

열정으로 무장한 미래 연대생 이정아입니다.

1999 - 2020 PORTFOLIO

개인 프로젝트 및 학습자료 보관

<https://github.com/JungA-lee?tab=repositories>

<https://blog.naver.com/dlwjddk218>





꿈꾸는 대학생 이정아

“안되면 되게 해라”
수단과 방법을 가리지 않고 과제를 해내는 능력을
가지고 있습니다.

이정아 / JungA Lee

1999.02.18 / 서울특별시

Blog

<https://blog.naver.com/dlwjddk218>

GitHub

<https://github.com/JungA-lee?tab=repositories>

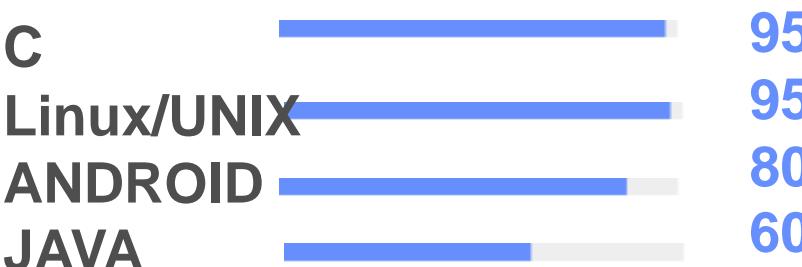
GRADUATION

2017 백석고등학교 졸업

2017 동양미래대학교 컴퓨터소프트웨어과 입학

2021 연세대학교 컴퓨터공학과 입학 예정

SKILL



AWARDS

2019 임베디드 경진대회 입상

2019 연수구 멘토링 프로그램 위촉장

2020 현장실습 수기 공모전 입상

2020 졸업작품 경진대회 수상 예정

PROJECT

2019 소켓 채팅 서버 프로젝트

2019 화성 탐사로봇 프로젝트

2020 구조체 정보 출력 서버 프로젝트

2020 AWS CRUD API 프로젝트

2020 VR 언어 게임 프로젝트

교외 활동 실적

[삼성SDS-JA Korea] Coding Campus

대학생 봉사단

하루 평균 30명이 드나드는
프로그래밍 블로거

Github Arctic Code Vault Contributor

뱃지 수여자

이정아는 어떤 능력을 가지고 있을까?

전공에 대한 열정

전공에 대한 열정으로
경진대회 수상, 봉사활동 외
팀프로젝트의 경험 다수



밝은 영향력

긍정적이며 사교성이 좋아 동기들을
모아 대외활동을 하는 학생



프로그래밍 스킬

프로그래밍에 대한 높은 이해도!
회사의 외주를 받아 프로젝트 진행 중



적응이 빠른 학생

어디에서나 빠르게 적응하며
주변사람들과 원만한 관계를 유지



PROJECT.1

소켓 채팅 서버
흥미로워서 혼자
만들었더니 다음 학기 A+

ABOUT PROJECT

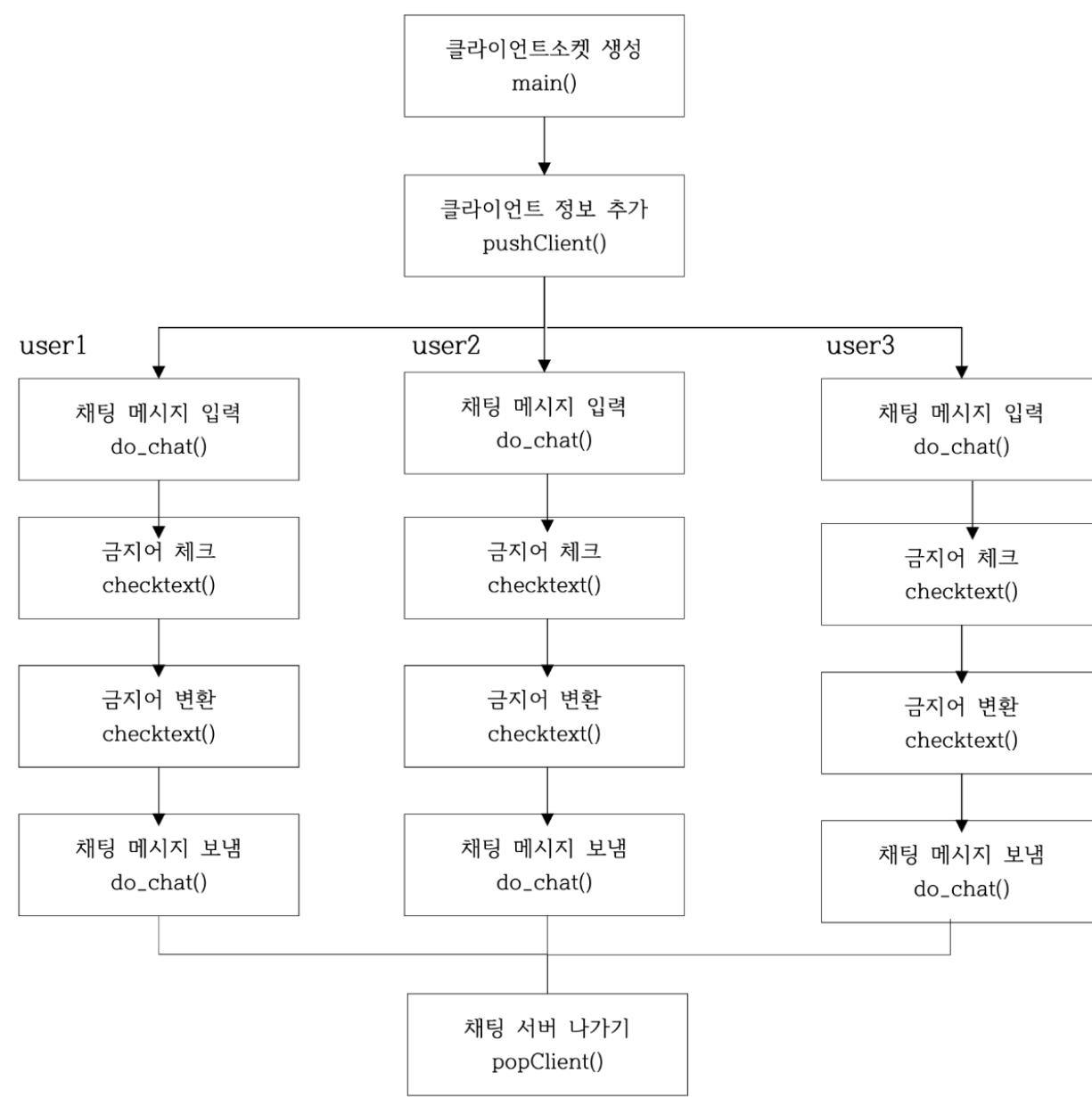
TCP/IP 소켓을 통한 채팅 서버입니다.
소켓 프로그래밍 수업에서 만들 서버가 흥미로워
기능을 추가했습니다.

다음학기 기말 시험 대체 과제로 제출했더니
A+이라는 성적을 받았습니다.

TCP/IP 채팅 서버 개요

개인 프로젝트 활동

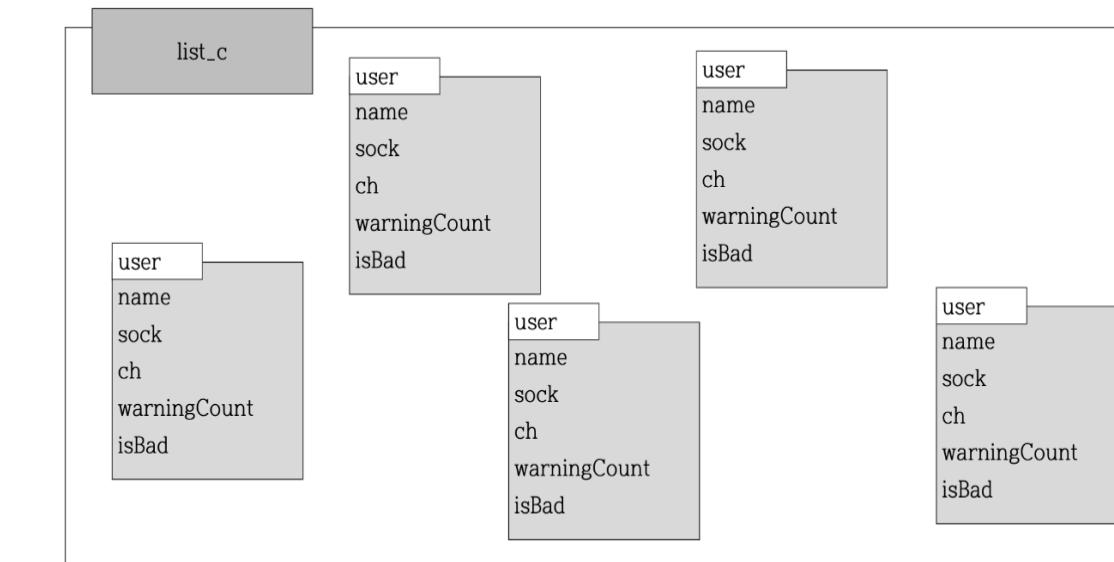
채팅 서버 구조



채팅 서버 기능

기능	설명.
chanel : [number]	채팅방을 선택하여 입장합니다.
/w [nickname] [message]	귓속말 기능으로 원하는 사용자에게만 메시지를 보냅니다.
/?	채팅방 기능과 사용법에 대해 알려줍니다.
****	욕설 입력 시 경고 횟수를 알려줍니다. 경고 3회 누적 시 채팅서버에서 퇴장 당합니다.
/userlist	현재 채팅 채널에 접속중인 사용자들의 닉네임을 출력합니다.
/nickname	현재 접속중인 클라이언트의 닉네임을 알려줍니다.
exit	채팅방이 종료됩니다.

클라이언트 기본 구조



기여도

기획

100

개발

100

TCP/IP 채팅 서버 산출물

```
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : cc
Input Channel : 2
Welcome to chatting room
help command > /?
/? : 
/w : 콤속말 대화 가능
/userlist : 채널 접속중인 사용자 보기
/nickname : 내 아이디 보기
exit : 나가기
**욕설누적 3회시 강퇴입니다.
채널을 바꾸시려면 채팅방 종료 후 재접속 해주세요
```

```
/nickname : 내 아이디 보기
exit : 나가기
**욕설누적 3회시 강퇴입니다.
채널을 바꾸시려면 채팅방 종료 후 재접속 해주세요

/nickname
your nickname : cc
```

```
파일(F) 편집(E) 템(T) 도움말(H)
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : jiyoon
Input Channel : 1
Welcome to chatting room
help command > /?
[hyunwoo] ****
[hyunwoo] ****
[hyunwoo] ****
```

```
Input Nickname : dongjae
Input Channel : 1
Welcome to chatting room
help command > /?
/userlist
[junga]
[dongjae]
[jiyoon]
```

```
파일(F) 편집(E) 템(T) 도움말(H)
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : hyunwoo
Input Channel : 1
Welcome to chatting room
help command > /?
fuck
비속어가 포함된 문장입니다.
욕설 누적 횟수 : 1회!!
[hyunwoo] ****
fuck
비속어가 포함된 문장입니다.
욕설 누적 횟수 : 2회!!
[hyunwoo] ****
fuck
비속어가 포함된 문장입니다.
욕설 누적 횟수 : 3회!!
욕설 누적 3회 강퇴입니다.
```

```
pi@raspberrypi:~ $ cd webserver/
pi@raspberrypi:~/webserver $ ./chatserver
[1] junga님이 1번 방에 접속하였습니다.
[2] dongjae님이 1번 방에 접속하였습니다.
[3] jiyoon님이 2번 방에 접속하였습니다.
[4] suhan님이 1번 방에 접속하였습니다.
```

```
pi@raspberrypi:~ $ cd webserver/
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : junga
Input Channel : 1
Welcome to chatting room
help command > /?
/w dongjae what r u doing ?
/w dongjae i miss u
```

```
pi@raspberrypi:~ $ cd webserver/
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : suhan
Input Channel : 1
Welcome to chatting room
help command > /?
```

```
pi@raspberrypi:~ $ cd webserver/
pi@raspberrypi:~/webserver $ ./chatclient
Input Nickname : suhan
Input Channel : 1
Welcome to chatting room
help command > /?
[junga] what r u doing ?
[junga] i miss u
```

PROJECT.2

화성탐사로봇 SNAKE BOT 임베디드 경진대회 입상작

ABOUT PROJECT

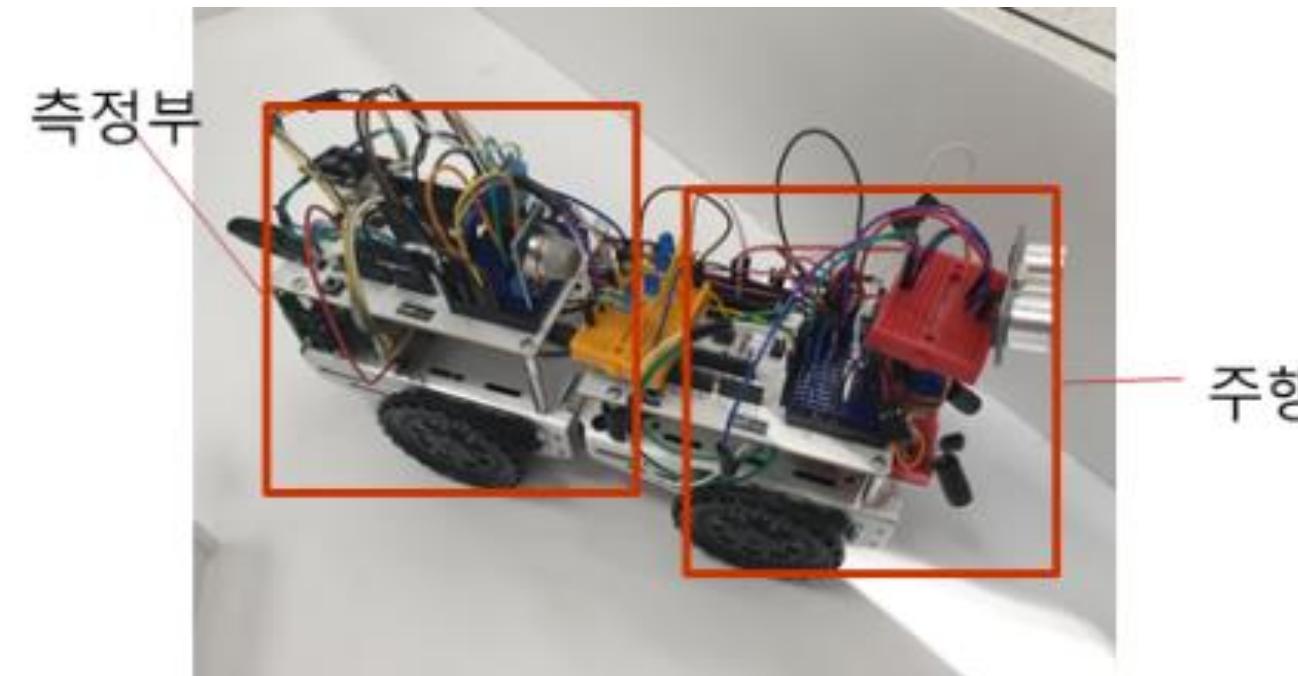
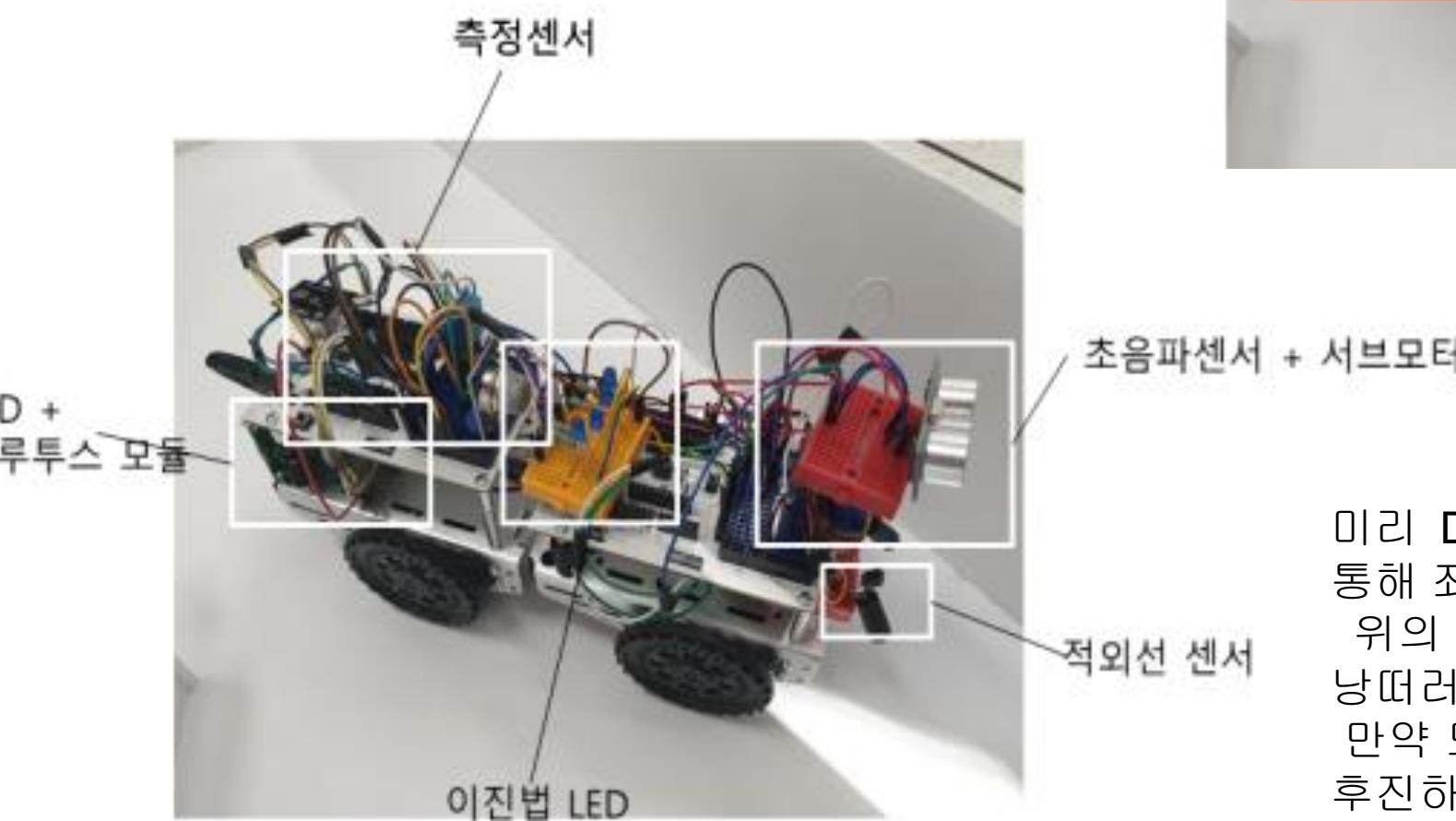
2019 임 베 디 드 경 진 대 회 에 서 입 상 한
프로젝트입니다.
팀원들과 화성에 있을 법 한 장애물과 판을 설계하여
직접 제작하며 기억에 남는 프로젝트입니다.

화성탐사로봇 SNAKE BOT

측정센서들을 통해 해당 행성의 정보를 수집하여 실시간으로 블루투스 모듈을 통해 스마트폰으로 정보를 송출합니다. 또한 LED이진법 장치를 통해 지구의 수학적 문명을 외계에 전파하는 역할을 수행합니다.

기여도

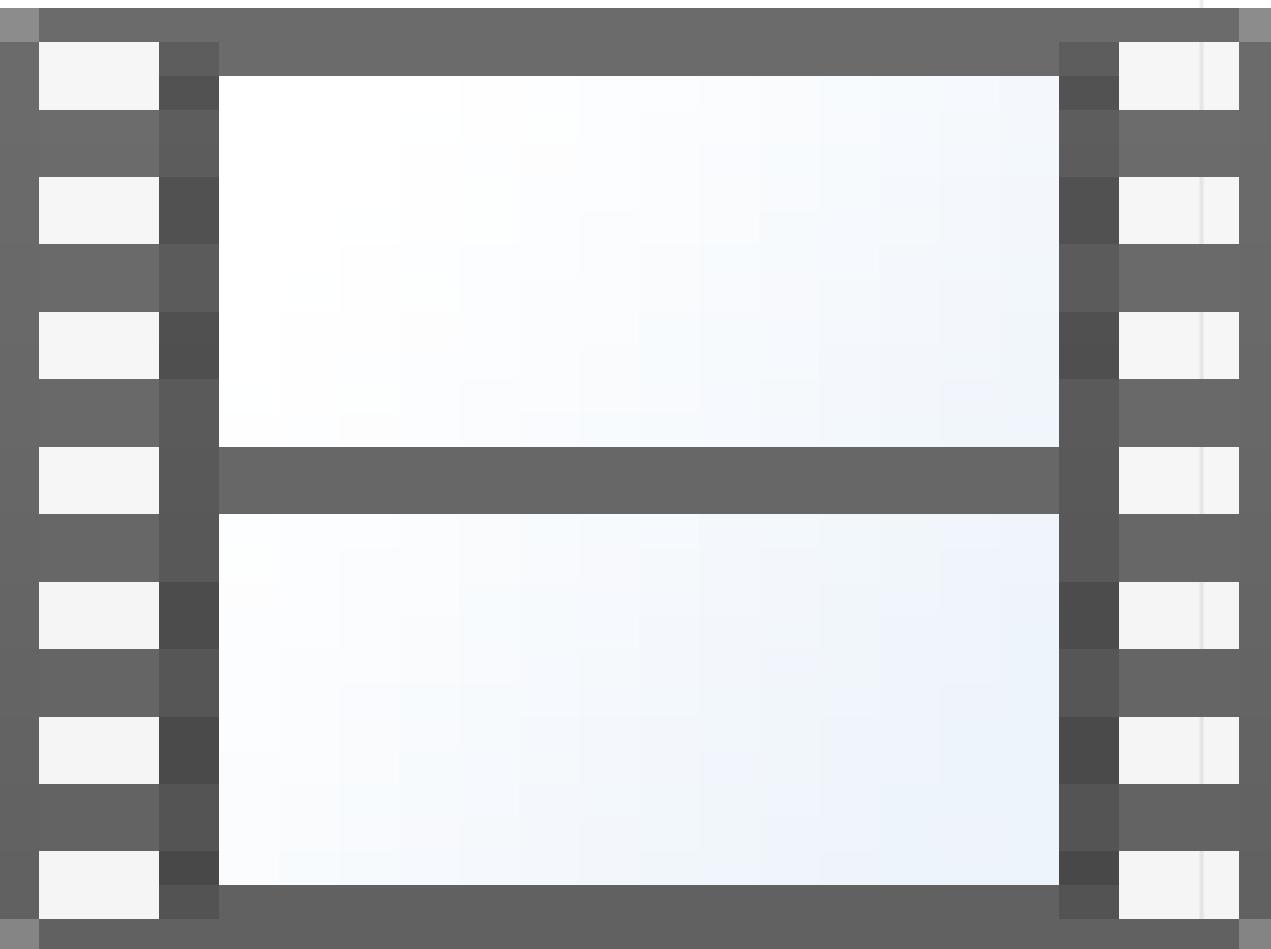
기획	95
개발	95



미리 Define된 안전거리보다 정면의 초음파로 측정한 거리가 가깝다면 초음파 센서를 통해 좌우의 거리를 판정하고 더 장애물로부터 거리적 여유가 있는 곳으로 회전합니다. 위의 동작은 적외선 센서로 탐지한 하단부가 낭떠러지가 아닐 경우에만 실행되며, 낭떠러지인 경우 안전한 위치로 회전 혹은 후진하고 회전 동작을 시행합니다. 만약 모서리에 걸려 회전이 한없이 반복되는 경우에는 이를 인지하여 조건 충족 시, 크게 후진하고 회전 동작을 하여 벗어납니다.

화성 탐사 로봇

SNAKE BOT 시현 영상



PROJECT.3

구조체 정보 출력 서버

수기 공모전 입상

ABOUT PROJECT

통신에 사용되는 구조체에 값을 넣어 TPS를 측정하는 시뮬레이터 중 일부분입니다.

현직 개발자의 코드를 분석하고 그에 맞춰 일부분을 구현했습니다.

“ Data storage method & Node data ”

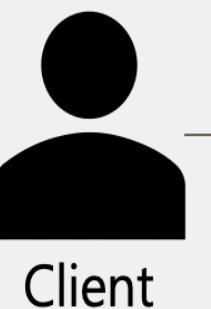
구조체를 사용한 재귀적 데이터 저장 방식

구조체 정보 출력 서버 TPS 시뮬레이터

프로그램 삭제 기능

삭제 요청 전 클라이언트 맵

```
*PRINT INFO* g_ciMapMax[0~2]
=====
g_ciMap[0]
  > list[0] choice = 8 index = 0
    > list[1] choice = 9 index = -1
      > list[2] choice = 9 index = -1
        > TPS : 10
=====
g_ciMap[1]
  > list[0] choice = 25 index = -1
    > TPS : 20
=====
g_ciMap[2]
  > list[0] choice = 26 index = 0
    > list[1] choice = 1 index = -1
      > TPS : 30
=====
```



Send : -3, 2

삭제 요청 후 클라이언트 맵

```
DELETE 2
*PRINT INFO* g_ciMapMax[0~1]
=====
g_ciMap[0]
  > list[0] choice = 8 index = 0
    > list[1] choice = 9 index = -1
      > list[2] choice = 9 index = -1
        > TPS : 10
=====
g_ciMap[1]
  > list[0] choice = 26 index = 0
    > list[1] choice = 1 index = -1
      > TPS : 30
=====
```

클라이언트의 삭제 요청이 오면 해당 인덱스부터 앞으로 밀어 구조체에 저장한 뒤 메모리를 재할당

구조체가 저장된 파일을 모두 파싱 후 MAP에 저장합니다.

이후 클라이언트가 필요한 구조체를 선택하면 클라이언트 MAP에 입력 후 TPS 통계를 알려줍니다.

기여도

기획
개발

100
100

Node data allocation on SHM

데이터의 깊이가 다른 재귀적 구조체 데이터를 공유메모리에
할당

오프셋 출력 프로그램 개발

구조체 정보가 담긴 헤더 파일

```
struct
{
    GULLInt_T data;
    GULLInt_T aspup;
    GULLInt_T aspup_ack;
    GULLInt_T aspdn;
    GULLInt_T aspdn_ack;
    GULLInt_T beat;
    GULLInt_T beat_ack;

    GULLInt_T aspac;
    GULLInt_T aspac_ack;
    GULLInt_T aspia;
    GULLInt_T aspia_ack;

    GULLInt_T notify;
    GULLInt_T error;

    GULLInt_T duna;
    GULLInt_T dava;
    GULLInt_T scon;
    GULLInt_T drst;
    GULLInt_T dupu;
    GULLInt_T daud;

    GULLInt_T reg;
    GULLInt_T reg_rsp;
    GULLInt_T dereg;
    GULLInt_T dereg_rsp;
} tx;
} G_MsM3ua_T;
```

오프셋 매크로가 지원하는 형태로 파일 출력

```
printf( fp, "G_MsM3ua_T tx.notify %d\n", offsetof(G_MsM3ua_T, tx.notify) );
printf( fp, "G_MsM3ua_T tx.error %d\n", offsetof(G_MsM3ua_T, tx.error) );
printf( fp, "G_MsM3ua_T tx.duna %d\n", offsetof(G_MsM3ua_T, tx.duna) );
printf( fp, "G_MsM3ua_T tx.dava %d\n", offsetof(G_MsM3ua_T, tx.dava) );
printf( fp, "G_MsM3ua_T tx.scon %d\n", offsetof(G_MsM3ua_T, tx.scon) );
```

구조체

```
printf( fp, "__fsid_t __val %d\n", offsetof(__fsid_t, __val) );
printf( fp, "__sigset_t __val %d\n", offsetof(__sigset_t, __val) );
printf( fp, "fd_set __fds_bits %d\n", offsetof(fd_set, __fds_bits) );
printf( fp, "__mbstate_t __count %d\n", offsetof(__mbstate_t, __count) );
printf( fp, "__G_fpos_t __pos %d\n", offsetof(__G_fpos_t, __pos) );
printf( fp, "__G_fpos_t __state %d\n", offsetof(__G_fpos_t, __state) );
printf( fp, "__G_fpos64_t __pos %d\n", offsetof(__G_fpos64_t, __pos) );
printf( fp, "__G_fpos64_t __state %d\n", offsetof(__G_fpos64_t, __state) );
```

다중 구조체일 경우 구조체 탑의 부모 구조체 탑을
찾아 출력합니다.

offset 값 출력 텍스트 파일

```
72 m3ua_sgw_stat_t tx_aspac_ack 88
73 m3ua_sgw_stat_t tx_aspia 96
74 m3ua_sgw_stat_t tx_aspia_ack 104
75 m3ua_sgw_stat_t tx_notify 112
76 m3ua_sgw_stat_t tx_error 120
77 m3ua_sgw_stat_t tx_duna 128
78 m3ua_sgw_stat_t tx_dava 136
79 m3ua_sgw_stat_t tx_scon 144
80 m3ua_sgw_stat_t tx_drst 152
81 m3ua_sgw_stat_t tx_dupu 160
82 m3ua_sgw_stat_t tx_daud 168
83 m3ua_sgw_stat_t tx_reg 176
```

구조체 정보 출력 서버 TPS 시뮬레이터

구조체가 저장된 파일을 모두 파싱 후 MAP에
저장합니다.
이후 클라이언트가 필요한 구조체를 선택
하면 클라이언트 MAP에 입력 후
TPS 통계를 알려줍니다.

기여도

기획	<div style="width: 100%;"></div>	100
개발	<div style="width: 100%;"></div>	100

PROJECT.4

AWS CRUD API

안 되는걸 되게 만든 프로젝트

ABOUT PROJECT

Lambda, AWS GATEWAY, DynamoDB로 구현한
API입니다.

전혀 지식이 없던 부분에서 구현했기에 가장
기억에 남는 프로젝트입니다.

AWS CRUD API 기능 소개

기여도

기획 100
개발 100

INSERT

[Method Execution /customer - POST - Method Test](#)

Make a test call to your method with the provided input

Path

No path parameters exist for this resource. You can define path parameters by using the syntax `{myPathParam}` in a resource path.

Request: /customer
Status: 200
Latency: 183 ms
Response Body

```
{
  "code": 200,
  "message": "customer add sucessfully"
}
```

Query Strings

`{customer}`
param1=value1¶m2=value2

Headers

`{customer}`
Use a colon (:) to separate header name and value, and new lines to declare multiple headers. e.g.
Accept:application/json.

Response Headers

```
{"X-Amzn-Trace-Id": "Root=1-5ef9d80f-7165ccde51900216fbeff0f;Sampled=0", "Content-Type": "application/json"}
```

Logs

```
Execution log for request 2c8c2218-0ed1-47-8081-2ab0bd8b31a4
Mon Jun 29 12:01:19 UTC 2020 : Starting execution for request: 2c8c2218-0ed1-4747-8081-2ab0bd8b31a4
Mon Jun 29 12:01:19 UTC 2020 : HTTP Method: POST, Resource Path: /customer
Mon Jun 29 12:01:19 UTC 2020 : Method request path: {}
Mon Jun 29 12:01:19 UTC 2020 : Method request query string: {}
Mon Jun 29 12:01:19 UTC 2020 : Method request headers: {}
Mon Jun 29 12:01:19 UTC 2020 : Method requ
```

[Method Execution /customer/scan - POST - Method Test](#)

Make a test call to your method with the provided input

Path

No path parameters exist for this resource. You can define path parameters by using the syntax `{myPathParam}` in a resource path.

Request: /customer/scan
Status: 200
Latency: 193 ms
Response Body

```
{
  "statusCode": 200,
  "body": [
    {
      "CUSTOMERS_ID": "a2345",
      "CUSTOMERS_NAME": "Lee Jung A",
      "LOCATION": "인천 서구"
    },
    {
      "CUSTOMERS_ID": "c31132",
      "CUSTOMERS_NAME": "Park Jun Gu",
      "LOCATION": "서울 강남구"
    },
    {
      "CUSTOMERS_ID": "20170735",
      "CUSTOMERS_NAME": "Kim soo han",
      "LOCATION": "서울 동작구"
    }
  ]
}
```

Query Strings

`{scan}`
param1=value1¶m2=value2

Headers

`{scan}`
Use a colon (:) to separate header name and value, and new lines to declare multiple headers. e.g.
Accept:application/json.

Stage Variables

No stage variables exist for this method.

Request Body

SCAN

DELETE

[Method Execution /delete - DELETE - Method Test](#)

Make a test call to your method with the provided input

Path

No path parameters exist for this resource. You can define path parameters by using the syntax `{myPathParam}` in a resource path.

Query Strings

`{delete}`
customerId=b3942

Headers

`{delete}`
Use a colon (:) to separate header name and value, and new lines to declare multiple headers. e.g.
Accept:application/json.

Logs

Request: /delete?customerId=b3942
Status: 200
Latency: 725 ms
Response Body

```
null
```

[조회]

[/customer](#)

- GET**
- POST**
- DELETE**
- UPDATE**
- POST**

Path

No path parameters exist for this resource. `{myPathParam}` in a resource path.

Query Strings

`{customer}`
customerId=a2345

[결과]

Request: /customer?customerId=a2345
Status: 200
Latency: 754 ms
Response Body

```
{
  "CUSTOMERS_ID": "a2345",
  "CUSTOMERS_NAME": "Lee Jung A",
  "LOCATION": "Kr"
}
```

QUERY

AWS CRUD API 구현 소개

Lambda

The Lambda Functions console displays a list of five functions:

Function name	Description	Runtime	Code size	Last modified
updateItem		Python 3.8	343 bytes	8 minutes ago
DeleteItem		Python 3.6	339 bytes	4 hours ago
AllItem		Python 3.8	302 bytes	2 hours ago
AddCustomer		Python 3.6	302 bytes	6 hours ago
GetCustomerById		Python 3.6	306 bytes	4 hours ago

함수	기능
AddCustomer	Insert 기능
DeleteItem	Delete 기능
AllItem	Scan기능
GetCustomerById	Query
UpdateItem	Update 기능

Lambda와 DynamoDB를 연결 허가해주는 IAM입니다.

The IAM Roles console shows the details for the 'dynamodb_all_role' role:

- Role ARN: arn:aws:iam::[REDACTED]:role/dynamodb_all_role
- Role description: Allows Lambda functions to call AWS services on your behalf. | Edit
- Instance Profile ARNs: [REDACTED]
- Path: /
- Creation time: 2020-06-29 17:24 UTC+0900
- Last activity: 2020-06-29 23:31 UTC+0900 (Today)
- Maximum CLI/API session duration: 1 hour | Edit

The Permissions tab shows the attached policy 'dynamodb_all' is a Managed policy.

IAM

기여도
기획
개발

100
100

Dynamo DB

GATEWAY GET/POST

The API Gateway console shows the methods for the '/customer' resource:

- GET**: arn:aws:lambda:ap-northeast-2:810014102669:function:customers-api-13b... (Authorization: None, API Key: Not required)
- POST**: arn:aws:lambda:ap-northeast-2:810014102669:function:customers-api-13b... (Authorization: None, API Key: Not required)

Gateway는 GET, POST, DELETE 메소드로 구현했습니다.

PROJECT.5

언어학습 VR 게임

졸업작품 경진대회

코엑스 출품작

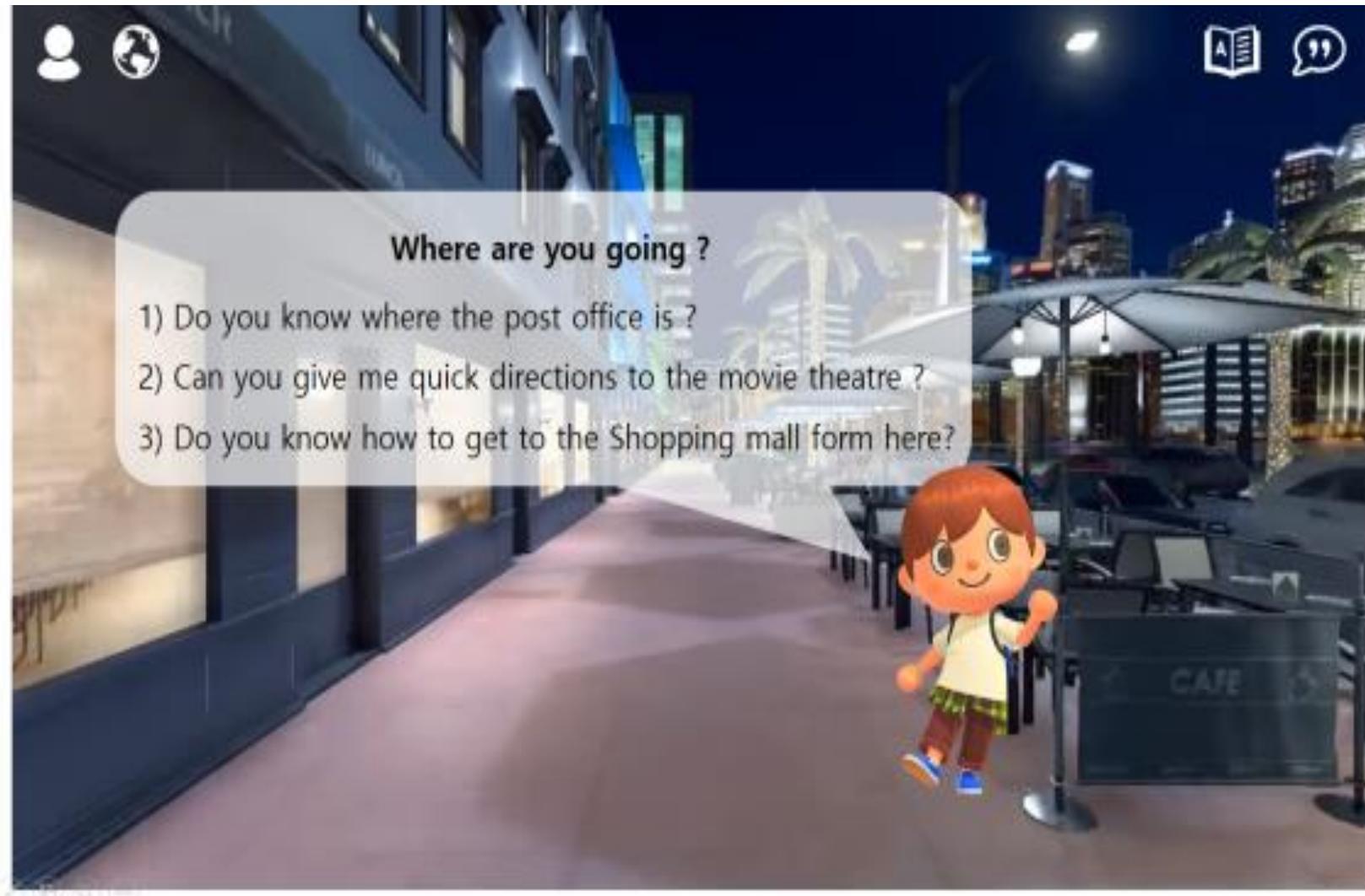
ABOUT PROJECT

Speech To Text, Text To Speech, 단어 교정 등의
API를 연동하여 게임의 진행 요소로 사용했습니다.

언어학습 VR 게임

졸업작품 경진대회 코엑스 출품작

기능 소개



기여도

기획

개발

100

60

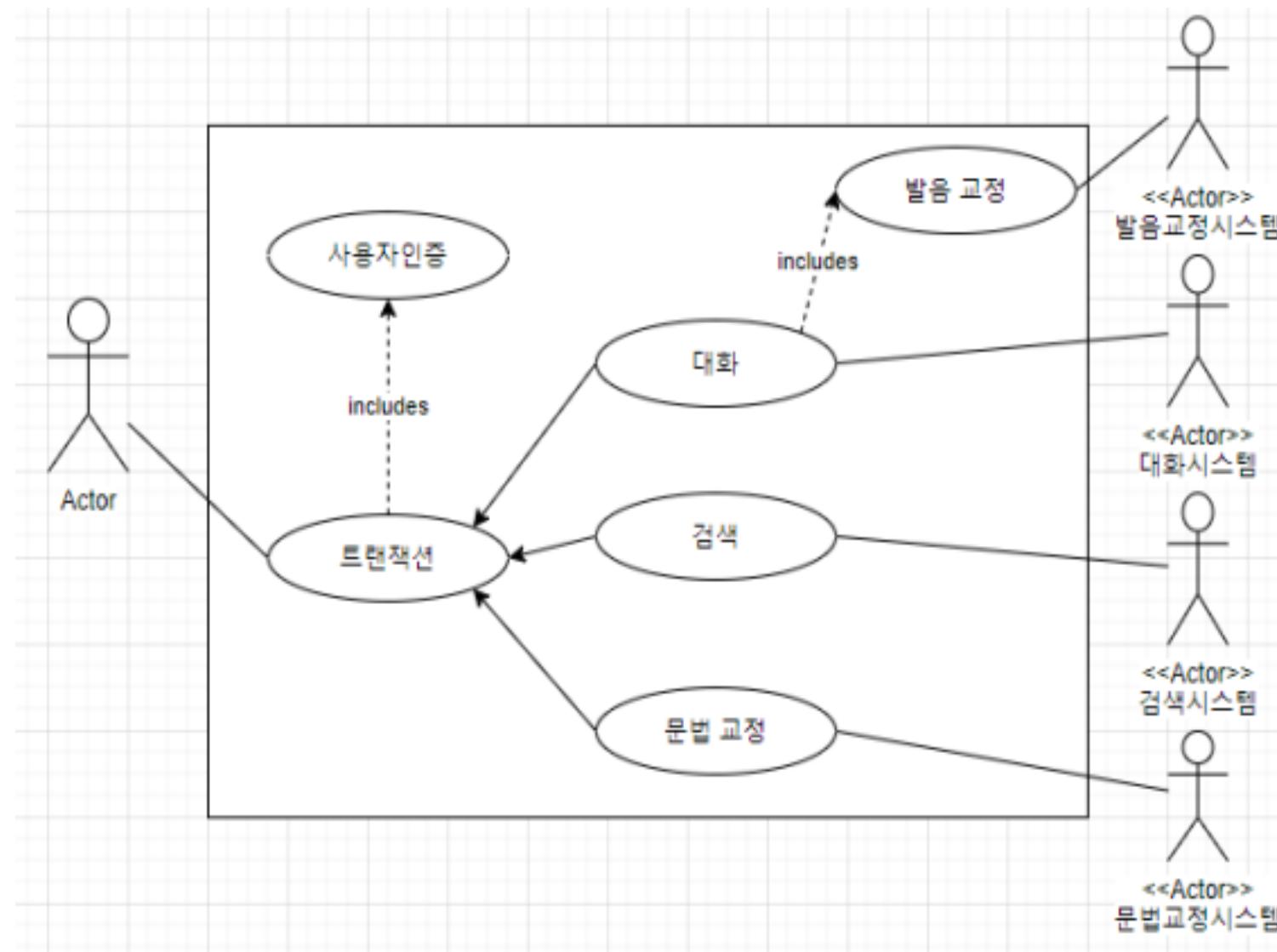
ETC

코로나로 인해 비대면 수업으로 변경되며 어렵거나 힘든 부분을 교수님께 여쭤보지 못해 팀원들과 힘을 합쳐 만든 작품입니다.

언어학습 VR 게임

졸업작품 경진대회 코엑스 출품작

유즈케이스 다이어그램



음성인식을 위한 API **Google Cloud Speech-to-Text**

사용하기 간편한 API로 강력한 신경망 모델을 적용하는 **Google Speech-to-Text**를 사용하면 개발자가 오디오를 텍스트로 변환할 수 있습니다. 이 API는 120개 언어와 방언을 인식하여 전 세계 사용자층을 지원합니다. 음성 명령 및 제어 기능을 구현하고 콜센터의 오디오를 텍스트로 변환하는 등의 작업을 할 수 있고 **Google**의 머신러닝 기술을 사용하여 실시간 스트리밍 또는 사전 녹음 오디오를 처리할 수 있습니다.

편입 후 계획

ABOUT PLAN

학습목표, 자격증, 어학 등 세부적인 계획을
수립했습니다.

편입 후 계획

전공

상위 10프로 이상 성적 유지

전공 필수과목 이수 및
네트워크 관련 수업 이수

어학

토익 800점 이상의 점수

자격증

AWS 자격증

정보처리기사

정보보안산업기사

대외활동

재능 기부 봉사활동
코딩 경진대회

감사합니다!

1999 – 2020 PORTFOLIO

CONTACT

dlwjddk218@naver.com

010 2603 8084

