

PROFILE

이름 : 정지원

생년월일 : 1998.03.18

E-mail : ckjjw2953@naver.com

Phone : 010-3680-2259

학력-경력 ▼

- 상명대학교, 스마트정보통신공학과 (2017. 03 ~ 졸업예정)
- 육군 만기전역, 통신병 (2018. 01 ~ 2019. 09)

수상내역 ▼

- 제12회 대한전기학회 대학생 작품경진대회, 금상 (2022. 11)

대외활동 ▼

- 타이드 스퀘어 (IT부서 아르바이트, 2019. 09 ~ 2019. 10)
- SK 증권 (IT부서 아르바이트, 2019. 11 ~ 2019. 12)
- 코딩온 (코딩 학원 강사, 2022.08 ~)

INDEX

자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선

01

온라인 게임 '리그 오브 레전드' 캐릭터들의 승률 랭킹 시스템

02

NFC태그 기술을 활용한 코로나 출입명부

03

랜덤 매칭 기능을 포함한 SNS 어플리케이션

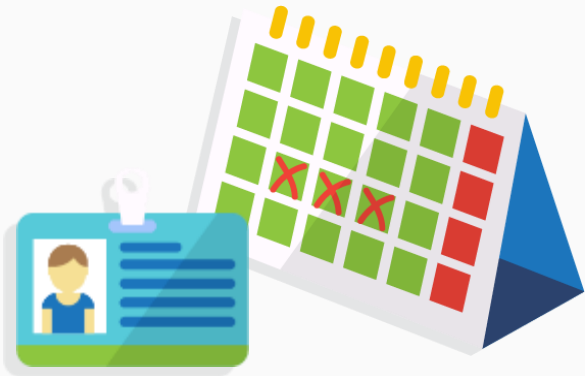
04

ATmega128을 활용한 마스크, 손세정제 자판기

05

PHP 서버를 활용한 뉴스 스크랩 웹 페이지

06



01. 자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선



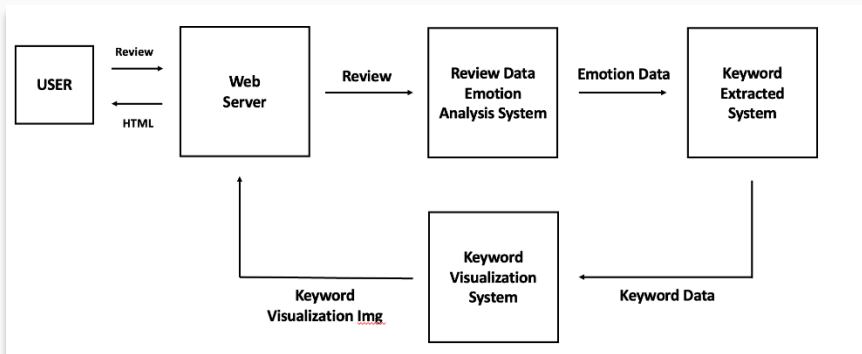
프로젝트 인원 : 4명

개발환경 및 언어 : Google Colab ,Python

프로젝트 內 역할 : 모델 학습을 위한 리뷰 데이터 수집 및 전처리, KoBERT 모델에 리뷰 데이터 학습, 분석 결과 시각화

프로젝트 요약

- 현재 쇼핑몰 리뷰 시스템을 개선해보고자 프로젝트 주제를 선정(한정적인 리뷰 키워드, 직관적이지 못함, 사용자의 키워드 직접 입력 시 오입력 가능성, 평점은 높지만 부정 리뷰)
- 사용자가 입력한 텍스트 리뷰를 학습된 모델을 통해 긍정/부정 리뷰로 분류
- 긍정/부정 리뷰에서 각각 명사 키워드 추출
- 긍정/부정 리뷰에서 추출된 긍정리뷰의 명사, 부정리뷰의 명사 각각에서 빈도수가 높은 3개의 명사에 해당하는 형용사들만 추출
- 추출된 긍정/부정 명사는 WordCloud로 시각화 하여 사용자에게 노출
- 상위 3개의 명사에 연관된 형용사는 Seaborn으로 시각화 하여 사용자에게 노출



01. 자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선

```
from sklearn.model_selection import train_test_split
dataset_train, dataset_test = train_test_split(data_list, test_size=0.25, shuffle=True, random_state=42)
```

```
dataset_train[1]
```

```
['사이즈는 정사이즈여서 본인입는 반팔티 참고해서 주문시키면될거같아요!!', '1']
```



```
(array([ 2, 2618, 7310, 5760, 4092, 6493, 7096, 7310, 6916, 6553, 2415,
        7138, 5760, 2207, 7690, 7673, 4430, 7850, 4227, 6722, 6198, 5902,
        5377, 5354, 6797, 6999, 517, 5, 517, 5, 3, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1], dtype=int32),
array(31, dtype=int32),
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        dtype=int32), 1)
```

훈련용 데이터셋과 검증용 데이터셋을 0.75 : 0.25 로 나눠준다

BERT의 Tokenizer를 통한 토큰화 및 패딩 작업

첫 번째 array: Padding Sequence
 두 번째 array: 길이와 타입에 대한 내용
 세 번째 array: Attention Mask Sequence

01. 자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선

```
for e in range(num_epochs):
    train_acc = 0.0
    test_acc = 0.0
    model.train()
    for batch_id, (token_ids, valid_length, segment_ids, label) in enumerate(tqdm(train_dataloader)):
        optimizer.zero_grad()
        token_ids = token_ids.long().to(device)
        segment_ids = segment_ids.long().to(device)
        valid_length = valid_length
        label = label.long().to(device)
        out = model(token_ids, valid_length, segment_ids)
        loss = loss_fn(out, label)
        loss.backward()
        torch.nn.utils.clip_grad_norm_(model.parameters(), max_grad_norm)
        optimizer.step()
        scheduler.step() # Update learning rate schedule
        train_acc += calc_accuracy(out, label)
        if batch_id % log_interval == 0:
            print("epoch {} batch id {} loss {} train acc {}".format(e+1, batch_id+1, loss.data.cpu().numpy(), train_acc / (batch_id+1)))
    print("epoch {} train acc {}".format(e+1, train_acc / (batch_id+1)))

    model.eval()
    for batch_id, (token_ids, valid_length, segment_ids, label) in enumerate(tqdm(test_dataloader)):
        token_ids = token_ids.long().to(device)
        segment_ids = segment_ids.long().to(device)
        valid_length = valid_length
        label = label.long().to(device)
        out = model(token_ids, valid_length, segment_ids)
        loss = loss_fn(out, label)
        loss.backward()
        test_acc += calc_accuracy(out, label)
    print("epoch {} test acc {} val_loss {}".format(e+1, test_acc / (batch_id+1), loss.data.cpu().numpy()))
```

```
epoch 10 batch id 1001 loss 0.0012824744917452335 train acc 0.9948020729270729
epoch 10 train acc 0.9947659337944664
```

100%  338/338 [03:21<00:00, 2.23it/s]

```
epoch 10 test acc 0.9283468934911243 val_loss 0.005354325752705336
```

num_classes = 2

dr_rate = 0.1

optimizer = AdamW

loss function = CrossEntropyLoss

	train	test
acc	0.99	0.93
loss	0.001	0.005

<학습 결과>

01. 자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선

```
하고싶은 말을 입력해주세요 : 가성비 좋네요  
/usr/local/lib/python3.7/dist-packages/torch/utils/data/dataloader.py:490: UserWarning:  
  cpuset_checked))  
>> 해당 리뷰는 긍정적 입니다.  
  
하고싶은 말을 입력해주세요 : 생각했던 것 보다 괜찮네요  
>> 해당 리뷰는 긍정적 입니다.  
  
하고싶은 말을 입력해주세요 : 마감처리도 별로고 색감도 사진이랑 너무 다르네요  
>> 해당 리뷰는 부정적 입니다.  
  
하고싶은 말을 입력해주세요 : 돈 주고 이 옷 살바에는 다른데로 갈아타렵니다  
>> 해당 리뷰는 부정적 입니다.  
  
하고싶은 말을 입력해주세요 : 가격이 너무 저렴해서 별로일 줄 알았는데 생각보다 괜찮네요  
>> 해당 리뷰는 긍정적 입니다.
```

<모델 검증>



```
0.22300000488758087  
99.7770004272461  
0.10300000011920929  
99.89700317382812  
0.07800000160932541  
99.9219970703125  
0.07199999690055847  
99.9280014038086  
99.84500122070312  
0.1550000011920929  
0.08500000089406967  
99.91500091552734  
0.07400000095367432  
99.9260025024414  
0.07199999690055847  
99.9280014038086  
0.07199999690055847  
99.9280014038086
```

<긍정 / 부정 판단 후 각각 리스트에 따로 저장>

01. 자연어 처리기술을 이용한 쇼핑몰 리뷰 시스템 개선

```
pos_nouns = []
for review in pos_reviews:
    for noun in tagger.nouns(review):
        if noun not in stop_words:
            pos_nouns.append(noun)

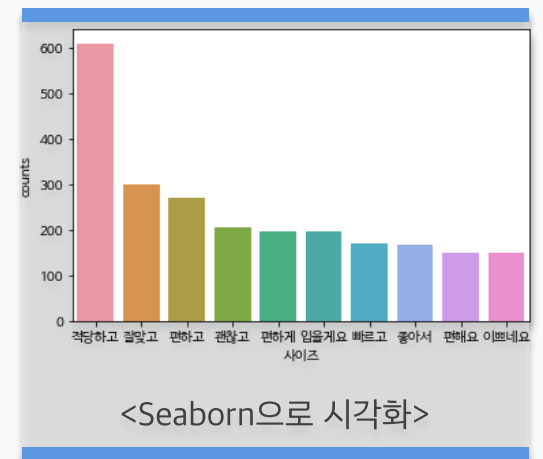
neg_nouns = []
for review in neg_reviews:
    for noun in tagger.nouns(review):
        if noun not in stop_words:
            neg_nouns.append(noun)
```

<긍정 / 부정 키워드 따로 추출>



```
p1_pos_adj_list = []
for my_sentence in sentences_tag:
    for word, tag in my_sentence:
        if tag in ['Adjective']:
            if word not in stop_words:
                p1_pos_adj_list.append(word)
```

<명사에 따른 형용사 추출>



02. 온라인 게임 '리그 오브 레전드' 캐릭터들의 승률 랭킹 시스템

<테이블 항목>

1. 챔피언승률
2. 챔피언의 소환사주문 별 승률
3. 챔피언의 룬 별 승률
4. 챔피언의 시작 아이템 별 승률
5. 챔피언의 신발 별 승률

<Metrix 항목>

1. 나만의 스타일 승률(자신이 고른 시작아이템, 룬, 소환사주문에 따른 승률들)
2. 챔피언들의 스펠, 신발, 시작아이템 조합
3. 챔피언들의 룬과 스펠의 승률이 50% 이상인 스펠, 룬 조합
4. '챔피언승률' 테이블의 'champ_winning_rate' 컬럼을 챔피언과 포지션에 상관없이 내림차순으로 정렬
5. 챔프들의 승률 상관없이 룬, 스펠, 신발의 조합

프로젝트 인원 : 1명

개발환경 및 언어 : MySQL Workbench, Python(크롤링)

프로젝트 內 역할 : E-R모델 작성, 관계 데이터 모델 작성, 함수 종속 다이어그램 작성,
Metrix 코드 작성

프로젝트 요약

- 크롤링을 통해 온라인 게임 '리그 오브 레전드'의 승률 사이트에서 수집한 데이터를 MySQL Workbench에 적재
- E-R모델, 관계 데이터 모델, 함수 종속 다이어그램을 작성하고 구현하고자 하는 Metrix 목록을 선정한다.
- 나만의 스타일(자신 고른 시작아이템, 룬, 소환사 주문에 따른)의 승률, 캐릭터들의 소환사 주문, 신발, 시작아이템의 조합에 따른 승률, 승률이 50% 이상인 소환사 주문, 룬의 조합, 캐릭터의 독립적인 승률에 상관없는 룬, 소환사 주문, 신발 조합의 승률 등의 Metrix 코드를 작성한다.

02. 온라인 게임 '리그 오브 레전드' 캐릭터들의 승률 랭킹 시스템

Query 1

```

1 * select rune_winning_rate * spell_winning_rate * startitem_winning_rate / 10000
2   # 나만의 스타일
3   from rune_winrate, spell_winrate, startitem_winrate

```

Result Grid

나만의 스타일
12.5410079344
12.2512815808
11.975598048000002
12.579962624800002
13.721563811999999
12.672480252
12.7747363768
11.0266403576
11.698610492
12.467967602400002
12.8867315992
12.0418995824
12.7552591816
12.3851886728
12.7382164608
13.013334668
12.772301902400002
11.735130608
11.791389421600002
11.761912526400002

Result 1

나만의 스타일 승률(자신이 고른 시작아이템, 룬, 소환사 주문에 따른 승률)

캐릭터들의 소환사 주문, 신발, 시작아이템 조합에 따른 승률

Query 1

```

1 * select spell_winrate.champ_name, spell, spell_winning_rate, shoes, shoes_winning_rate, start_item, startitem_winning_rate
2   from spell_winrate
3   left join shoes_winrate
4   on spell_winrate.champ_name = shoes_winrate.champ_name
5   join startitem_winrate
6   on spell_winrate.champ_name = startitem_winrate.champ_name

```

Result Grid

champ_name	spell	spell_winning_rate	shoes	shoes_winning_rate	start_item	startitem_winning_rate
가현	절차포트/점멸	40.86	한글 장화	44.88	도란의 곶	47.72
가현	절차포트/점멸	40.86	책트레스의 발걸음	52	도란의 곶	47.72
가현	절차포트/점멸	40.86	광천사의 군화	53.11	도란의 곶	47.72
가현	절차포트/점멸	40.86	한글 장화	44.88	도란의 방패	49.49
가현	절차포트/점멸	40.86	책트레스의 발걸음	52	도란의 방패	49.49
가현	절차포트/점멸	40.86	광천사의 군화	53.11	도란의 방패	49.49
가현	점착/점멸	52.15	한글 장화	44.88	도란의 곶	47.72
가현	점착/점멸	52.15	책트레스의 발걸음	52	도란의 곶	47.72
가현	점착/점멸	52.15	광천사의 군화	53.11	도란의 곶	47.72
가현	점착/점멸	52.15	한글 장화	44.88	도란의 방패	49.49
가현	점착/점멸	52.15	책트레스의 발걸음	52	도란의 방패	49.49
가현	점착/점멸	52.15	광천사의 군화	53.11	도란의 방패	49.49
갈리오	절차포트/점멸	51.72	책트레스의 발걸음	52.54	도란의 방치	53.14
갈리오	절차포트/점멸	51.72	한글 장화	54.15	도란의 방치	53.14
갈리오	절차포트/점멸	51.72	마법사의 신발	63.19	도란의 방치	53.14
갈리오	점착/점멸	52.67	책트레스의 발걸음	52.54	도란의 방치	53.14
갈리오	점착/점멸	52.67	한글 장화	54.15	도란의 방치	53.14
갈리오	점착/점멸	52.67	마법사의 신발	63.19	도란의 방치	53.14

Result 2

Query 1

```

1 * select champ_winrate.champ_name, rune_winrate.rune, spell_winrate.spell
2   from champ_winrate, rune_winrate, spell_winrate
3  where champ_winrate.champ_name = rune_winrate.champ_name && champ_winrate.champ_name = spell_winrate.champ_name
4        and rune_winrate.rune_winning_rate > 50 && spell_winrate.spell_winning_rate > 50;

```

Result Grid

	champ_name	rune	spell
1	갈리오	모닉자	텔레포트/장풍
2	갈리오	모닉자	장화/장풍
3	갈리오	여진	텔레포트/장풍
4	갈리오	여진	장화/장풍
5	그라프스	북	강타/장풍
6	그릴	장벽자	장화/텔레포트
7	나미	풍생의 소환	장진/장풍
8	나미	수호자	장진/장풍
9	나세스	가진한 발물함	텔레포트/장풍
10	나세스	가진한 발물함	텔레포트/유체화
11	나세스	풍인물함 주...	텔레포트/장풍
12	나세스	풍인물함 주...	텔레포트/유체화
13	누누와 유희포	여름의 수확	강타/장풍
14	누누와 유희포	여름의 수확	텔레포트/강타
15	누누와 유희포	난입	강타/장풍
16	누누와 유희포	난입	텔레포트/강타
17	니코	장벽	장화/장풍
18	다리우스	장벽자	유체화/장풍
19	다이애나	장벽자	강타/장풍
20	라칸	수호자	장화/장풍

캐릭터들의 룬과 소환사 주문의 승률이 50% 이상인 조합

캐릭터들의 승률과 상관없는 룬, 소환사 주문, 신발의 조합

```

Query 1
1 * select rune_winrate.champ_name, rune, rune_winning_rate, spell, spell_winning_rate, shoes, shoes_winning_rate from rune_winrate
2 left join spell_winrate on rune_winrate.champ_name = spell_winrate.champ_name
3 join shoes_winrate on rune_winrate.champ_name = shoes_winrate.champ_name

```

champ_name	rune	rune_winning_rate	spell	spell_winning_rate	shoes	shoes_winning_rate
갈리오	포식자	50.49	텔레포트/점멸	51.72	책상에서의 발걸음	52.54
갈리오	포식자	50.49	텔레포트/점멸	51.72	한글 강화	54.15
갈리오	포식자	50.49	텔레포트/점멸	51.72	각본서의 신발	63.19
갈리오	포식자	50.49	점멸/점멸	52.67	책상에서의 발걸음	52.54
갈리오	포식자	50.49	점멸/점멸	52.67	한글 강화	54.15
갈리오	포식자	50.49	점멸/점멸	52.67	각본서의 신발	63.19
갈리오	여전	52.4	텔레포트/점멸	51.72	책상에서의 발걸음	52.54
갈리오	여전	52.4	텔레포트/점멸	51.72	한글 강화	54.15
갈리오	여전	52.4	점멸/점멸	52.67	각본서의 신발	63.19
갈리오	여전	52.4	점멸/점멸	52.67	책상에서의 발걸음	52.54
갈리오	여전	52.4	점멸/점멸	52.67	한글 강화	54.15
갈리오	여전	52.4	점멸/점멸	52.67	각본서의 신발	63.19
갈리오	착취...	47.52	텔레포트/점멸	47.71	책상에서의 발걸음	40
갈리오	착취...	47.52	텔레포트/점멸	47.71	한글 강화	47.73
갈리오	착취...	47.52	텔레포트/점멸	47.71	명석탑의 아이오...	53.18
그라가스	포	52.21	강타/점멸	51.9	책상에서의 발걸음	28.57
그라가스	포	52.21	강타/점멸	51.9	명석탑의 아이오...	42.74
그라가스	포	52.21	강타/점멸	51.9	각본서의 신발	63.35
그레이브즈	선입	47.52	강타/점멸	46.32	한글 강화	52.45
그레이브즈	선입	47.52	강타/점멸	46.32	책상에서의 발걸음	55.36

03. NFC태그 기술을 활용한 코로나 출입명부



“NFC를 이용한 인증 절차 간소화”

어플을 백그라운드 실행 상태에 두고
핸드폰을 NFC 스티커에 태그만 하면
사용자 정보가 암호화된 상태로 전송

프로젝트 인원 : 3명

개발환경 및 언어 : Android Studio, Firebase, JAVA

프로젝트 內 역할 : 앱 Layout 설계 및 제작, 사용자 데이터 암호·복호화

Q. QR코드 체크인을 하면서 불편해 하시는 분을 목격했거나 불편함을 겪은 적이 있는가?



78.3%

경험이 있다
(106명 중 83명이 답변)

프로젝트 요약

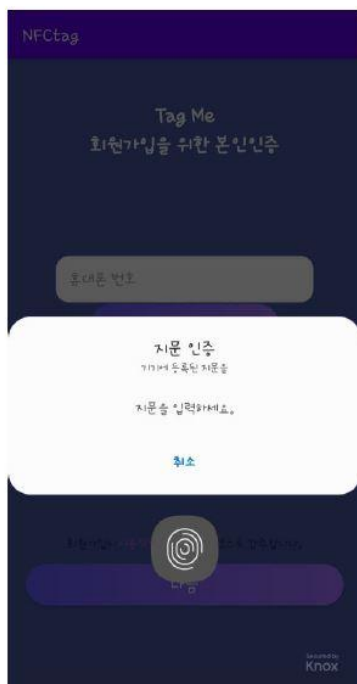
- QR코드를 사용한 코로나 출입명부 작성 시 여러 번거로움을 해결하기 위해 프로젝트 주제를 선정
- 사용자용, 관리자용 어플리케이션을 따로 제작
- 어플리케이션 실행 시 휴대폰 내의 NFC 태그 모드 활성화
- 스티커형 NFC 태그에 태그 시 사용자 정보가 암호화 되어 서버로 전송
- 서버의 데이터를 복호화 하여 관리자 어플리케이션에서 사용자 정보 확인

03. NFC태그 기술을 활용한 코로나 출입명부

| Layout



로그인 Layout



지문 인식 Layout



회원가입 Layout

회원가입 시 휴대폰 내 생체정보를 통해 지문 인식 후 회원가입 완료

03. NFC태그 기술을 활용한 코로나 출입명부

| Layout



User Layout

사용자 어플리케이션 실행 시 NFC 태그 모드
활성화

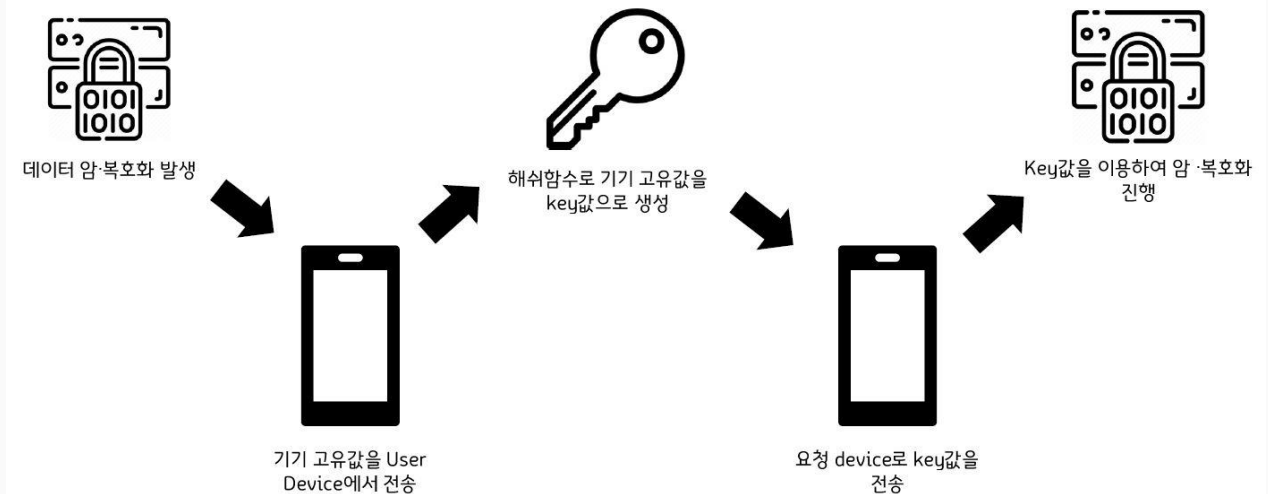
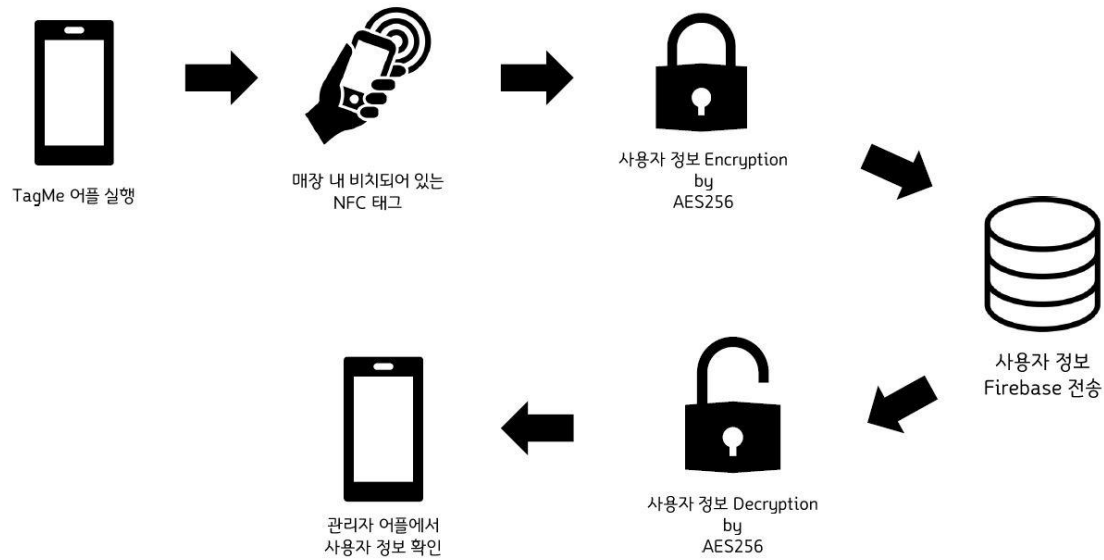


Administer Layout

관리자 계정으로 관리자 어플리케이션 실행 시
NFC 스티커에 정보 쓰기 및 방문자 정보 확인
가능

03. NFC태그 기술을 활용한 코로나 출입명부

| Function



04. 랜덤 매칭 기술을 포함한 SNS 어플리케이션



프로젝트 인원 : 4명

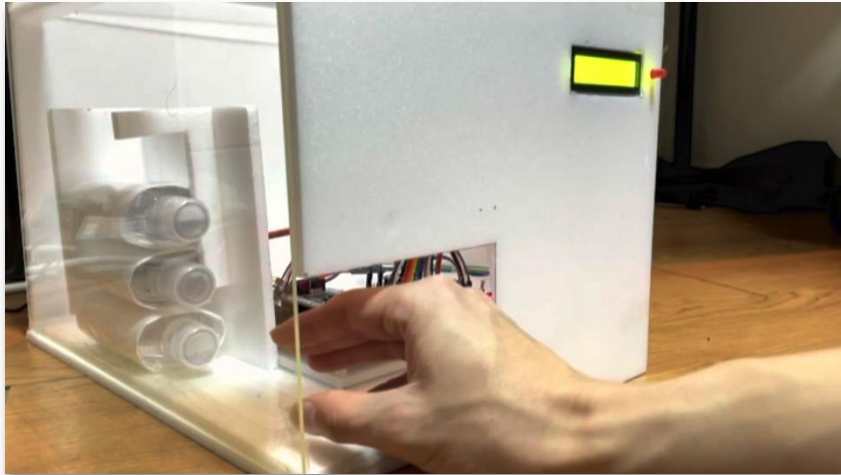
개발환경 및 언어 : Android Studio, Firebase, JAVA

프로젝트 內 역할 : 앱 Layout 설계 및 제작, 로그인·회원가입 기능 구현, 랜덤매칭 알고리즘 설계, 게시판 기능 구현

프로젝트 요약

>> [프로젝트 깃허브 이동](#)

05. ATmega128을 활용한 마스크, 손세정제 자판기



프로젝트 인원 : 4명

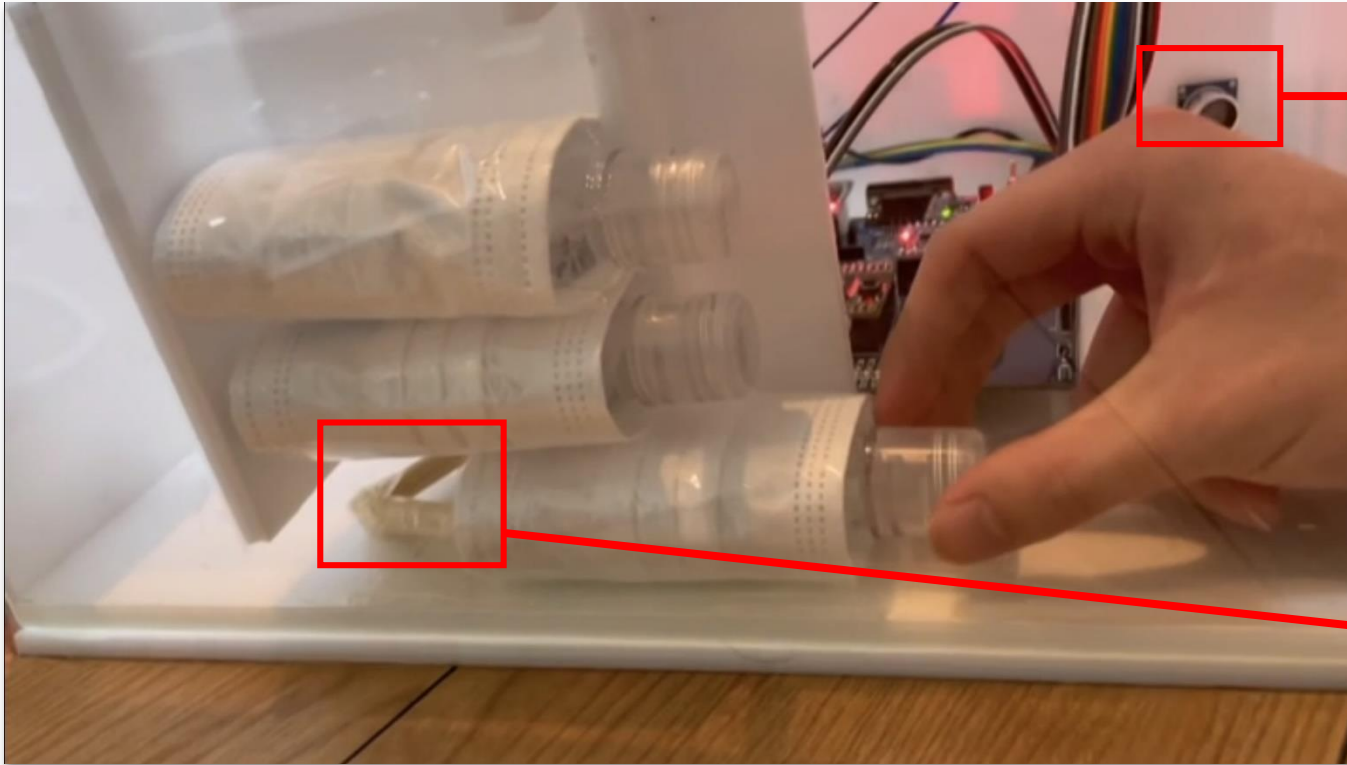
개발환경 및 언어 : ATmega128 MCU, CodeVision AVR, C언어

프로젝트 內 역할 : 초음파 센서 연동 모터 설계, 회로 및 외형 제작

프로젝트 요약

- ATmega128 MCU, CodeVision AVR를 통해 동작하는 마스크+손세정제 자판기를 제작한다.
- 손을 넣는 입구 내 우측벽에 설치된 초음파 센서를 통해 사용자의 손을 감지 시 동작한다.
- 정면 LCD 패널에는 처음 설정해놓은 마스크+손세정제의 잔여량이 표시된다.
- 자판기가 한번 동작할 때 마다 LCD에 표기되는 숫자가 감소한다.
- 마스크+손세정제의 잔여량이 0이되면 LCD에 모두 소진되었다고 표기되고 옆에 달린 LED가 점등되어 모두 소진되었다고 알려준다.

05. ATmega128을 활용한 마스크, 손세정제 자판기



손을 집어넣으면 벽면에 설치된 초음파 센서가
사용자의 손을 인식

초음파 센서가 작동되면 연동 되어있는 모터가
작동해 막대가 마스크+손세정제 키트를 밀어
내보내 준다

05. ATmega128을 활용한 마스크, 손세정제 자판기



초기 설정된 값인 3개의 마스크+손세정제 키트가
남아있다고 알려주는 LCD패널 화면



자판기 동작 시 마다 잔여량이 1씩 줄어들고
잔여량이 0이 되면 다시 채워달라는 문구가
출력된다

06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지



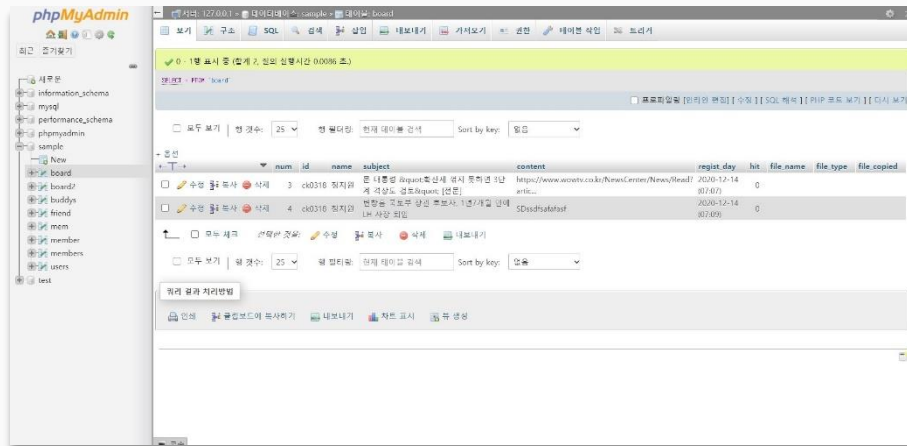
프로젝트 인원 : 1명

개발환경 및 언어 : phpMyAdmin, PHP, HTML, Javascript, CSS

프로젝트 內 역할 : PHP서버 설계, 웹 페이지 설계 및 구현

프로젝트 요약

- PHP와 HTML공부를 하며 제작한 뉴스 스크랩 웹 페이지이다.
- 회원가입을 하게 되면 서버로 회원가입 페이지에서 입력된 정보가 전송된다.
- 로그인을 하면 메인 페이지의 'Login', 'Sign Up'버튼이 'Logout'버튼으로 변하게 된다.
- 메뉴 바 왼쪽 상단에는 로그인 된 사용자의 이름과 아이디가 보여지고 레벨과 포인트를 확인할 수 있다.
- 뉴스 스크랩 작성시 포인트가 지급된다.
- 사용자 정보수정 페이지에서 입력된 정보를 수정하면 서버에 저장된 정보가 수정된다.
- 나중에 볼 뉴스를 저장하면 메인 페이지에 '아직 안 읽은 기사목록'으로 노출된다.



06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지



로그인 후 메뉴바의 목록 변화

06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지

My News Scraper

[Login](#)[Sign Up](#)[HOME](#)[For Later](#)[My News](#)

로그인

My News Scraper

[Login](#)[Sign Up](#)[HOME](#)[For Later](#)[My News](#)

회원 가입

아이디 비밀번호 비밀번호 확인 이름 이메일

@

정지원(ck0318)님[Level:9, Point:1200]

[Log Out](#)[정보 수정](#)[HOME](#)[For Later](#)[My News](#)

회원 정보수정

아이디 비밀번호 비밀번호 확인 이름 이메일

@

로그인, 회원가입, 회원정보 수정 페이지

06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지

[정보 수정](#)
[HOME](#)
[For Later](#)
[My News](#)

For Later > 나중에 볼 기사 저장하기

이름 : 정지원

제목 : 문 대통령 "확진세 가지 못하면 3단계 격상도 검토" [전문]

내용 및 기사링크 : <https://www.wootv.co.kr/NewsCenter/News/Read?articleId=420201213005611-388>

카테고리 : 코로나19

완료 목록

My News Scraper

정지원(ck0318)님[Level:9, Point:1200]

[Log Out](#)
[정보 수정](#)
[HOME](#)
[For Later](#)
[My News](#)

아직 안읽은 기사목록

문 대통령 "확진세 가지 못하면 3단계 격상도 검토" [전문]

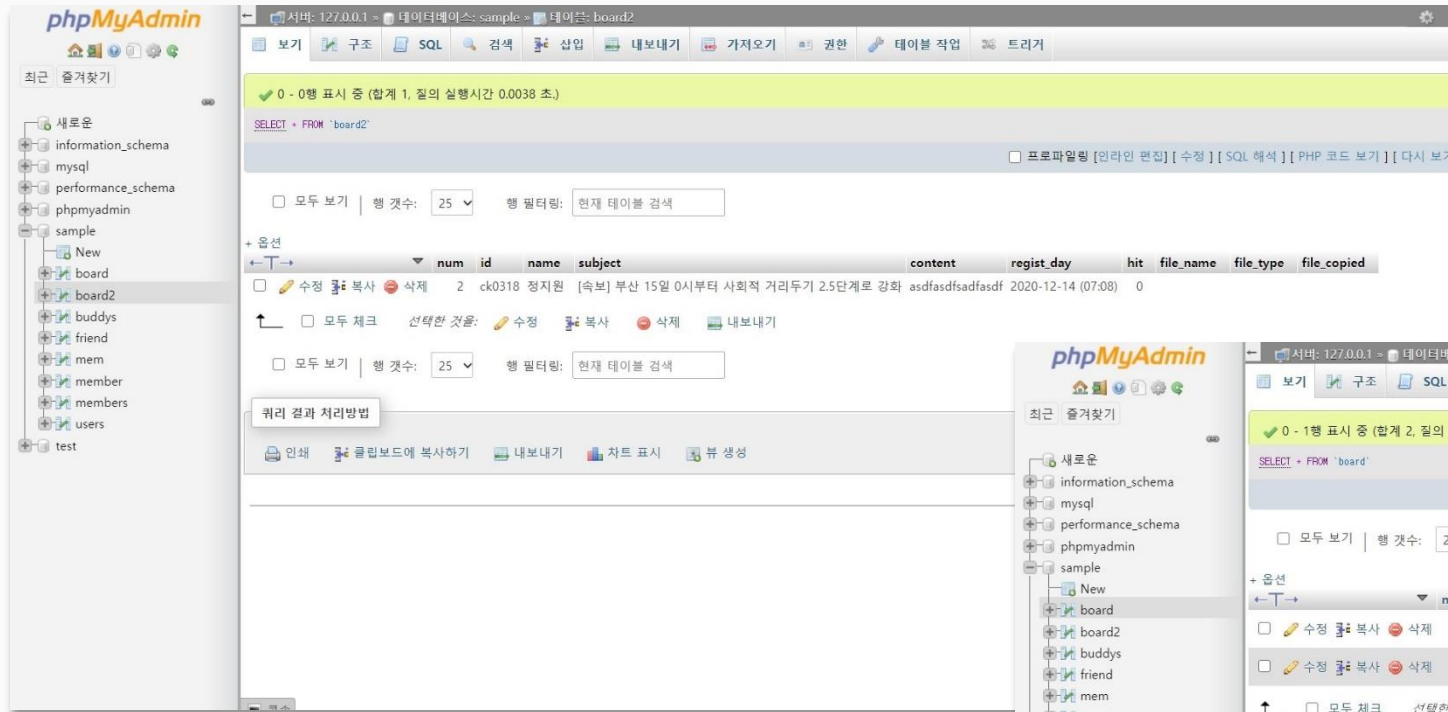
정지원 2020-12-14

회원 랭킹

1	정지원	ck0318	1300
---	-----	--------	------

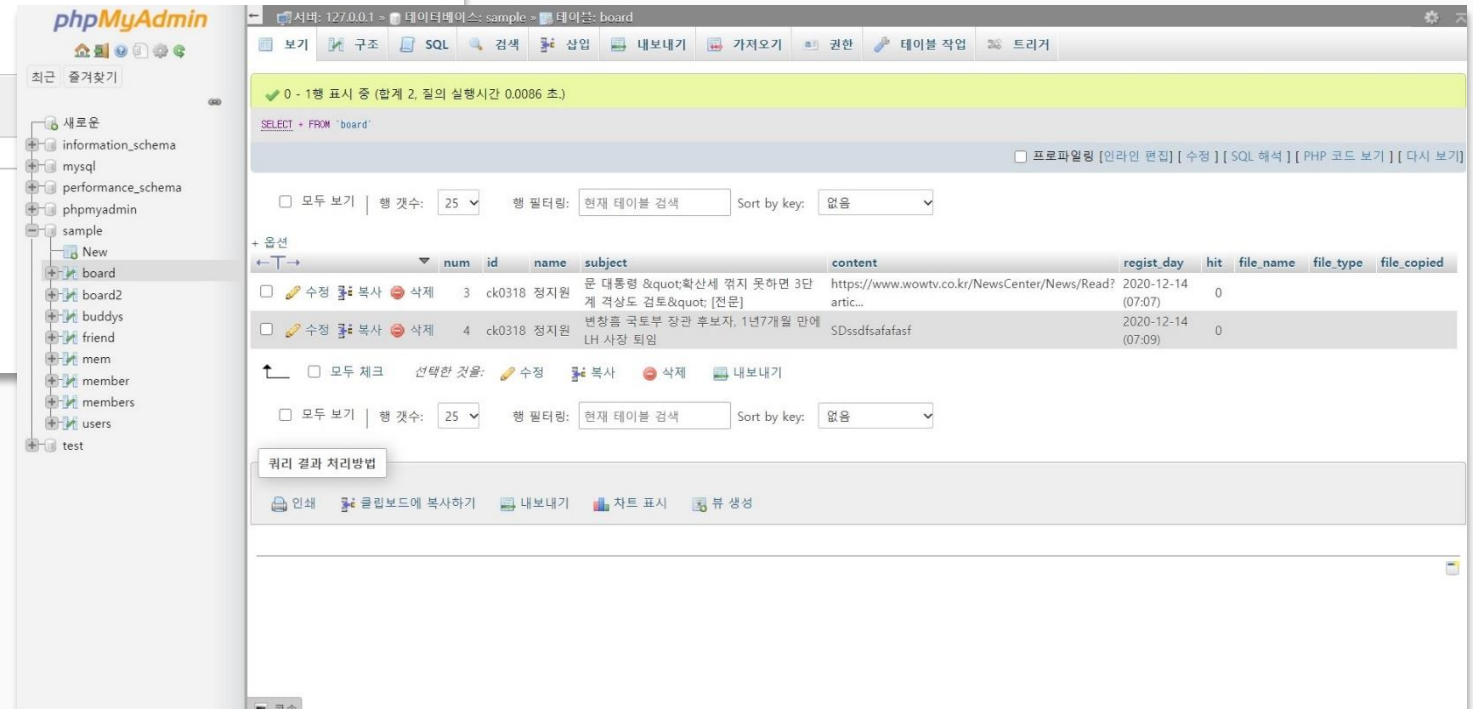
나중에 볼 기사를 저장하면 메인 페이지에 '아직 안 읽은 기사목록'에 보여진다.

06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지

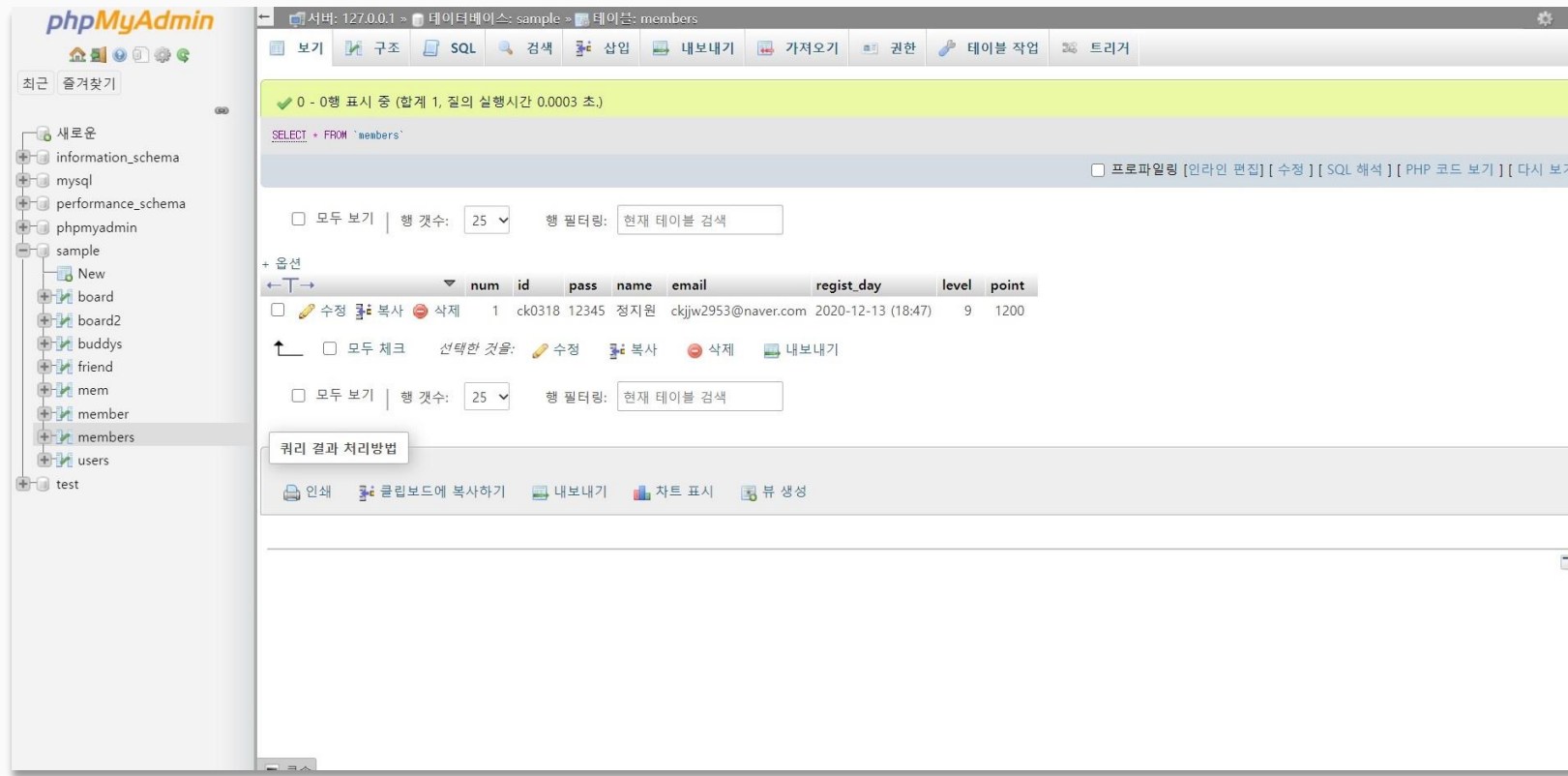


‘나중에 볼 뉴스 기사목록’ 서버에 적재

‘관심 뉴스 기사목록’ 서버에 적재



06. PHP 서버를 활용한 뉴스 스크랩 웹 페이지



회원가입 한 사용자의 정보가 서버에
적재된 모습을 확인할 수 있다.