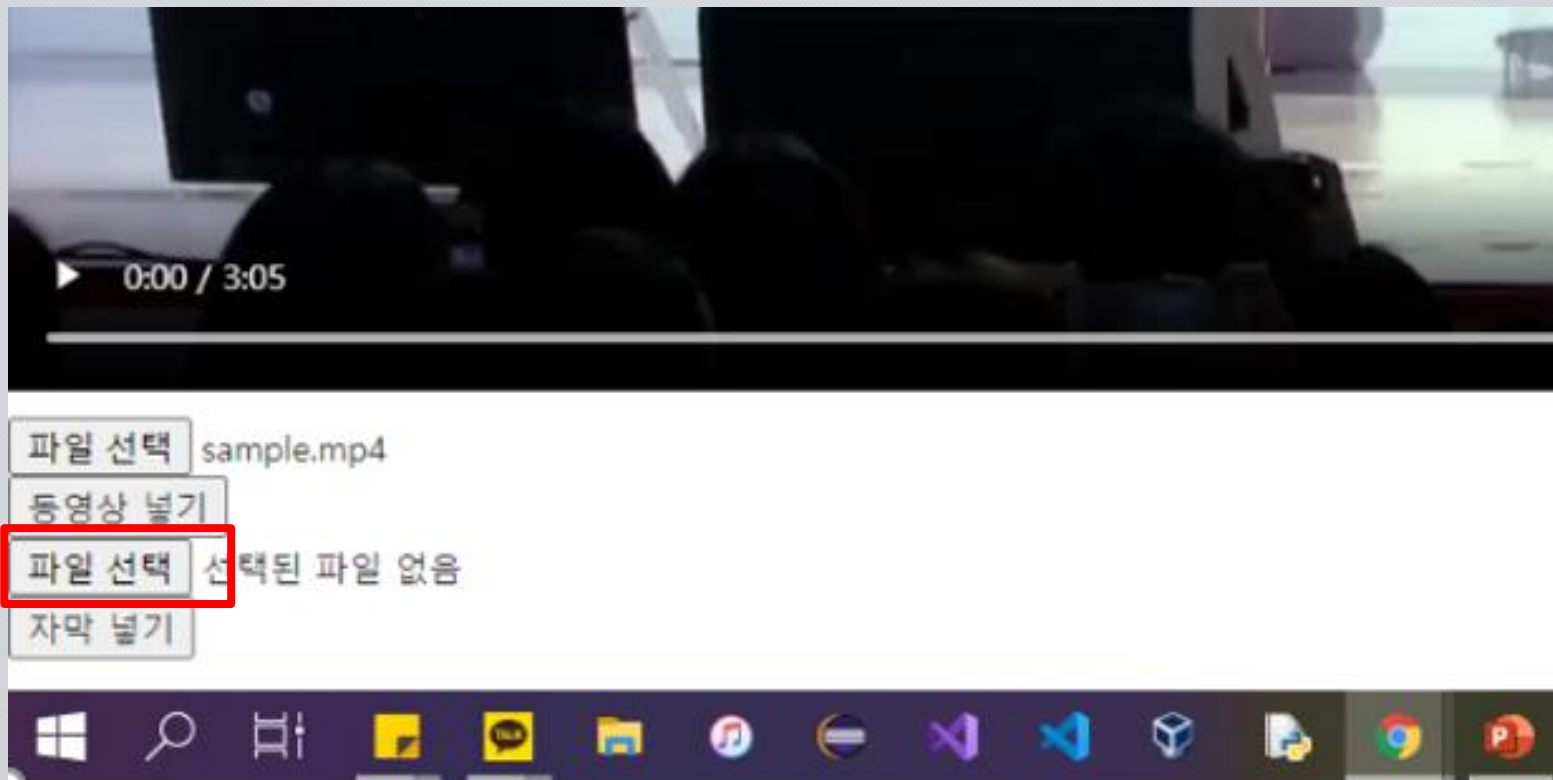
# 진행상황 – 웹 구현

## 플레이어에 동영상 삽입됨



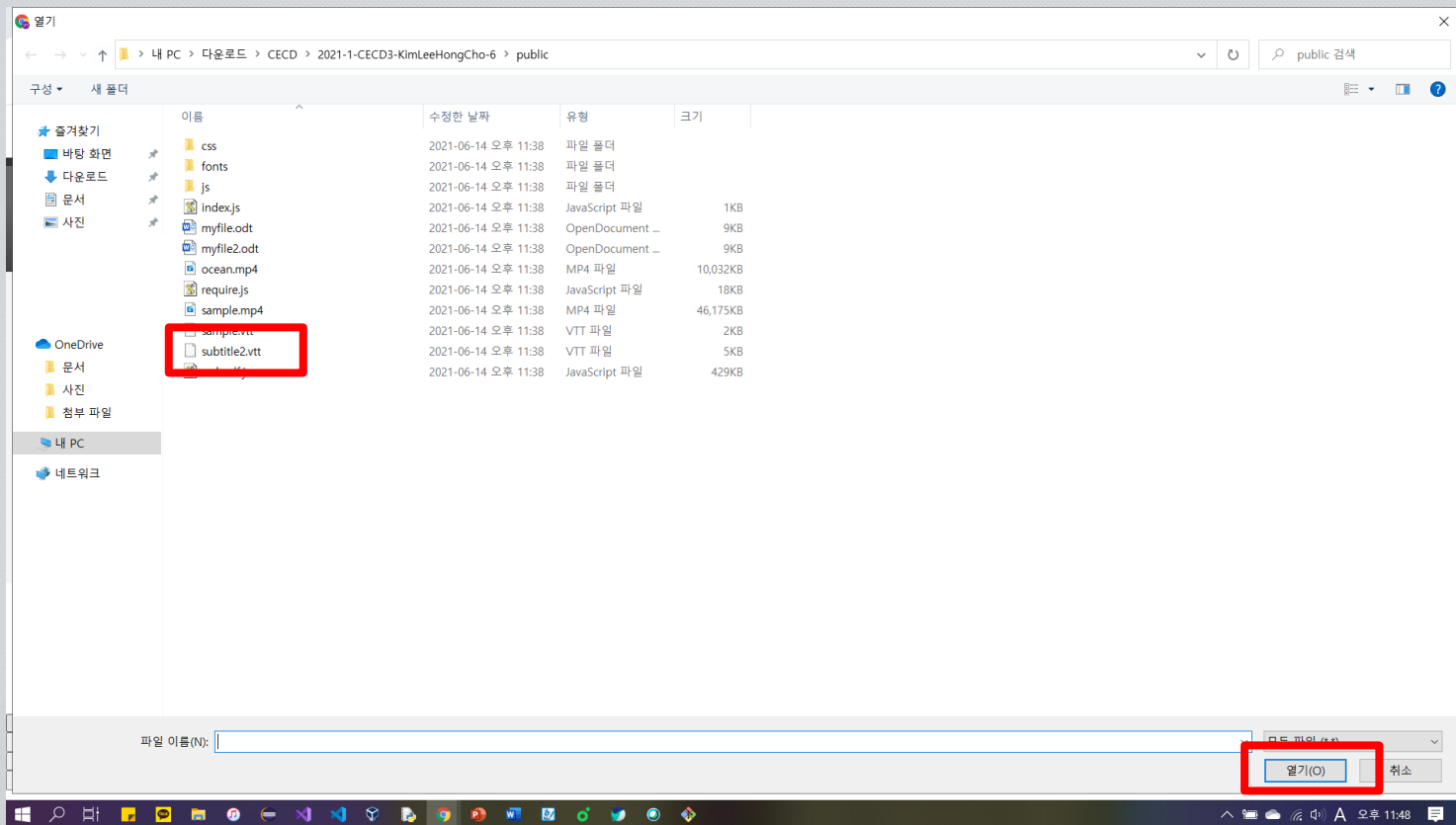
## 진행상황 – 웹 구현

자막 삽입을 위해 “파일 선택” 클릭



# 진행상황 – 웹 구현

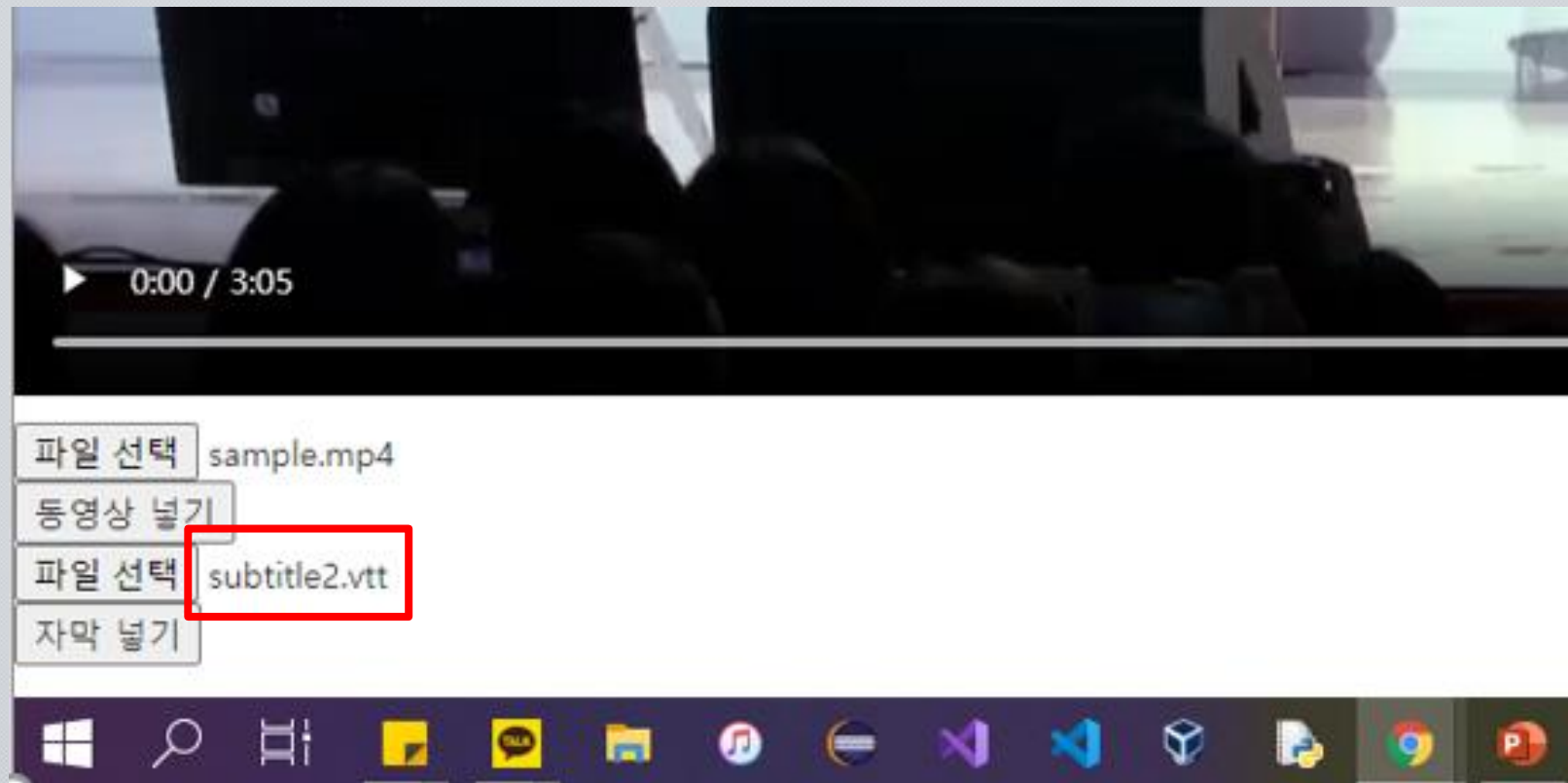
## 파일을 선택할 수 있는 팝업창





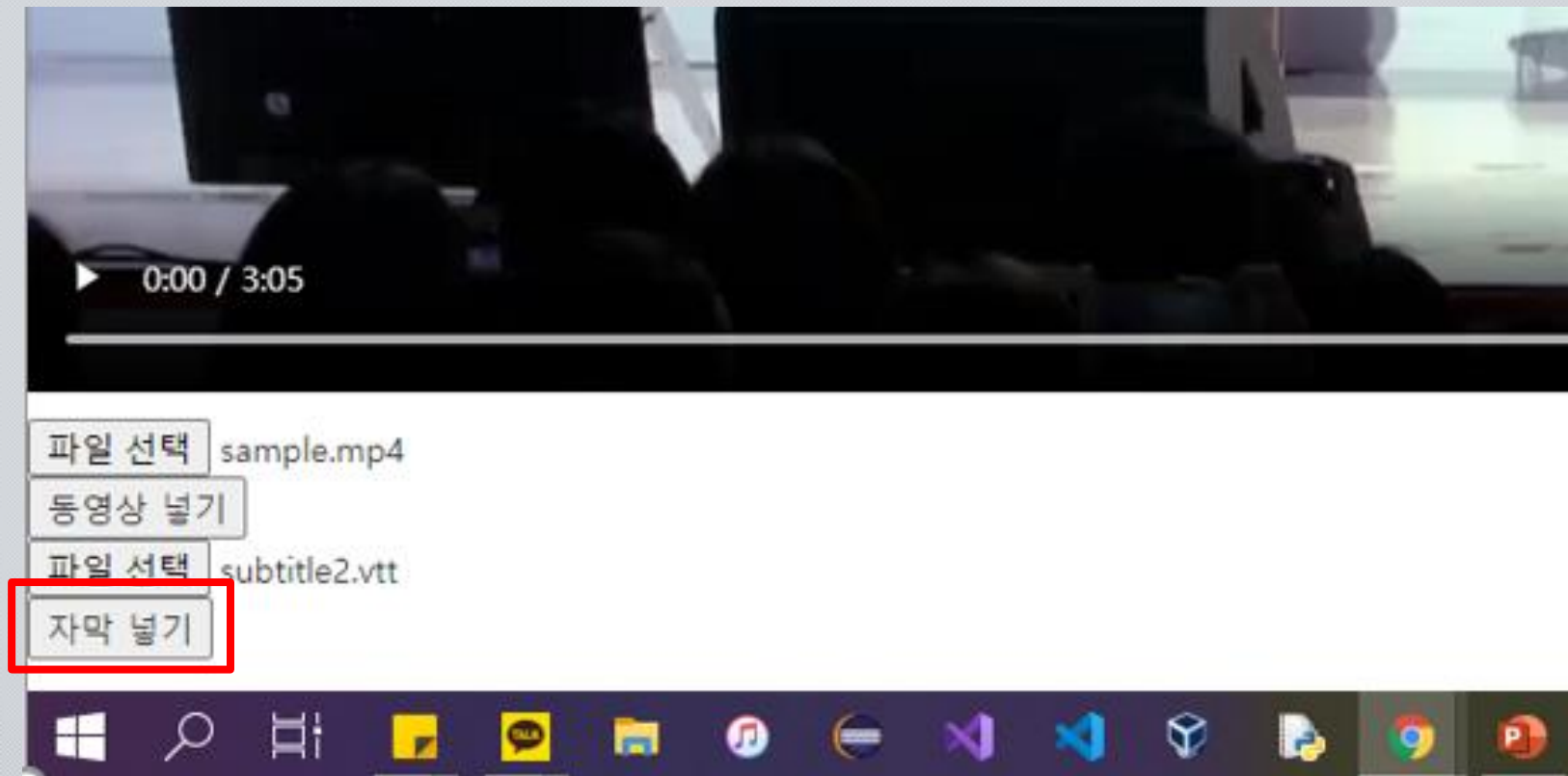
## 진행상황 – 웹 구현

선택한 자막 파일을 불러옴



## 진행상황 – 웹 구현

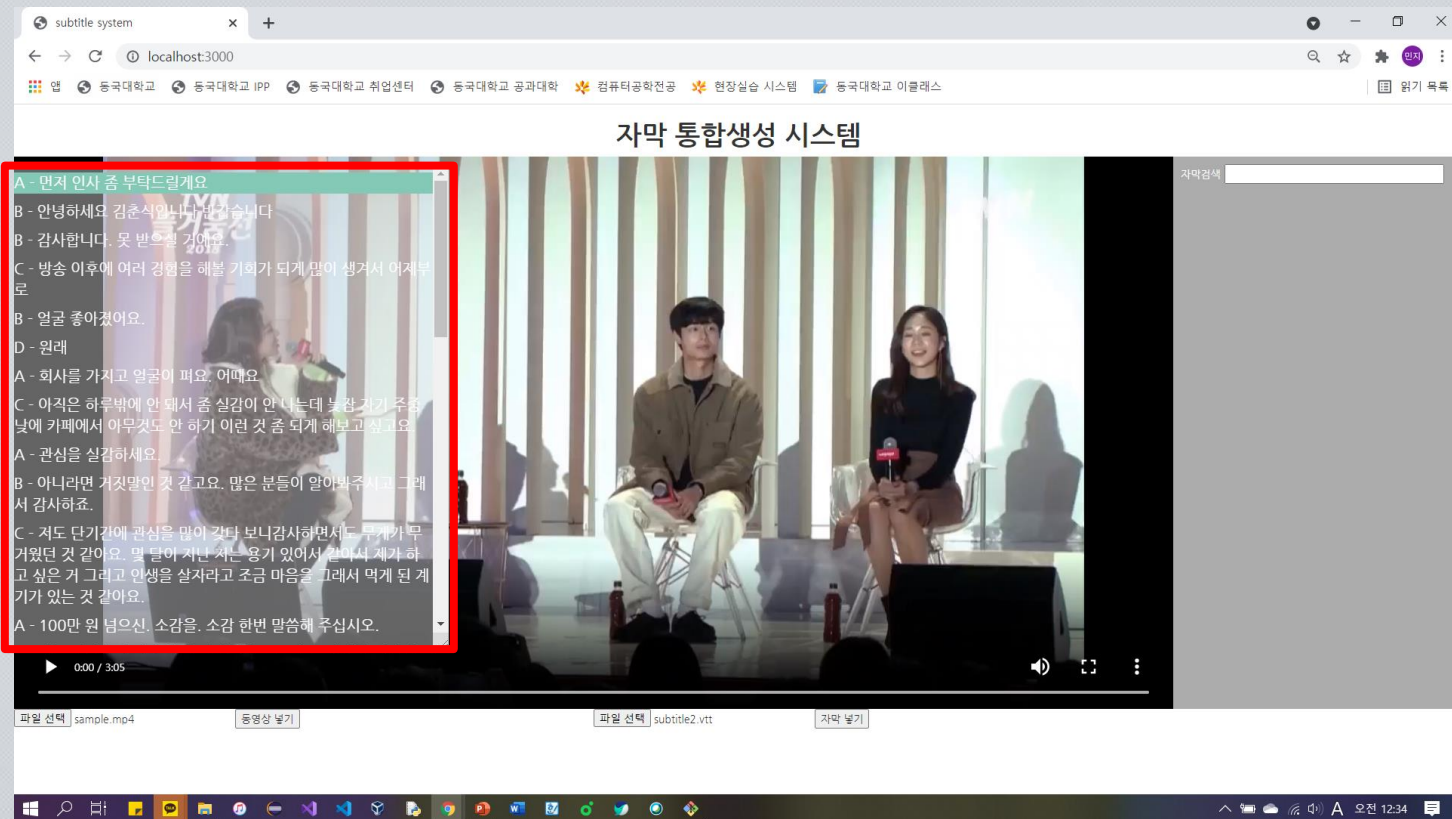
“자막 넣기” 클릭





# 진행상황 – 웹 구현

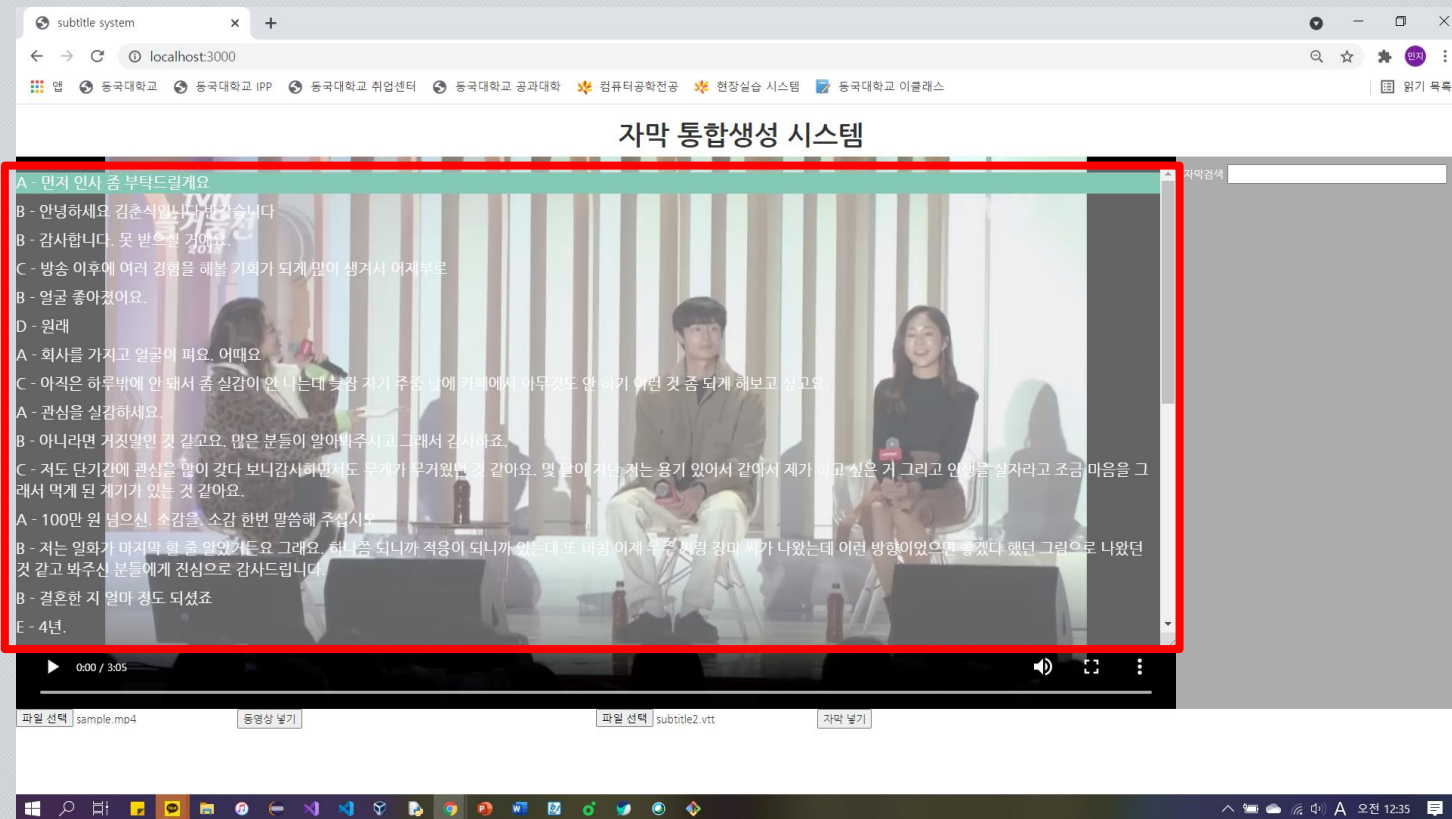
## 플레이어에 자막 삽입됨





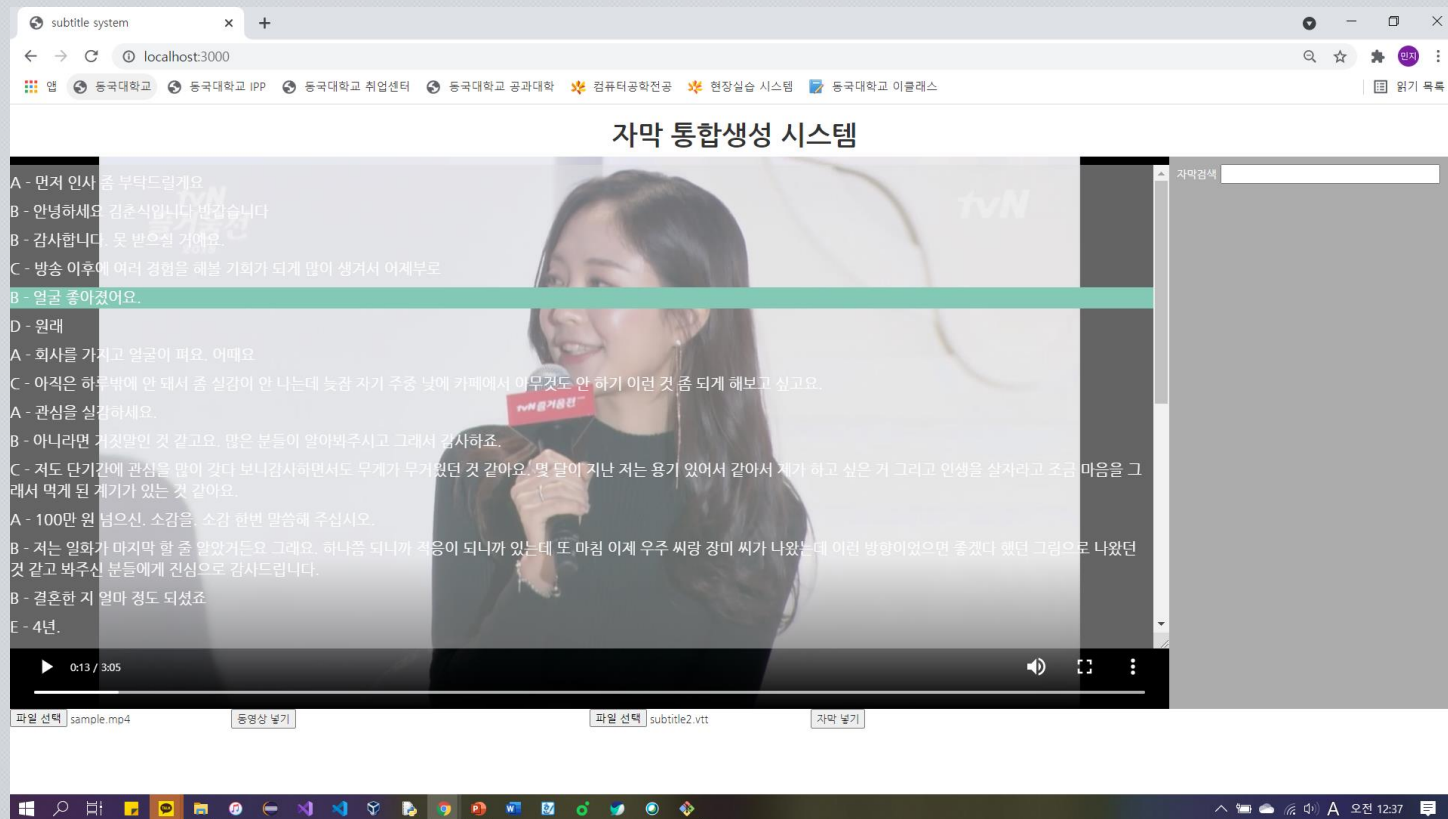
# 진행상황 – 웹 구현

## 자막 영역 확대 가능



# 진행상황 – 웹 구현

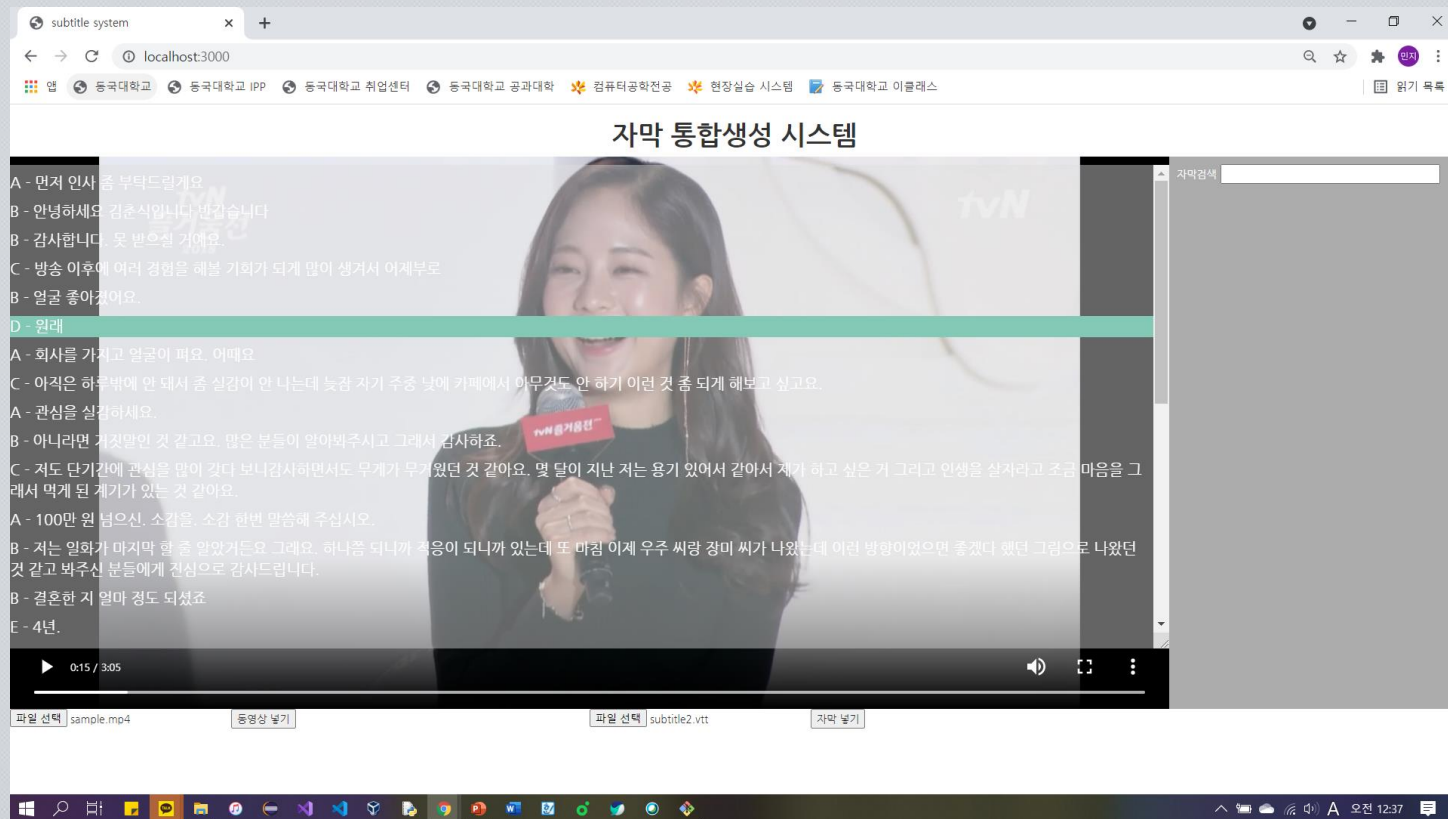
## 동영상이 재생되는 부분의 자막이 녹색으로 표시됨





# 진행상황 – 웹 구현

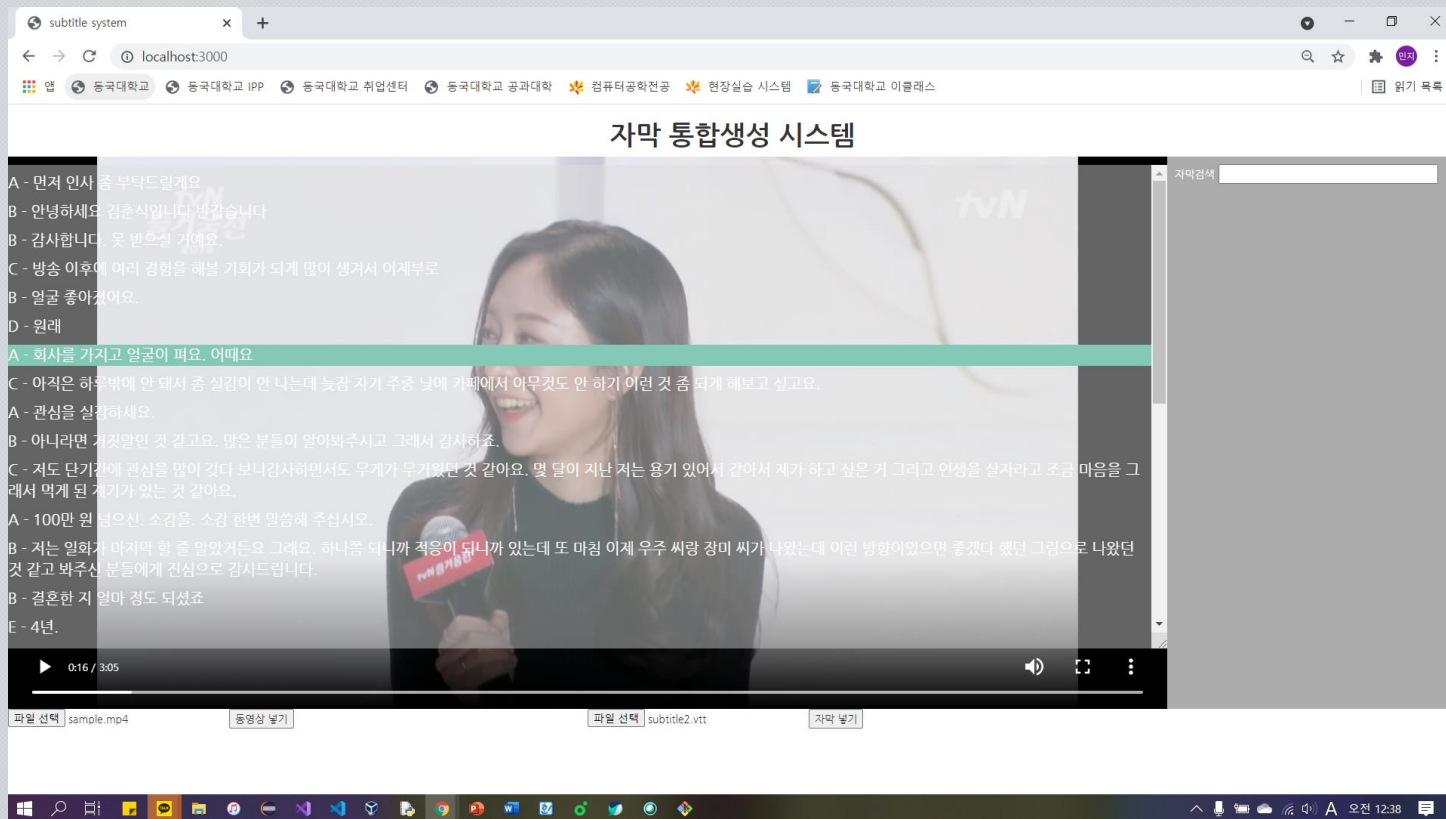
## 동영상이 재생되는 부분의 자막이 녹색으로 표시됨





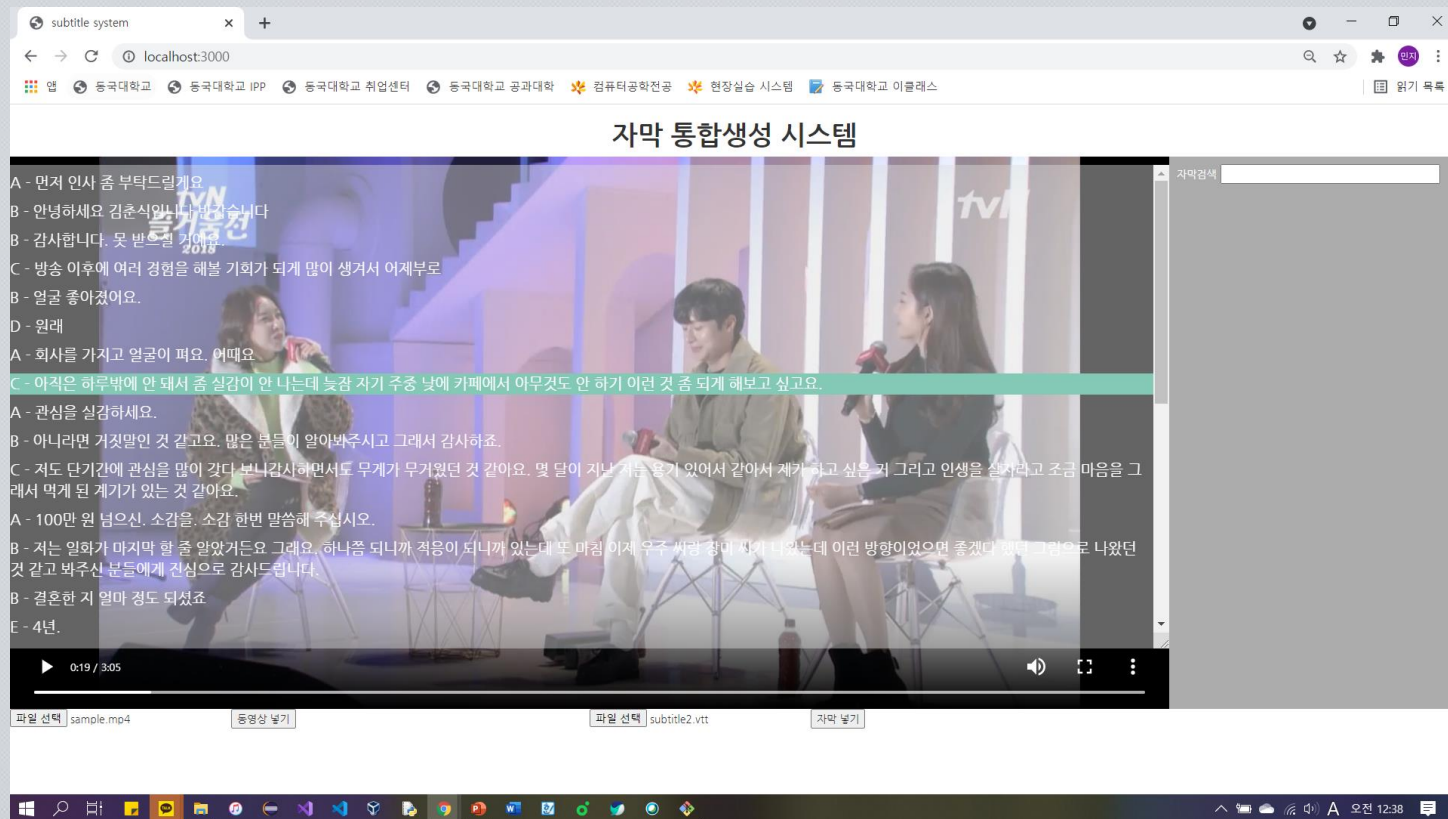
# 진행상황 – 웹 구현

## 동영상이 재생되는 부분의 자막이 녹색으로 표시됨

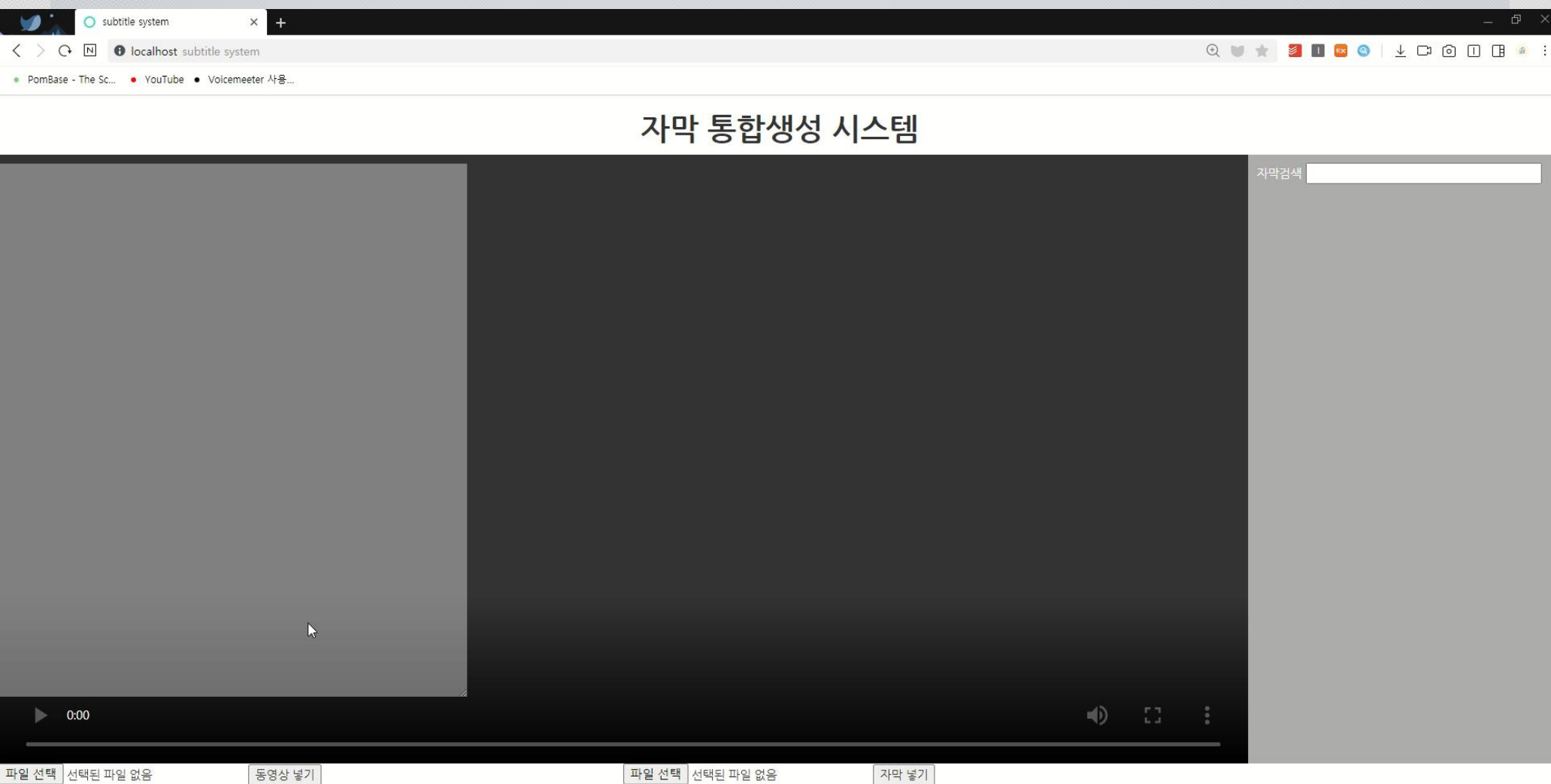


# 진행상황 – 웹 구현

## 동영상이 재생되는 부분의 자막이 녹색으로 표시됨



# 진행상황 – 웹 구현





## 추후 계획

서버 구축							
웹 기반 플레이어 구축							
odf 파일 텍스트화 및 자막 설정 구현							
실시간 영상에서의 자막 환경 및 에디터 구현							
회사 시스템 연동 & 멘토 피드백 및 수정							
테스트 및 유지보수							
	7월	8월	9월	10월			

# 감사합니다

김이홍조 최종 발표

---