CHAINE PORTFOLIO

<mark>혁신적인</mark> 개발자를 꿈꾸는 김태후입니다.

LIST OF CONTENTS

Ol ABOUT ME

02 TECH STACK

03 PROJECT

in goding ABOUT/ME

협업 능력이 뛰어나고 책임감 있는 개발자, 김태후입니다.

- 새로운 것을 두려워하지 않고 지속적인 성장을 원합니다.
- 목표를 성취하기 위한 열정이 차별화된 강점입니다.
- 협업을 위한 코드 품질을 높이는 데 항상 노력합니다.
- 팀과 조직의 소속감을 중요시하며, 책임감 또한 매우 강합니다.



김태후 Software Developer









언어 / 프레임워크

C/C++
Java/JavaScript
Python
Dart/Flutter
Kotlin

데이터베이스

MySQL



Apache AWS EC2

기타

Visual Studio Code, Visual Studio IntelliJ IDEA, Eclipse FireBase, REST Git, Discord

EasyTrip! 여행 코스 추천 AI 애플리케이션

Poject PORTFOLIO

분산 파일 시스템분산 파일 공유 및 백업 시스템

 악성 BOTNET 탐지

 악성 네트워크 트래픽 탐지 시스템

EasyTrip!

- 사용자의 맞춤형 서비스를 제공 - 자연어 처리를 통한 텍스트 입력 방식 - AI 모델을 이용해 조건에 맞춘 코스 추천

여행 코스 추천 요청



추천 여행 코스 출력

기간 / 인원 2024.03 ~ 2024.07 (5개월) / 3명

기술 스택 [프론트엔드] Kotlin

[백엔드] Kotlin, Python

[3rd Party] Firebase Auth, SNS Login API

[협업 툴] Slack, Github

담당 기능

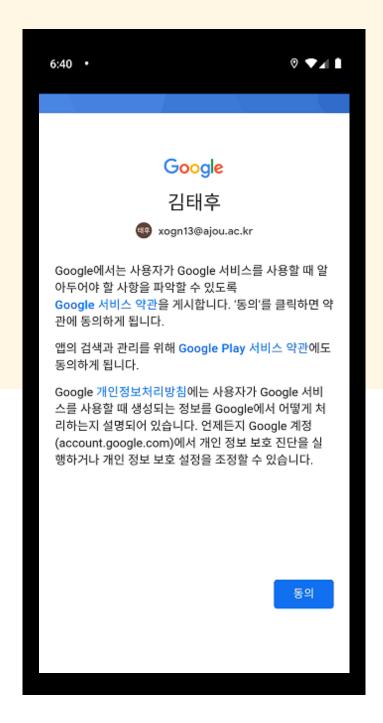
- 프론트엔드 설계 및 구현
- Firebase API를 활용한 인증과 SNS API를 활용한 SNS 로그인, 회원가입 구현
- RESTful API 설계 및 개발
- Python Al Model 개발

배운 점

- 여러 API를 설계하고 연동하며 API에 대한 지식을 얻을 수 있었습니다.
- Python Al Model을 팀원들과 함께 개발하면서 커뮤니케이션과 협업의 중요성을 깨달을수 있었습니다.

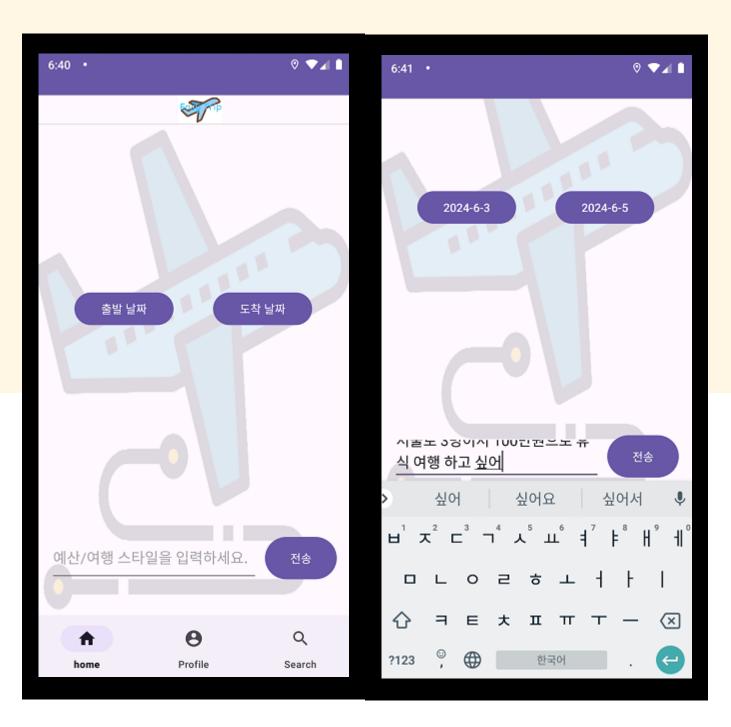
어려웠던 점

팀원들이 모바일 애플리케이션 개발 경험이 미흡하여 Kotlin 개발을 전담하는 상황이 생겼지만, 이 기회에 기획과 개발을 전담해보자는 생각으로 극복하였고, 이로 인해 관련 지식에 대해 많은 것을 배웠습니다.



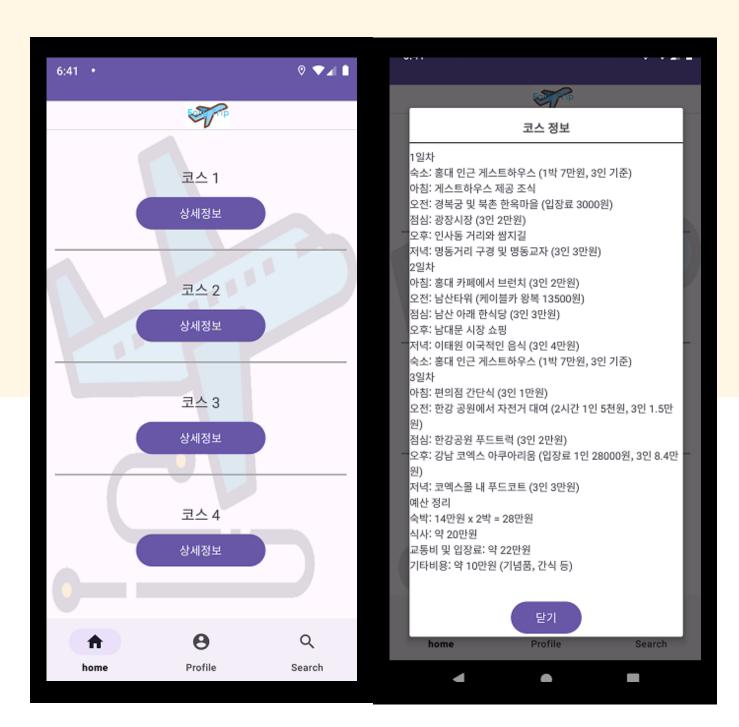


API를 활용하여 간편 로그인 구현



자연어 입력 처리

사용자는 출발 날짜와 도착 날짜를 선택하고, 자연어로 여행 스타일을 입력하여 프로그램에 전달.

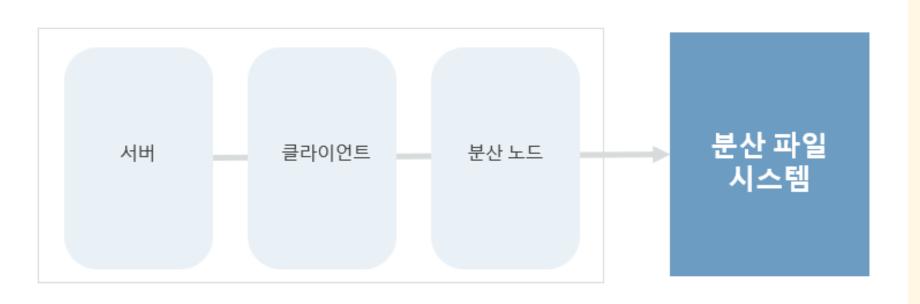


여행 코스 제공

사용자에게 입력 받은 날짜와 자연어를 바탕으로 맞춤화된 여행 코스를 화면에 출력.

분산 파일 공유 및 백업 시스템 개발

- 분산 파일 시스템을 구축하여 여러 사용자 간의 파일 공유 및 백업을 효율적으로 수행
- 데이터의 안전한 저장 및 신속한 접근 보장
- 시스템의 확장성과 고가용성 확보



기간 / 인원 2024.03 ~ 2024.07 (5개월) / 1명

기술 스택 [백엔드] Python [3rd Party] Flask

담당 기능

- Consistent Hashing 알고리즘을 사용하여 대용량 데이터를 안전하게 저장하고 공유할 수 있도록 시스템을 구축하였습니다.
- Flask로 REST API을 구현하였습니다.

배운 점

- 노드 간 데이터 전송 시 발생할 수 있는 병목 현상을 해결하기 위해 알고리즘을 최적화하는 방법을 배웠습니다.
- 분산 파일 시스템을 설계하며 대규모 데이터 관리 시스템이 실무에서 어떻게 동작하는지 이해할 수 있었습니다.
- Consistent Hashing 알고리즘을 사용하며 데이터의 일관성과 무결성을 유지하는 원리를 배울 수 있었습니다.

악성 BOTNET 탐지

- 네트워크 내의 악성 트래픽을 탐지하기 위해 방대한 데이터 를 추출, 가공하여 이상 패턴을 발견하고 이를 분석.
- 전처리 과정과 ML과 DL의 작업을 마치고 시스템을 구축.

	precision	recall	f1-score	support
0 1	1.00 0.66	0.99 1.00	0.99 0.80	1884766 38110
accuracy macro avg weighted avg	0.83 0.99	0.99 0.99	0.99 0.90 0.99	1922876 1922876 1922876
F1-Score: 0.7973553995669048 AUC: 0.9948611657892811 Accuracy: 0.9899260274713502				

탐지 성능 일부: TabNet (DL)

기간 / 인원 2023.03 ~ 2024.07 (5개월) / 4명

기술 스택 [백엔드] Python

담당 기능

- 데이터 전처리
- ML (Gradient Boosting, Random Forest, Ensemble)
- DL (TabNet)

배운 점

- 네트워크 트래픽 데이터를 분석하며 정상 패턴 과 비정상 패턴을 구분하는 과정에서, 데이터의 정제와 특징 추출을 하며, 네트워크 보안과 데이터 분석이 밀접하게 연결되어 있음을 깨달았습니다.
- 머신러닝에서 과적합을 방지하기 위해 하이퍼 파라미터 튜닝이 중요하다는 것을 알 수 있었습 니다.
- 처음 접해본 딥러닝 TabNet 알고리즘을 적용 하면서 관련 지식을 넓힐 수 있었습니다.



Random Forest 결과



Gradient Boosting 결과



Ensemble 결과

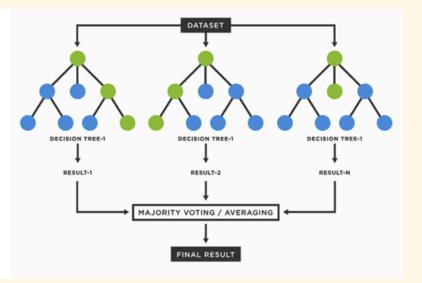
Random Forest Classifier

Accuracy: 0.9925591550378985

Precision: 0.7937510118180346

Recall: 0.6893982002249719

F1: 0.7379035292347054



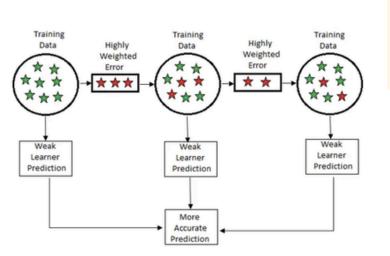
Gradient Boosting Classifier

Accuracy: 0.9860326602462763

Precision: 0.5483816587997303

Recall: 0.4573959505061867

F1: 0.4987733823980375



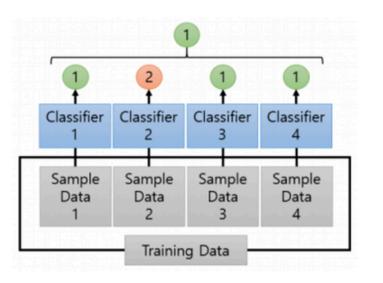
Ensemble

Accuracy: 0.9921041162848329

Precision: 0.7949913644214163

Recall: 0.6472159730033745

F1: 0.7135327856146334



THANKS FOR WATCHING