



*"Our greatest glory is not in never failing, but in rising up every time we fail."*  
- Ralph Waldo Emerson

# Portfolio.

윤준석

빠른 기술 습득력과 열정을 발판으로,  
인공지능 역량과 웹 개발 역량을 가진 개발자

01

# Cat Swimming

## (머신러닝 기반 중고거래글 필터링 검색 웹사이트)

CAT SWIMMING

모두

닌텐도

검색

모두

카테고리	제목	작성자	작성일	광고
NDSL/WII/닌텐도	[미개봉] 닌텐도스위치 제노블레이드	직거래대환영(jcig****)	2020.06.19. 11:08	일반
NDSL/WII/닌텐도	(미개봉)닌텐도스위치 신형 판매	직거래대환영(jcig****)	2020.06.19. 11:08	일반
NDSL/WII/닌텐도	(미개봉)닌텐도스위치 동물의숲 에디션 (광주)	직거래대환영(jcig****)	2020.06.19. 11:07	일반
M 게임/취미/스포츠/레저	[공식앱][스위치 라이트 터과이즈(액보필름,파우치포함) + 마리오][280,000원]	dnrkdnrk(dnrk****)	2020.06.19. 11:07	일반
노트북/맥북/넷북	[매입]최고가에 구매중이에요!!	전자제품회사(kore****)	2020.06.19. 11:07	위험
NDSL/WII/닌텐도	닌텐도 스위치 네온 + 동숲 게임칩	Asaspap(awak****)	2020.06.19. 11:07	일반
NDSL/WII/닌텐도	닌텐도 스위치 구형 팝니다.	freemaim(free****)	2020.06.19. 11:06	일반
NDSL/WII/닌텐도	닌텐도 스위치 네온 신형(풀박), 파우치 일괄 판매합니다	하루마음(tls8****)	2020.06.19. 11:05	일반

프로젝트 기간 : 2020/03 ~ 2020/04

프로젝트 현황 : 개발 완료

# Cat Swimming (머신러닝 기반 중고거래글 필터링 사이트)

원하는 물품을 키워드 검색을 통해 검색한 뒤, RNN 기반의 머신러닝 모델을 통하여 게시글이 광고, 업자일 확률을 표기해주는 검색엔진 페이지.

#Flask #ReactJS #AWS #Python #TensorFlow #Docker

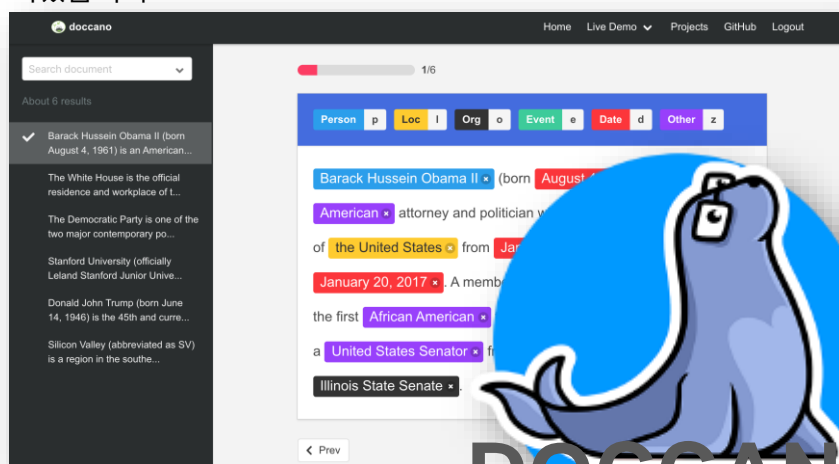
## 중고 거래의 새 이름 “Cat Swimming”

Cat Swimming 프로젝트는 전자제품 중고거래를 자주 하던 팀원들의 문제에 공감하여 기획되었으며, 저는 **Flask Backend, TensorFlow AI, AWS DevOps Engineer**를 맡았습니다. **Backend Engineer**로써, **키워드 검색 기능과 검색결과 크롤링, 네이버 카페 로그인 기능 등을 구현하였으며, AI Engineer로써, 빅데이터 라벨링 총괄 관리와, AI 모델의 추가 학습 및 학습 자동화 파트를** 맡았습니다. **AWS EC2 서버 상에 Docker를 이용해 배포**하기도 하였습니다.

이렇게 많은 역할을 맡을 수 밖에 없었던 이유는, 중간에 팀원 한명이 나간 것과, 코로나 유행으로 인한 지연된 일정 시간, 완전 비대면으로 진행된 원격 프로젝트로 진행되었기 때문이었으며, 다들 의사소통과 협업, 시간 부족 등의 문제를 겪었습니다. 하지만 **엄격한 스크럼 회의와 스프린트 관리, 화상 회의 프로그램의 도입** 등으로 최대한 효율을 뽑아내었고, 우리 프로젝트는 **높은 정확도와 실용성, 참신함**을 높이 평가받을 수 있었습니다.

아쉬운 점은 당시 일정의 촉박함으로, 기획했던 가격비교, 급락, 급등 상품, 당근 마켓 등 다른 중고거래 글 등의 통합 기능 등을 구현하지 못했다는 점이었습니다.

인공지능과 웹을 융합한 이번 프로젝트를 진행하면서, 저는 인공지능과 웹은 땀해야 땀 수 없는 필수적인 역량이라 느끼게 되었고, 이 프로젝트는 **인공지능 역량과 웹 개발 역량을 가진 개발자가 되겠다고 마음 먹게 된 계기**가 되었습니다.



DOCCANO

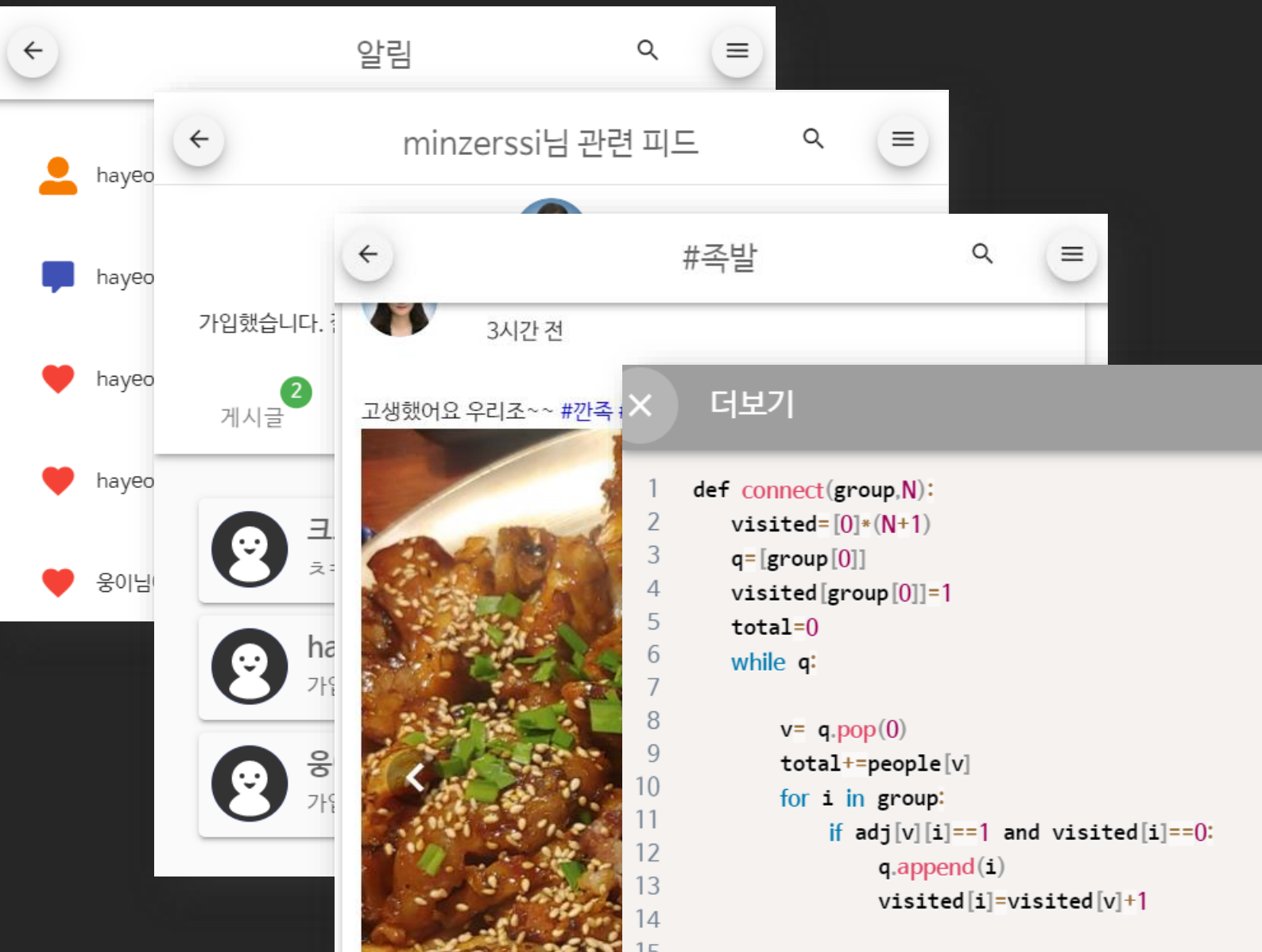
## 오픈 소스 데이터 라벨링 툴인 “DOCCANO”

당시 프로젝트의 **가장 큰 장벽은 질 좋은 데이터셋의 확보**였습니다. 중고 거래의 광고 글에 대한 공개 데이터셋은 존재하지 않았으므로, 5명에서 이를 남짓한 시간 내에 3만개가 넘는 거래글을 직접 레이블링하여 확보하는 수 밖에 없었습니다.

이때, 저는 **팀원들에게 해당 툴을 사용할 것을 설득하고, 서버상에 배포한 뒤, 사용법과 일정, 시스템 정보를 문서화하여 공유**하였습니다. 덕분에 **데이터 레이블링은 여유롭게 진행되어 문제를 쉽게 해결**할 수 있었습니다.

# 02

## DNS 개발자 커뮤니티 (개발자 친화적인 게시글을 지원 하는 웹 기반 SNS)



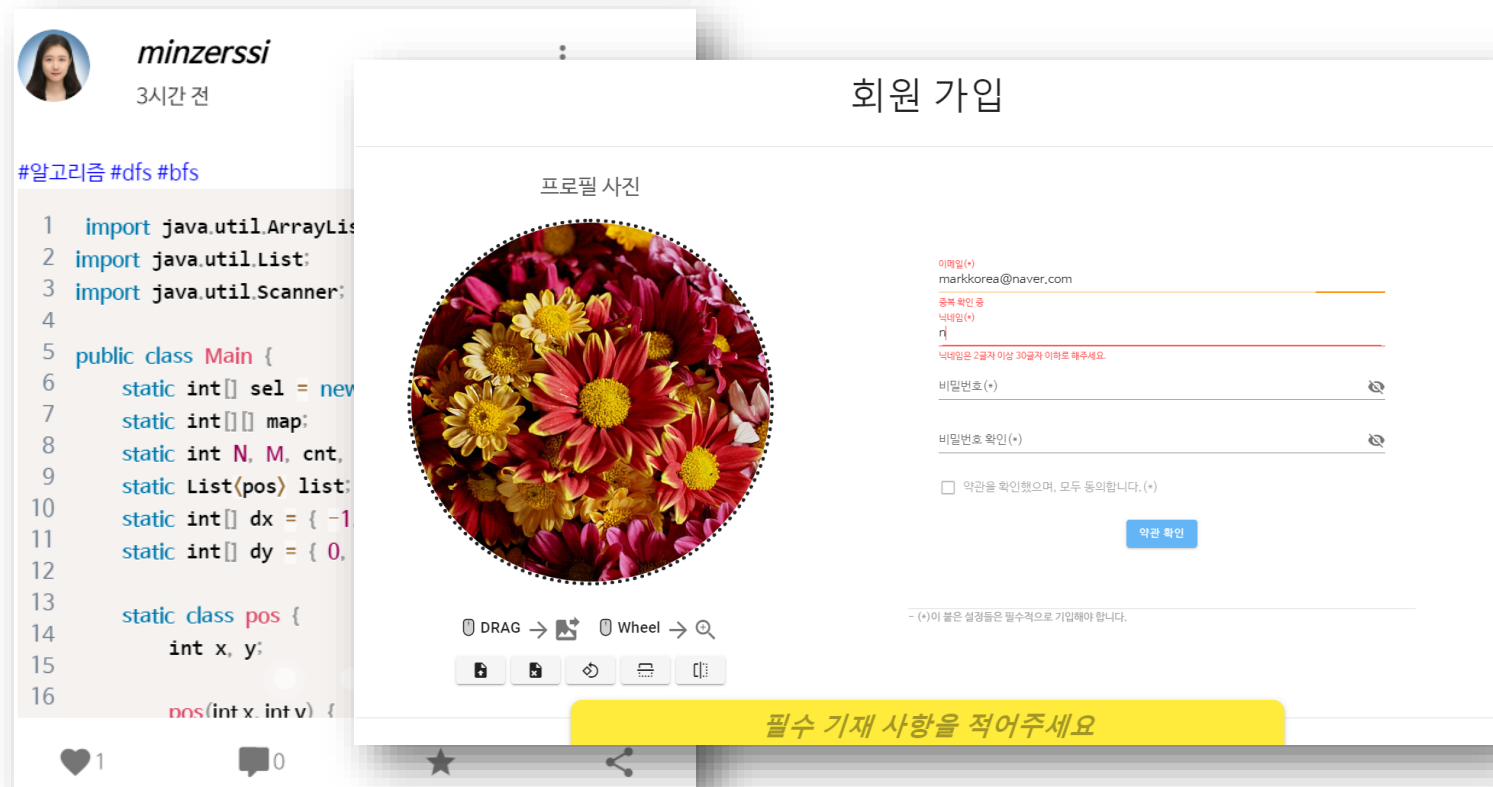
프로젝트 기간 : 2020/01 ~ 2020/02

프로젝트 현황 : 개발 완료

# DNS 개발자 커뮤니티 (개발자 친화적인 게시글을 지원하는 웹 기반 SNS)

해시태그 검색, 팔로우, 알림 기능 등, 고급 기능들을 구현하였으며, 게시글에 그림, 영상, 코드를 공유할 수 있게 만든 웹 기반 SNS

#Vue.js #Spring / Spring Boot #AWS #MySQL #Docker, Nginx



## 내 생애 첫 협업 IT 프로젝트,

단순히 기본적인 SNS 클론 프로젝트 보단, 참신하고 다른 사람들이 필요로 할 것 같은 SNS를 만들어 보자는 계기로 시작된 프로젝트입니다.

나는 이번 프로젝트에서 **팀장과 주 Vue 프론트엔드 엔지니어**, 부 Spring boot 백엔드 엔지니어를 맡았습니다.

**팀장**으로서 Git lab, Jira 등의 협업 툴 관리와, 애자일 스프린트, 스크럼 회의 등의 진행을 맡았으며, 주 Frontend engineer로서, Drawer, REST API 통신 전반, 회원 가입, 로그인, 해시 태그, 검색 기능과 유저 정보 페이지등을 맡았습니다. 그리고 부 Backend engineer로서, 유저 팔로우, 검색, 좋아요, 알림 기능 등의 REST API 기능과 DB 로직, ORM entity를 구현하였습니다.

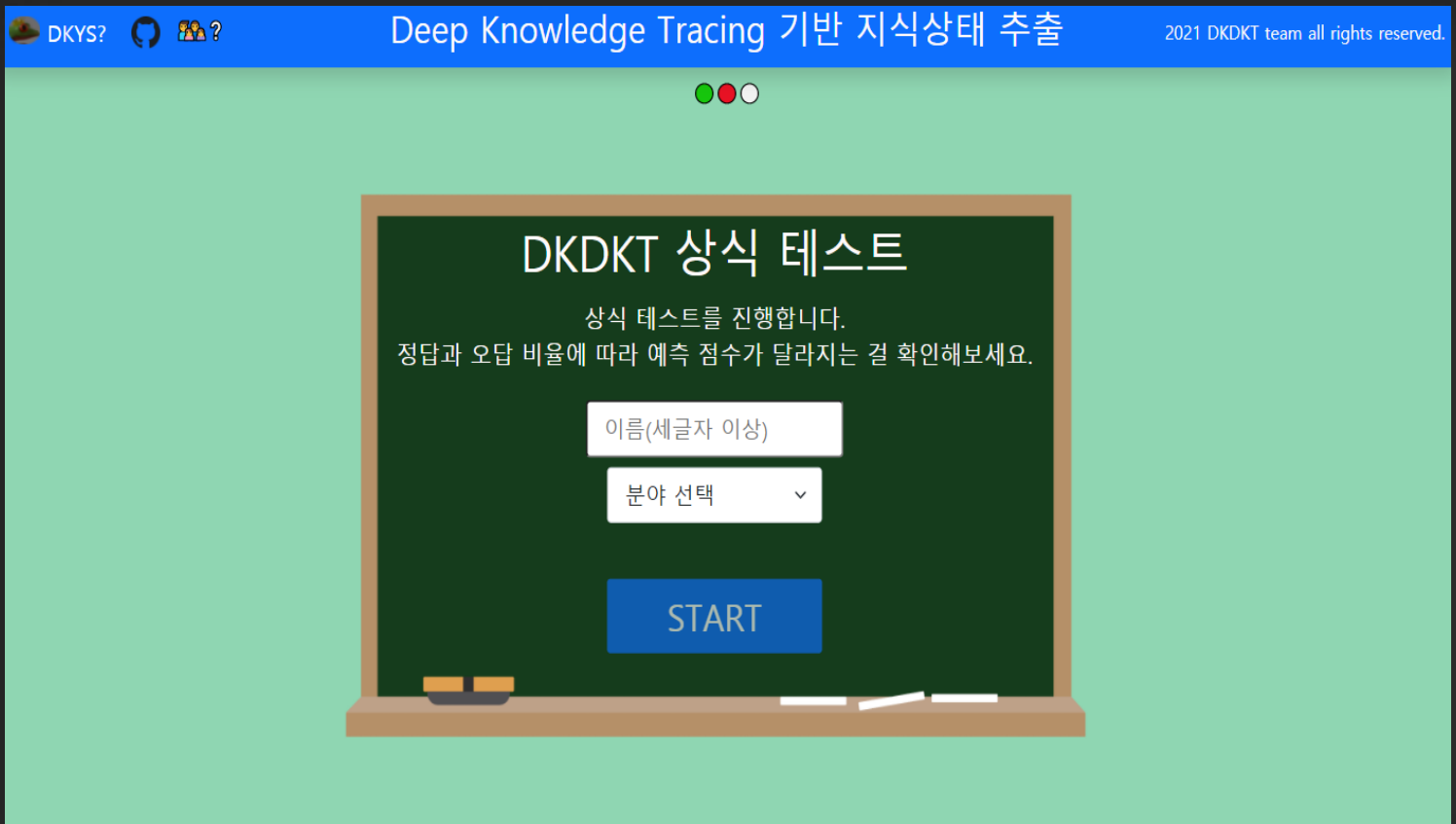
사실 이번 프로젝트는 **개발 기획이 중간에 뒤엎어졌는데**, 처음에는 심박수를 이용해 수면 리듬을 측정하는 임베디드 프로젝트를 진행하려 했으나, 첫 프로젝트로서 경험 부족, 지원되는 임베디드 장비의 한계, 교육 시간의 부족함, 생물학적 지식의 부재 등으로 기획을 바꾸게 되었고, **프로젝트 기획이 얼마나 어렵고 중요한 부분인지 깨달을 수 있었습니다.**

덕분에 다른 팀보다 시간이 촉박했지만 외부 라이브러리와 효율적인 신기술(기존에 사용하던 Bootstrap 대신 Vue 전용 프론트엔드 프레임워크인 Vuetify 도입)의 과감한 도입, 팀원 간 정, 부 직책을 나누어 책임감을 지게 하고, 빠르게 서로 교육하게 하여 일정을 앞당길 수 있었고, 상위 20% 정도의 높은 평가로 마무리할 수 있었습니다. 또한, 이번 프로젝트는 기존에 자주 사용해왔던 Django framework가 아닌 Spring boot를 사용하였는데, 아예 경험이 없던 기술을 빠르게 익힐 수 있는 요령과 자신감을 얻은 점이 좋았습니다.

첫 협업 프로젝트였지만, 팀원들의 추천으로 팀장이 되었으며, 팀원들과 일하기 전에 내심 실력과 경험이 비교적 부족하진 않을까 걱정했지만 오히려 프로젝트를 진행하면서 내 자신의 실력과 열정에 자신감을 가짐과 동시에 내게 필요한 역량을 알게 된 프로젝트였습니다.

# 03

## Deep Knowledge Tracing Competition



프로젝트 기간 : 2021/05/23 ~ 2021/06/22

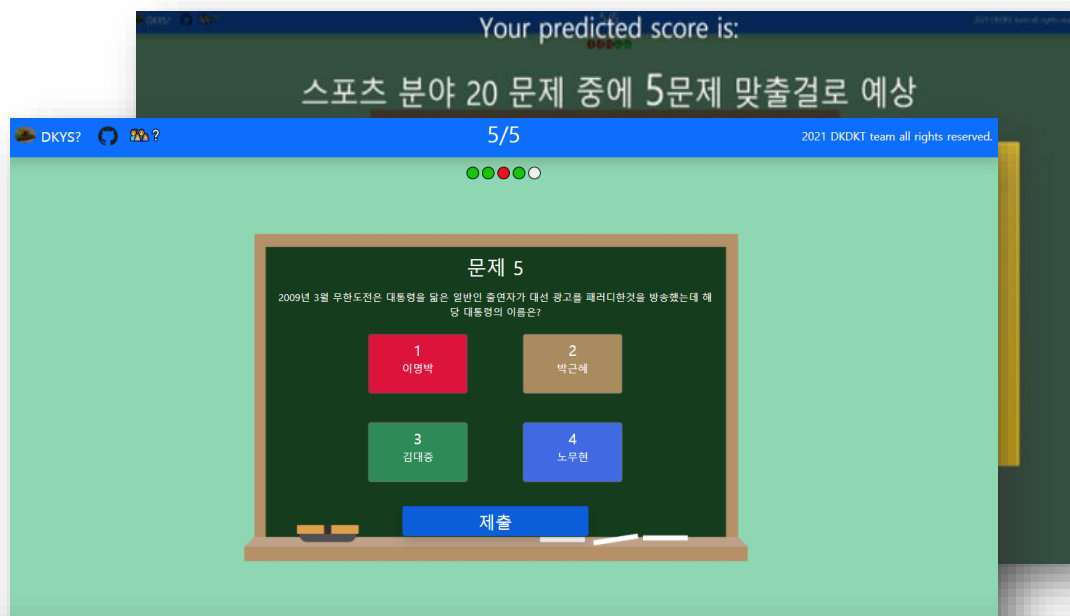
프로젝트 현황 : 개발 완료



# Deep Knowledge Tracing Competition

좋은 성과를 보이고 웹 서비스 구현, ML 모델 서빙까지 이어진 DKT Competition

#PYTORCH #DKT #Flask #javascript #LGBM #Git #Wandb

## Competition부터 Service까지, AI 비즈니스 전반을 경험한 프로젝트



Ranking	User (Team)	accuracy	auroc	Entries	Last
1	 DKT_6조	0.7581	0.8486	155	16 days ago
2	 DKT_1조	0.7527	0.8464	179	17 days ago

**Private 기준 최종 Competition 1 순위 달성!**

ODQA Competition 이후로, 두번째로 참여한 협업 머신 러닝 Competition이었습니다. Competition 간저는 **Baseline Code** 수정과 **HyperParameter Search**, **피처 엔지니어링** 그리고 **DKT** 모델을 사용한 **웹 서비스 구현** 등의 역할을 맡았습니다.

이번 Competition을 통해 **단순 머신 러닝 모델링** 뿐만 아니라, **모델 라이프사이클**, **데이터셋 수집 및 가공**, **최적화** 등을 포함한 **AI 비즈니스의 전반적인 파이프라인**을 경험하게 되었고, AI 풀스택 개발자가 되기로 완전히 마음 먹었습니다.

구체적으로 이야기해보자면, 최종 모델은 가장 단일모델 성능이 좋았던 **Tabnet**, **LGBM** 등의 **트리 계열 모델들을 blending 방법으로 Ensemble**하여 최종 점수를 제출하였고, **LSTM attention**, **SAINT**, **Last query Transformer** 등과 같은 딥러닝 모델은 남은 시간이 부족하여 Ensemble하지 못하였습니다. 그렇게 우리가 제출한 Submission은 **Public 점수 기준 7등** 이였고, 중간 정도의 성적을 냈었습니다. 평범한 결과였지만, 서로에게 박수를 보냈었으나, 놀랍게도 Private 점수가 공개된 그날 밤 **Shakeup이 발생하여 우리 조가 1등으로 치솟았습니다.**

**Validation set**을 **Feature class** 분포대로 잘 나눈 것이 **최고 비결**로 뽑았으며, 사기가 고무된 우리 팀은 이 후, 3일 밤낮으로 **협업하여 위와 같은 서비스까지 구현**해보게 되었습니다.

# 04

## Subbrain Blog (개인 지식 기반 시스템 연동 블로그)

luminous-bubblegum-8e9be4.netlify.app/2022/08/10/1-컴퓨터-시스템의-소개.html

중증 컴퓨터 하드웨어는 크게 프로세서, 메모리(기억장치), 주변 장치로 구성되고, 시스템 버스로 연결된다.

### 1 프로세서(processor)

중앙 처리 장치(CPU, Central processing unit)이라고도 하며, 컴퓨터 하드웨어에 부착한 모든 장치의 동작을 제어하고 명령을 실행한다.

**프로세서의 수에 따른 병렬 처리**

프로세서의 수가 많을 수록 병렬 처리로 처리속도를 높일 수 있다.  
프로세서의 수에 따라 1개인 싱글 코어부터 8개인 옥타 코어까지 다양하다.

그림 1-2 프로세서의 구성

프로세서는 그림 1-2와 같이 연산장치와 제어장치, 레지스터로 구성되고, 내부 버스(시스템 버스)로 연결됨.

레지스터(Register)는 다음과 같이 구분한다.

- 용도에 따라 구분
  - 전용 레지스터
  - 범용 레지스터
- 사용자의 정보 변경 가능 여부에 따라 다음과 같이 사용자 가시 레지스터와 사용자 불가시 레지스터로 구분한다.

**사용자 가시 레지스터의 구분**

- I'm a Fullstack web developer wannabe. 🌟
- My goal is the innovations ✨ about WEB and AI.
- I'm passionate about learning new techs. ❤️
- I love playing games 🎮. // especially indie games.
- I love my families, friends and you! 🍀

GITHUB ROADVIURSHN BLOG ROADVIURSHN PORTFOLIO 한국어(KOREAN)

프로젝트 기간 : 2022.08 ~

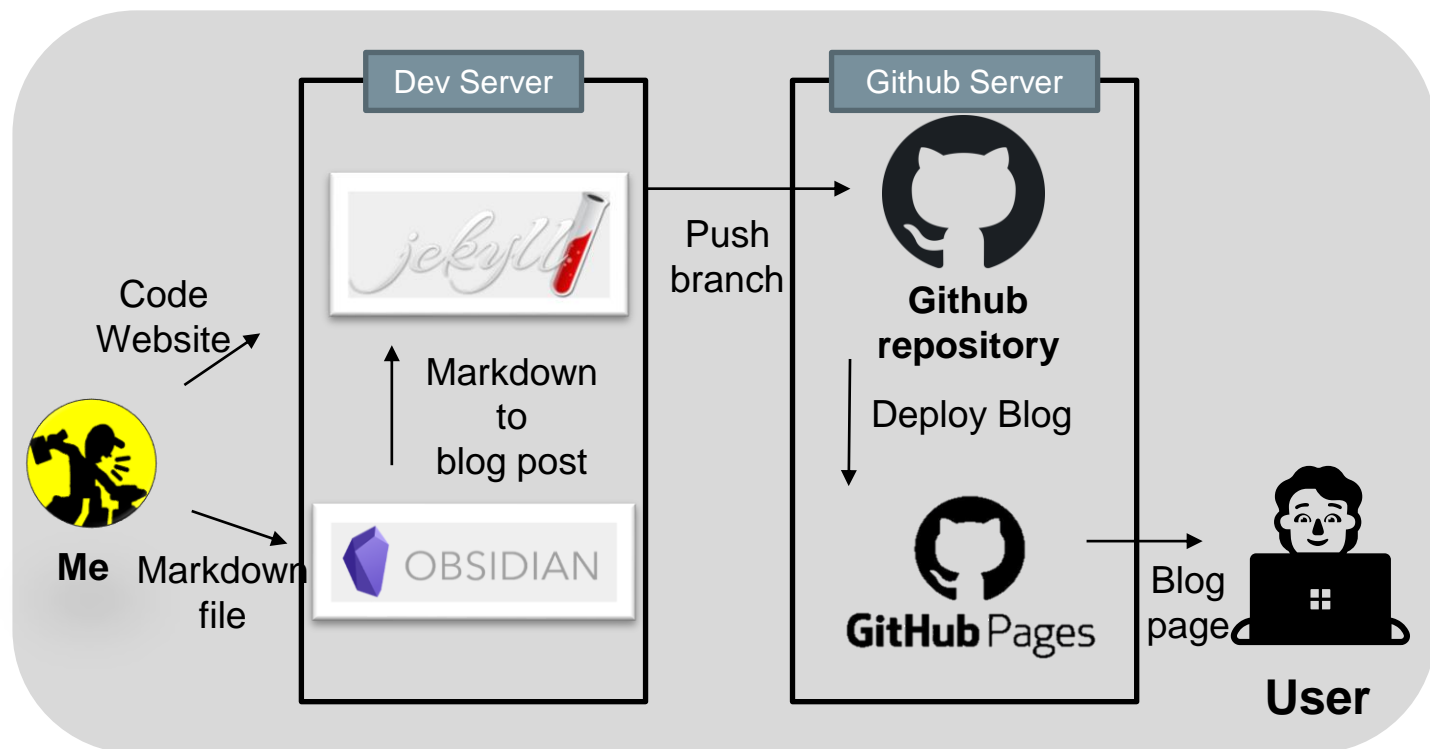
프로젝트 현황 : 개발 진행 중



# Subbrain Blog (개인 지식 기반 시스템 연동 블로그)

Obsidian 마크다운 포맷에 맞는 블로그 포스트와 추가적인 기능을 포함한 특별한 개인 블로그

#Jekyll #Typescript #Webpack #Sass



## 디지털 가드닝 입문을 위한 개인적 필요성에 의해 시작된 개인 프로젝트.

디지털 가드닝이란 자신의 지식을 블로그 포스트 형식으로 사람들에게 공개한 뒤, 이들을 가드닝 하듯이 주기적으로 관리하는 지식 관리 방법입니다. 효율적인 개인 학습 방법에 대해 연구해보는 와중에, **디지털 가드닝을 통해 정확한 학습, 내용기록, 의욕 상승, 객관적인 지식 수준 측정, 개인 브랜딩** 등을 이루어낼 수 있을 것이라 생각한 저는 **개인 지식 기록을 쉽게 관리하고 다른 사람들에게 보여줄 수 있는 특별한 개인 블로그 개발**을 시작했습니다.

처음에는 평범한 3-티어 구조의 웹 페이지를 생각했지만, 구현하려는 기능과 예상 방문자에 비해 지나치게 많이 드는 서버 비용과 개발 시간과 새로운 기술을 경험해보고 싶은 마음, 정적 사이트 생성 툴을 이용하면 기존에 마크다운 파일로 기록했던 TIL들을 쉽게 포스트로 변경할 수 있을 것 같아서 정적 사이트 프로젝트로 변경하였습니다.

이 프로젝트를 통해 **SASS, Webpack, Typescript, Ruby, Jekyll, lunr.js** 등 여러 기술을 습득할 수 있었고, 정적 사이트, SSR, 브라우저 API 등 여러 개념을 터득할 수 있었습니다. 또한 디지털 가드닝이 익숙해지면 더욱 효율적이게 학습하고 학습 내용을 쉽게 동료들과 나눌 수 있게 될 것 같습니다.

### ABOUT ME

Hi 🤖, I'm **JUNSEOK YUN.**  
a.k.a **RoadvyrushN.**  
I fell in ❤️ with **WEB** programming and I'm also intersted in **AI** too.  
discord: roadvyrushn Email: roadvyrushn@protonmail.com // contact me  
My experiences // Details in github, portfolios, and blog.

Still curious?  
• **MOTIVATION** // though I don't mind it.  
• A nerd from south koreaks.  
• I love playing games 🎮 // especially indie games.  
• I love my families, friends and you! 🥰

### ROADVYRUSHN

폴스택 웹 개발자 지망생  
+ 인공지능 관심

Categories

- COMPUTER SCIENCE (0:4)
- DATABASE (0:1)
- ALGORITHM (0:24)
- OPERATING\_SYSTEM (0:13)
- NETWORK (0:4)
- WEB (0:2)
- FRONTEND (0:2)
- BACKEND (0:7)
- CL/CD (0:4)
- ETC (0:4)
- OBSIDIANS (0:1)
- TESTS (0:1)
- OBSIDIAN\_TEST (0:2)

Recent views

1. 컴퓨터 시스템의 소개 - 윤영재 학습 정리

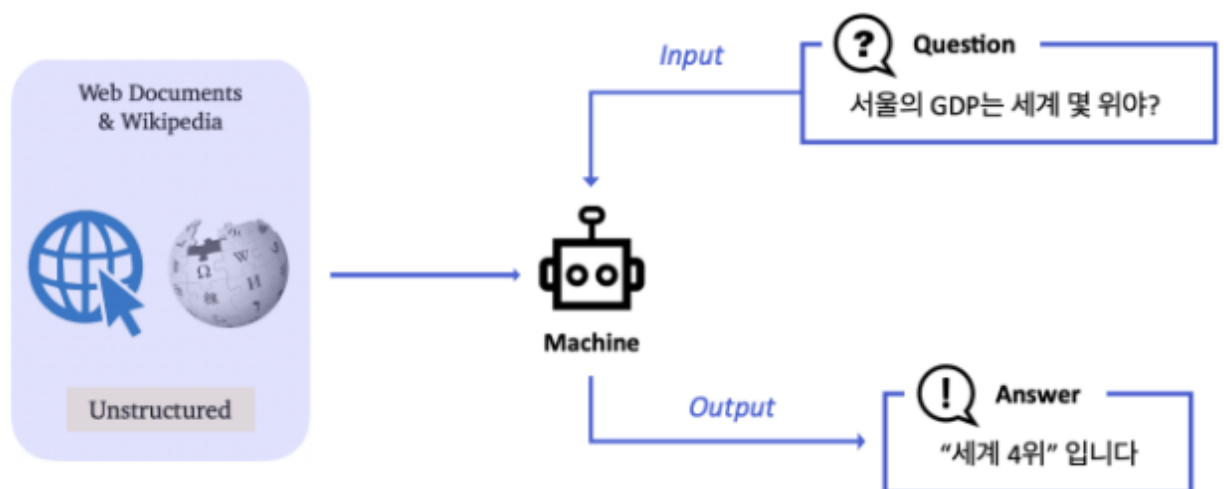
GITHUB ROADVYRUSHN BLOG ROADVYRUSHN PORTFOLIO 윤준석(KOREAN)

# 05

## Open domain Question Answering Competition

### Linking MRC and Retrieval: Open-domain Question Answering (ODQA)

ODQA: 지문이 따로 주어지지 않음. 방대한 World Knowledge에 기반해서 질의응답



프로젝트 기간 : 2021/04/26 ~ 2021/05/21

프로젝트 현황 : 개발 완료

# Open domain Question Answering Competition

NLP 전반의 파이프라인을 경험해볼 수 있었던 첫 협업 Competition

#PYTORCH #HuggingFace #NLP #Git #Wandb

## 머신 러닝 첫 협업 프로젝트

머신러닝에 대한 교육을 받기 위해 참여한 **Naver Boost AI Camp**에서 함께 교육을 받던 동료들과 **자연어 처리 Competition**에 팀을 이루어 참여하였습니다.

Competition 동안, **모델 튜닝, 데이터 전처리(EDA), HyperParameter Search, Tokenizer 변경, Retrieval Embedding 구현, 코드 리팩토링까지 프로젝트 기간동안 모든 부분에서 활동을 하고, 팀원과 활발한 지식교류를 이루었습니다.** 전체적인 파이프라인과 Competition 협업에 대해 배울 수 있던 좋은 시간이었습니다.

```
def compute_metrics(p: EvalPrediction):
    return metric.compute(predictions=p.predictions, references=p.label_ids)

print("Init trainer...")
# Initialize our Trainer
trainer = QuestionAnsweringTrainer(
    model=model,
    args=training_args,
    train_dataset=None,
    eval_dataset=eval_dataset,
    eval_examples=datasets['validation'],
    tokenizer=tokenizer,
    data_collator=data_collator,
    post_process_function=post_processing_function,
    compute_metrics=compute_metrics,
)

logger.info("**** Evaluate ****")

# eval dataset & eval example - will create predictions.json
if training_args.do_predict:
    predictions = trainer.predict(test_dataset=eval_dataset,
                                test_examples=datasets['validation'])

    # predictions.json is already saved when we call postprocess_qa_predictions(). so there is no need to further use pre
    print(
        "No metric can be presented because there is no correct answer given. Job done!"
    )

if training_args.do_eval:
    metrics = trainer.evaluate()
    metrics["eval_samples"] = len(eval_dataset)

    trainer.log_metrics("test", metrics)
    trainer.save_metrics("test", metrics)

__name__ == "__main__":
    # 가능한 arguments 들은 ./arguments.py 나 transformer package 안의 src/transformers/training_args.py 에서 확인 가능합니다.
    # --help flag 를 실행시켜서 확인할 수 도 있습니다.
    parser = HfArgumentParser( # hint 만들어주는 것인듯?
        (ModelArguments, DataTrainingArguments, InferenceArguments)
    )
    model_args, data_args, inf_args = parser.parse_args_into_dataclasses()
    main(model_args, data_args, inf_args)
```

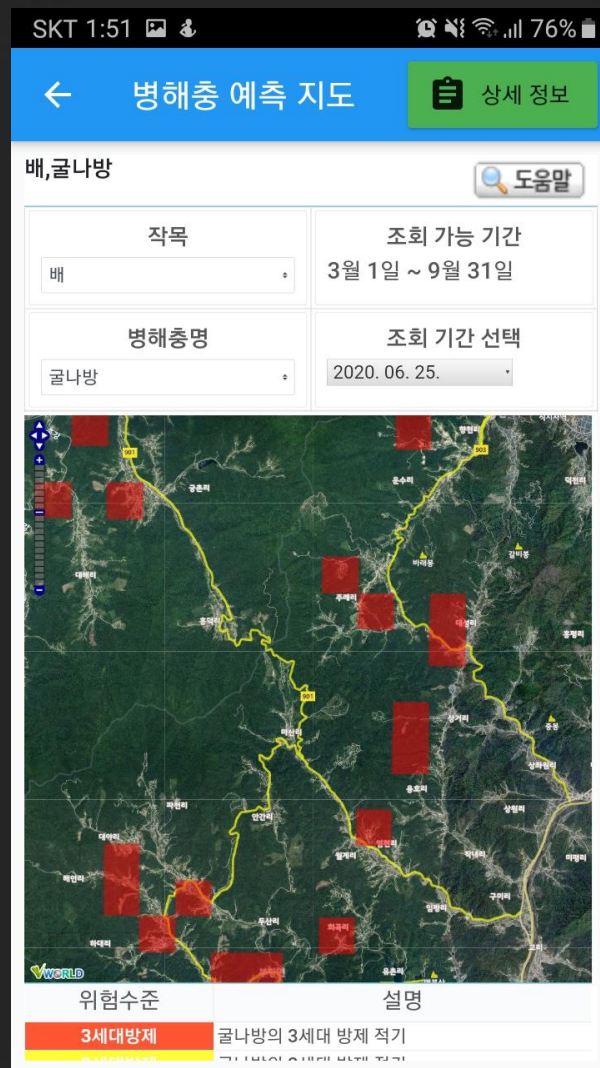
이전에 이미 Computer vision이나 가벼운 문장내 관계 추론 Task를 개인으로써 Competition에 참여한적이 있었지만, 미숙하고, 처음이라 학습에 성향이 강하였고, 실제로 제대로 된 **자연어 처리 협업 프로젝트는 팀원도, 나도 처음**이었으므로, 모두 처음 며칠 간은 우왕좌왕 하였습니다. 실제 머신러닝 파이프라인과 강의와 과제를 통해 배웠던 과정이 상당히 다름을 깨달았습니다. 구체적으로 Extraction based ODQA Task 였으며, 데이터셋은 KorQuad 1.0과 대회에서 주어진 데이터셋을 사용했으며, 모델은 HuggingFace에서 지원하는 Koelectra-korquad pretrained 모델을 활용하였으며, 토큰라이저는 카카오 Pororo 라이브러리와 Mecab을 활용하였습니다.

**결과적으로 목표했던 EM 기준 60%대의 성능은 도달하지 못하였고, 49% 성능으로 마쳤습니다.** 대회가 끝나고 회고해본 결과, Vocabulary의 부족, Korquad로 Pretraining된 모델이 Korquad 데이터셋에 대하여 높은 성능을 보이는 것을 실제 성능으로 오판하여, Retrieval 모델에 되려 신경을 너무 많이 쓴 것 등이 문제였습니다.

**아쉬움이 남았지만 Tokenizer부터, Retrieval 원리, 데이터 전처리, 분석 노하우, 머신러닝에서의 협업 방법, Wandb를 이용한 Hyperparameter Search와 데이터 분석, Competition을 위한 Baseline 코드 작성 등 정말 많은 것을 배웠으며, 앞으로 더 잘해나갈 자신이 생겼습니다.**

# 06

## 병해충 예측 정보 안드로이드 앱 (공공 API 활용 병해충 예측, 농약 정보 안내 앱)



프로젝트 기간 : 2020.05 ~ 2020.06

프로젝트 현황 : 개발 완료

# 병해충 예측 정보 안드로이드 앱 (공공 API 활용 병해충 예측, 농약 정보 안내 앱)

국가 농작물 병해충 관리 시스템에서 제공하는 공공 API를 이용하여 병해충 예보 정보를 검색 한 뒤, 관련 병해충 정보와 농약 정보를 얻을 수 있는 Flutter Cross-Platform 앱

#Flutter #Django #Docker #AWS

외부 API 활용과 모바일 어플리케이션 등, 도전이 주제였던 프로젝트,

당시, 다들 취직을 위한 자격증이나 이력서 등에 시간을 쏟으면서, 프로젝트를 진행하지 않거나, 허울 뿐인 팀이 많았지만, 저는 아직 경험이 부족하다고 판단하여 프로젝트를 원하는 팀원들을 모아서 진행한 프로젝트였습니다.

저는 이번 프로젝트에서 **Android Engineer**와 **Backend Engineer**, **DevOps Engineer**를 맡았습니다. **Android Engineer**로서, Flutter를 활용해 Cross-platform 앱을 구현하였으며, 특히 병충해 예측 지도 부분과 메인 화면, Drawer를 담당하였습니다. **Backend Engineer**로서, Django에서 웹 뷰를 구성하는 역할을 하였고, Docker와 AWS를 통하여 배포하였습니다.

문제가 심각했는데, 한달 반 정도의 프로젝트 기간 중에 무려 한달간 공공 API 키 발급을 기다려야 했고, 독촉에 독촉을 통해 얻어낸 API는 json 데이터가 아닌 Java Servlet 기반 Webview를 통째로 보여주는 것이라 가공할 수 없었습니다. 관련 담당자들에게 문의하였지만, 담당이 바뀌어 잘 모르겠다는 답변이 돌아왔었고, 저희는 **시간 부족과 경험 부족**에 쫓기게 되었습니다.

하지만 앱 안에 웹 뷰를 만드는 방식으로 해결했고, 기타 다른 농약, 해충 정보들은 다른 API를 활용해 가져오는 것으로 해결하였습니다. 또한, 외부 인터넷 강의와 교보재를 적극 활용하여 새로운 기술과 경험을 습득하여 경험 부족을 해소할 수 있었습니다.

Cross-platform 어플리케이션 프레임워크인 Flutter를 활용했음에도, 일정과 경험부족으로, IOS로는 포팅하지 못한 점이 너무 아쉬웠습니다. 또한, 해당 어플리케이션을 구글 플레이 마켓에 올리려 했지만, 진행한 프로젝트는 외부 공개가 불가능하다는 교육 주관사와의 계약 관계로 올리지 못하였습니다.

이번 프로젝트는 팀원들 뿐만 아니라 외부와 협업할 시 고려해야 할 점과, 어려움을 깨닫게 되었고, 모바일 어플리케이션 개발 경험과 공공 API를 활용하는 경험을 가져볼 수 있는 좋은 프로젝트였습니다.

관련 API 사용 기간이 중단되어 더 이상 서비스가 불가능합니다.

