



브랜드 검색 엔진

상표권 유사도 판단 모델

경기도 X Google 빅데이터, AI 전문인력 양성과정

APPTeam 1조 어?금지

원하선, 김지우, 이예진, 이재희

프로젝트 기간 : 23.11.17 - 30



Team 어?금지

- 원하선
 - 설계, PL
 - VISION search
 - glashuitte@gmail.com
 - 010-3938-2902
- 김지우
 - AI 개발
 - Vertex AI, AutoML 모델링
 - github.com/liljw
 - harutency@naver.com
 - 010-5237-5315
- 이예진
 - 풀스택 개발
 - 제품 개발 총괄
 - geniest92@gmail.com
- 이재희
 - PRM
 - 기획, 데이터 CRUD
 - hello.jhlee0@gmail.com

프로젝트 진행

01 기획

- 선정 배경
- 기대효과
- 멤버별 목업

02 기능

상표권 분류 : AutoML

- 데이터 구축
- 데이터 라벨링
- 데이터 전처리

유사 로고 검색 : Vision

- 개요
- 프로세스

03 배포

웹앱

- DataFlow
- 제품
- 최소기능제품

서버리스

- GCP를 통한
서버리스 앱
개발 장점

04 출시

- 제품 시연

Markit을 소개합니다.

상표권을 주제로 선택한 이유



상표권을 주제로 선택한 이유

상표 출원 / 등록 성공 비율 2023년 10월 기준

월 출원 건수

21,000건

월 평균 등록 성공 건수

15,000건

심판 청구

324건

30%

등록 실패



25%

유사성으로
인한 실패



상표권
침해

*30% 등록 실패 : https://www.ajupatent.com/ko/ip_report/ip_news/?type=view&idx=82

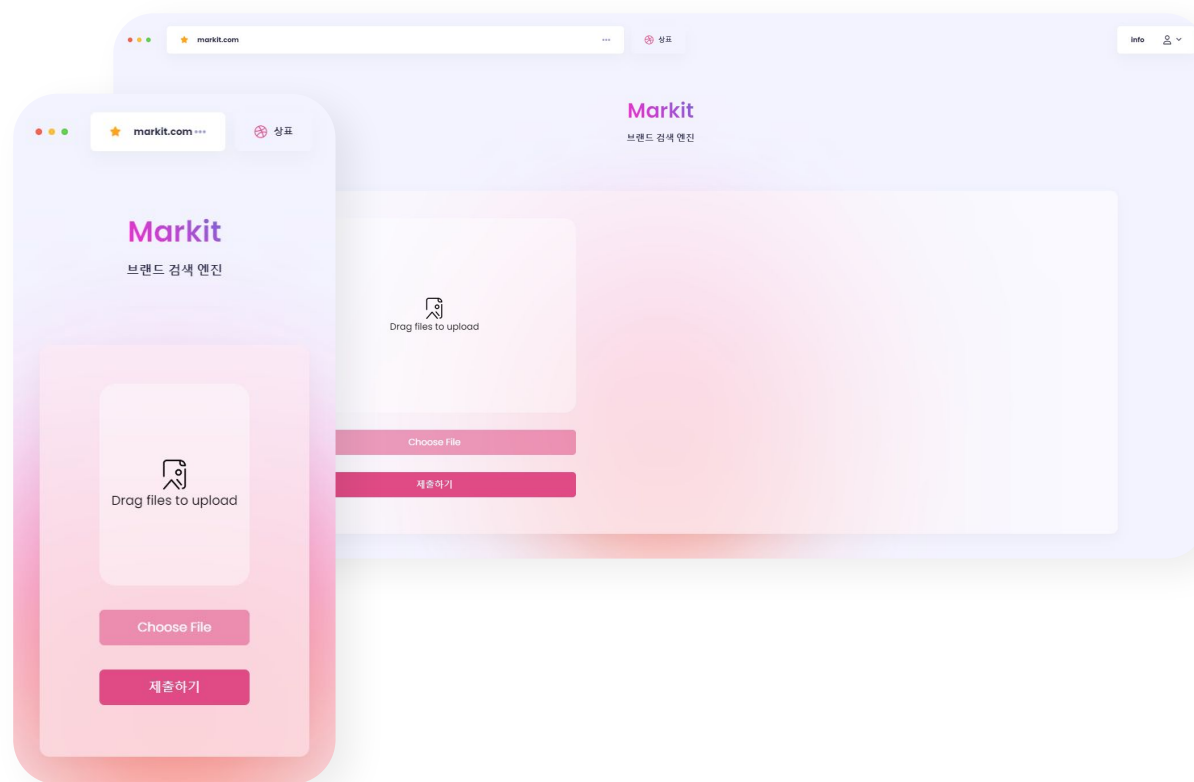
*과반수 유사성으로 인한 실패 출처 : https://piip.co.kr/ko/blog/Examiners_Statistics_Grounds_for_Trademark_Refusal

*패소 손실액 9800위안 : http://ccbrand.kr/customer/trademark_news.php?bo_table=trademark_news&wr_id=12

Markit 소개

브랜드 검색 엔진 : 마킷

상표권 로고를 넣으면 유사한 상표권을 찾아드립니다.



멤버별 목업 아이디어션

주요 화면 및 설명



메인 화면
로그? 설명 이미지 밑에 업로드 버튼 누르기

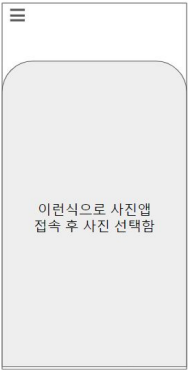


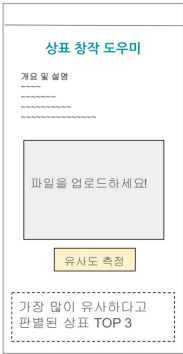
사진 선택
로컬의 사진업 접속해서 사진 선택



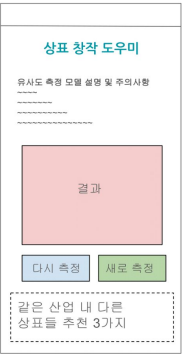
결과 화면
유사도 측정 및, 유사한 이미지 중 rank 5 확인 가능한 버튼

이예진

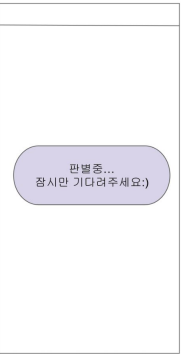
주요 화면 및 설명 - 지우



메인 화면
1. 로고 및 펍업 이름
2. 펍업 개요 및 설명
3. 파일 드래그 앤 드롭 박스
4. 유사도 측정 버튼
5. 가장 많이 유사하다고 판별된 상표 TOP 3



결과 화면
1. 로고 및 펍업 이름
2. 유사도 측정 모델 설명 및 주의사항
3. 측정 결과
4. 다시측정, 새로 측정 버튼
5. 같은 산업 내 다른 상표들 추천 3가지



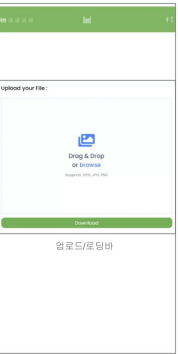
로딩 화면
말그대로 로딩화면! :)

김지우

주요 화면 및 설명



초기 화면
화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명



START버튼 이후 업로드화면
화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명 화면 설명



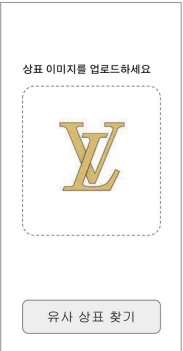
업로드 후 결과화면
각 브랜드

원하선

주요 화면 및 설명



업 첫 화면
업 이름과 업 설명 노출



이미지 업로드 화면
- 유사도 분석할 이미지 업로드
- 업에서는 사진형 열기
- 웹 브라우저에서는 드래그&드롭



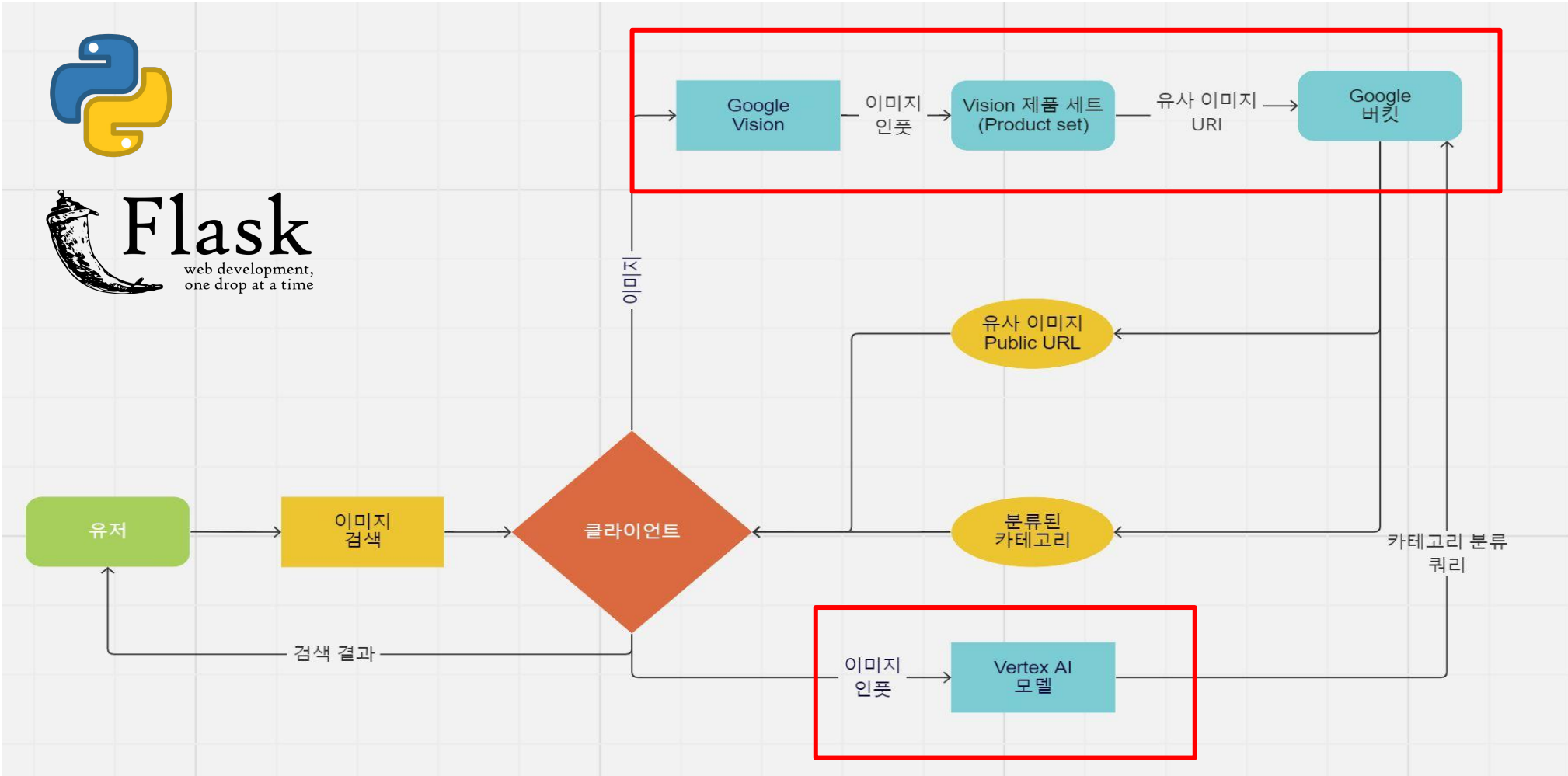
유사도 결과
- 상위 10개의 유사한 상표 목록 (상위 3개 강조)
- 유사한 상표의 유사도(%), 유사한 상표의 기업명, 국적

이재희

70% 이상 유사 이미지 신규 생성시 알림
유사한 이미지군 업로드한 사람에게, 이후 추가 유사한 이미지 생성시 알림 발송.

결과 사용자 만족도 측정
☆☆☆☆

Markit 앱 : 프로세스 - Data Flow



기술은 어떻게 개발됐나요?

1. 상표권 분류 : AutoML 활용
2. 유사 로고 검색 : VISION search 활용

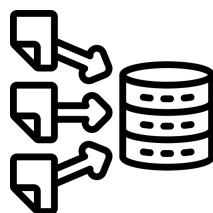
1. 상표권 분류 : AutoML 활용

상표권 분류 : AutoML - 데이터 웨어하우스 및 마트 구축

1. 데이터 수집



2. 데이터 클렌징



3. GCS 업로드



- 데이터 셋 구축

- 해외상표 이미지 AI 데이터 **22.77GB** 수



- 유사상표 이미지 검색 서비스의 사용자 입력 이미지 데이터 **12.59GB**



- 상표권 침해 관련 선행 프로덕트 및 판례, 연구 자료 수집

상표권 분류 : AutoML - 데이터 라벨링

1. 데이터 구축 규모

- 원천 데이터: 약 **84만 건** 규모의 jpg형태
- 라벨링 데이터: 약 **151만 건** 규모의 json형태

2. 데이터 분포

- 해외상표 디자인 코드 종류 분포
- 데이터 통계 다양성
 - 데이터 신뢰성 및 공정성을 위해 1800년대부터 2022년까지 미국에서 출원된 모든 상표 이미지 수집을 통해 **시대, 산업별 편향성 제거**

3. 비엔나 코드 중분류를 원천데이터의 label로 사용

중분류	desc	구성비
2601	Circles	9.70%
2617	Lines, bands, bars	6.20%
2609	Squares	5.70%
2611	Rectangles	4.60%
0115	Natural phenomena	4.00%
0301	Cats, dogs, wolves, foxes, bears, lions, tigers	3.70%
0315	Birds, bats	3.70%
0211	Parts of the human body, skeletons, skulls	2.90%
2615	Polygons (geometric figures with five or more sides)	2.80%
0501	Trees, bushes	2.70%
2801	Inscriptions	2.60%
2607	Diamonds	2.60%
0201	Men	2.60%
2401	Shields, crests	2.60%
0905	Headwear	2.40%
0101	Stars, comets	2.20%

상표권 분류 : AutoML - 데이터 전처리

```
import os
from google.cloud import storage

def generate_gcs_uri(bucket_name, folder_path):
    client = storage.Client()
    bucket = client.get_bucket(bucket_name)
    blobs = bucket.list_blobs(prefix=folder_path)

    uri_list = [f"gs://{bucket_name}/{blob.name}" for blob in blobs]

    return uri_list

# 사용 예
bucket_name = "apteam01_sample_02"
folder_path = "Trees,bushes/"
image_uris = generate_gcs_uri(bucket_name, folder_path)

# JSON에 적용
# json_data["imageGcsUri"] = image_uris

print(image_uris)
```

```
def make_labeled_json(image_uris, annotations_list, images_list):
    made_json = []

    # 이미지 파일 이름을 키로 하는 딕셔너리 생성
    images_dict = {image['fileName'][-17:]: image for image in images_list}

    folder_name = image_uris[0].split('/')[3]

    for image_uri in image_uris:
        image_filename = image_uri[-17:]

        # 이미지 파일 이름과 일치하는 annotation 찾기
        annotation = next((ann for ann in annotations_list if ann['fileName'] == image_filename), None)

        # 이미지 파일 이름과 일치하는 image 찾기
        image = images_dict.get(image_filename)

        # annotation과 image가 모두 존재하는 경우에만 처리
        if annotation and image:
            xMin = annotation['points'][0][0] / image['img_w']
            yMin = annotation['points'][0][1] / image['img_h']
            xMax = annotation['points'][1][0] / image['img_w']
            yMax = annotation['points'][1][1] / image['img_h']

            made_json.append({
                'imageGcsUri': f'{image_uri}',
                'boundingBoxAnnotations': [{
                    'displayName': f'{folder_name}',
                    'xMin': f'{xMin}',
                    'yMin': f'{yMin}',
                    'xMax': f'{xMax}',
                    'yMax': f'{yMax}'}]}

    return made_json
```



정규화 작업

1. RAWDATA를 비엔나코드에 맞춰 **label 별로 분류** 후 GCP 버킷에 업로드.
2. GCP 상에 있는 이미지들의 **URI들을 가져오는 자동화 코드** 작성.
3. Vertex AI가 원하는 형식에 맞춰 이미지의 **GCP 상의 URI와 분류된 label**, 그리고 **bounding box 크기의 정규화된 비율을 json형식**으로 만드는 자동화 코드를 작성

상표권 분류 : AutoML - 모델 결과



2. 유사 로고 검색 : VISION 활용

유사 로고 검색 - Vision 개요

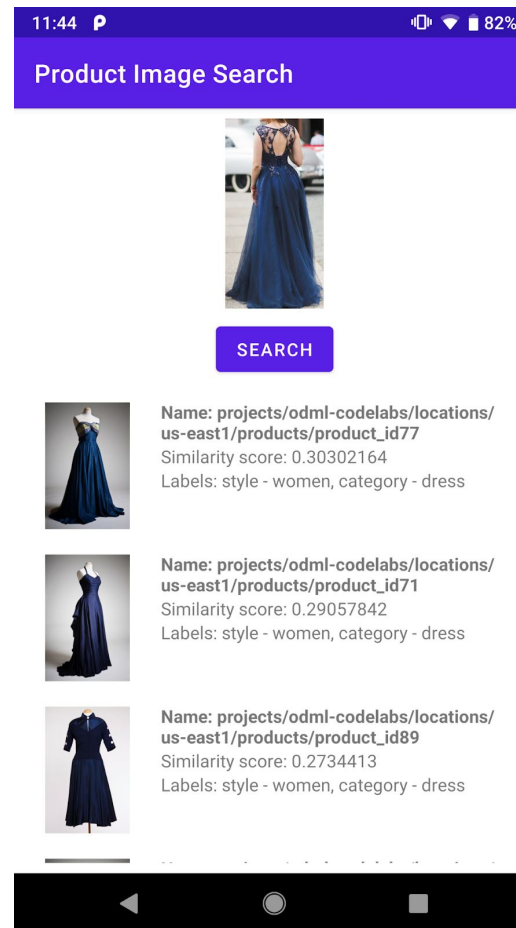


Cloud Vision

Image recognition and classification

제품검색 서비스

이미지 입력



유사 로고 검색 - 프로세스



1. Cloud 버킷 이미지 업로드

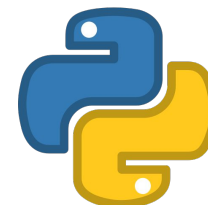


Product Set

2. 커스텀 Vision API
product set 생성



3. Flask 웹페이지로 이미지 업로드 후
검색, 결과반환



Flask
web development,
one drop at a time



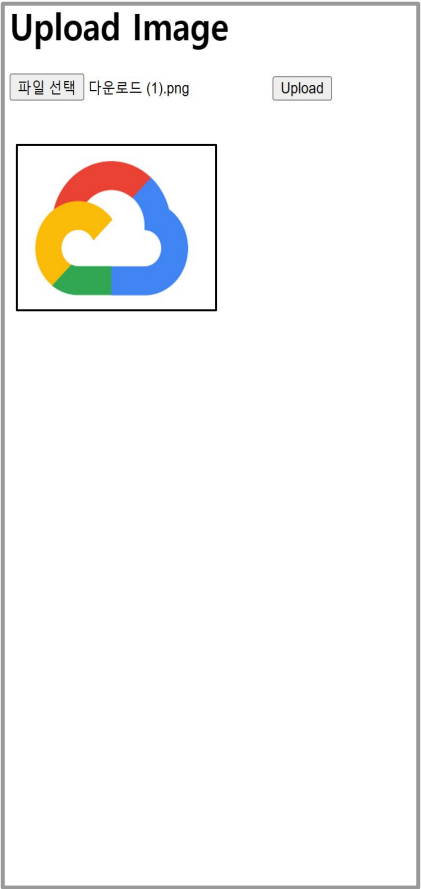
Cloud Vision

Image recognition and classification

Markit의 서비스 구조는 이렇습니다.

1. 웹앱 : 파이썬 플라스크
2. 배포 : GCP App Engine 활용

웹 앱 : 최소기능제품(MVP) 테스트



1. VISION SEARCH

Search Results

- 0.4646771252155304 - projects/appteam01/locations/us-west1/products/2022_97216042/referenceImages/17937464638078704825



2295

- 0.3982274830341339 - projects/appteam01/locations/us-west1/products/2022_97225326/referenceImages/179374646380787048253822



- 0.3957415223121643 - projects/appteam01/locations/us-west1/products/2022_97227747/referenceImages/179374646380787048254259



2. VERTEX ML

▼ **Circles (5)**

Circles 1	<div><div></div></div>	0.399
Circles 2	<div><div></div></div>	0.104
Circles 3	<div><div></div></div>	0.082
Circles 4	<div><div></div></div>	0.057
Circles 5	<div><div></div></div>	0.053

▶ **Ovals (1)**

▶ **Natural_phenomena (3)**

▶ **Rectangles (1)**

▶ **Lines,bands,bars (1)**

▶ **Squares (1)**

▶ **Polygons(geometric_figures_with_five_o**

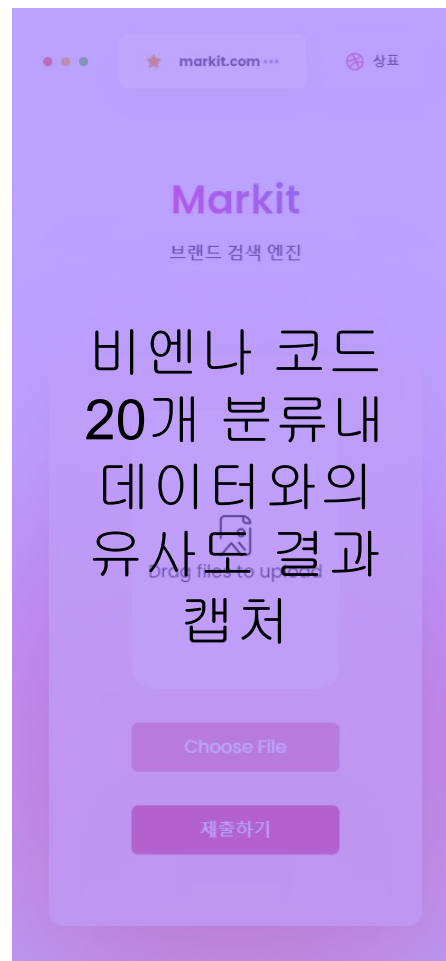
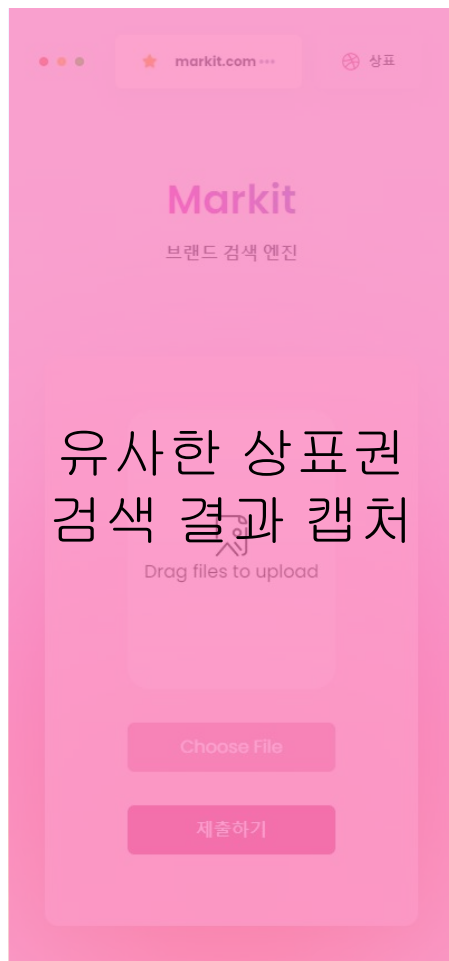
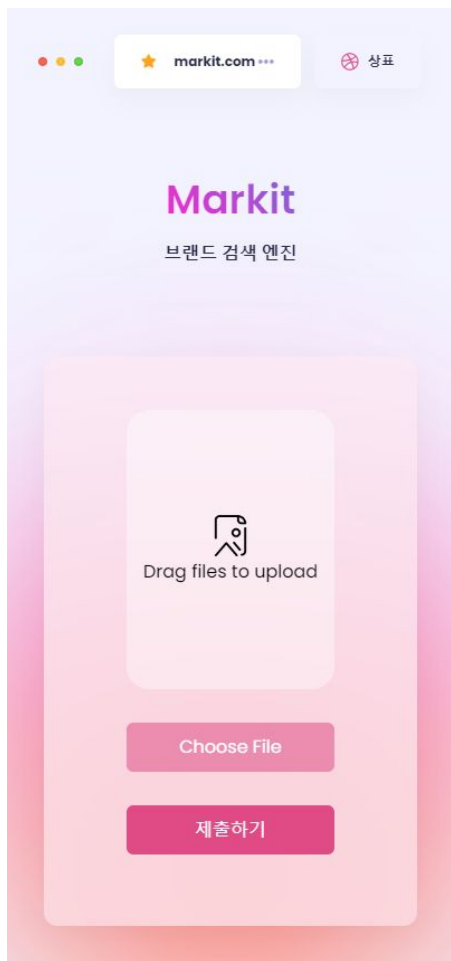
배 포 : GCP App Engine 활용



App Engine

Markit은 누구나 사용할 수 있습니다.

MarkIt을 직접 사용해 보세요.



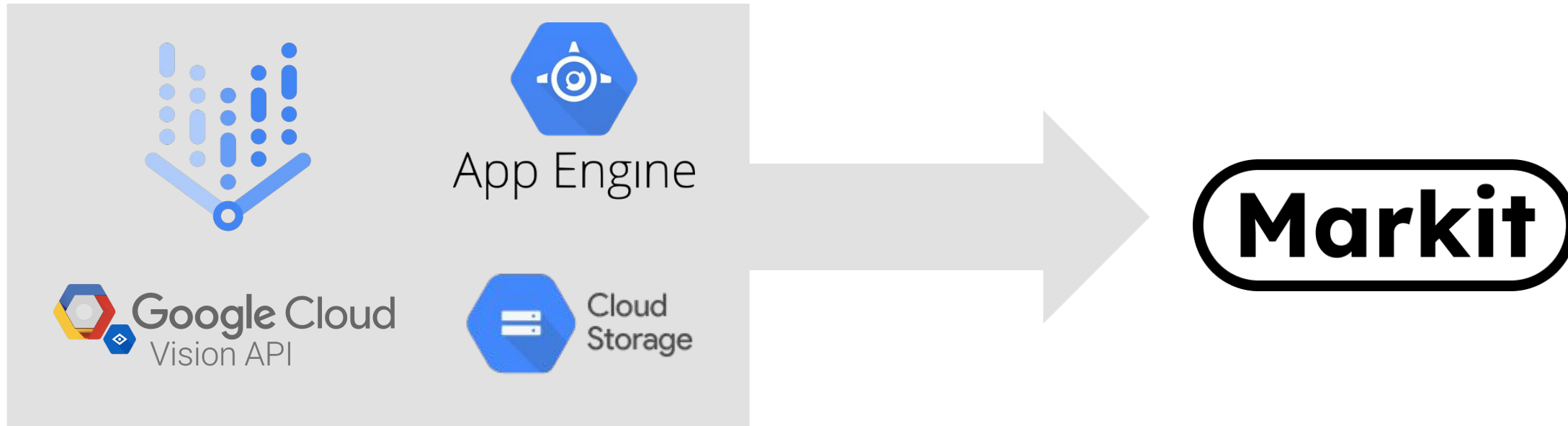
<https://20231129t084322-dot-appteam01.uc.r.appspot.com/>

Markit

GCP 및 서버리스 개발 후기

GCP 및 서버리스 개발 후기

- 1) 필요한 **GCP**서비스 융합을 통한 플랫폼 제공이 용이
- 2) 구글의 강력한 종단간 암호화 기반으로 안전한 어플리케이션 제공 가능
- 3) 앱 엔진을 통한 각 기능별 테스트, 배포 용이
- 4) **End-to-end** 송수신으로 프로비저닝 필요 X, 운영 효율 증대



서비스 확장 가능성

- **Generative AI**
 - 현존 상표권을 모두 참조하여 유사성을 회피한 상표권을 생성형 AI 추천하여 제공.
- **상표권 출원 자동화**
 - 각 국의 상표권 등록 과정 시스템과 연동하여 상표권 출원 과정을 대행하는 서비스 제공.
- **변리사와 전자 중개 서비스**
 - 변리사의 상표권 출원 서비스와 연동.
- **crunchbase와 같은 기업 리서치 엔진, 데이터 마트**
 - 수집 및 분석된 기업 정보를 바탕으로 전 세계 브랜드 리서치 엔진을 포괄적으로 제공.
- **브랜드 유관도 분석 및 추천**
 - 단순 산업권이나 이미지 분석 뿐만 아니라 실제 영업 연관도를 바탕으로 브랜드 간 협업 혹은 영업 기회 제공.

감사합니다.

Appendix

MVP Closed beta tester 후기

- 000 / 직업 / 소속
TBD

- 000 / 직업 / 소속
TBD

- 000 / 직업 / 소속
TBD

- 000 / 직업 / 소속
TBD

프로젝트 후기

- 원하선
TBD

- 김지우
Great.

- 이예진
TBD

- 이재희
독립성을 갖추고 다양한 관심사를 가진 분들과 같은 프로그램에서 팀을 이루어 프로젝트를 진행하게 되어 행복하고 즐거웠습니다. 특히 어?금지 팀을 만난 것을 행운이라고 생각합니다.