

[SwiftSight] AI Research Engineer 과제

I. 과제 주제

모델 레지스트리 시스템 구현

II. 과제 내용

SwiftSight의 AI Research Engineer로서 ML 모델을 관리하기 위한 간단한 모델 레지스트리 시스템을 구축해야 합니다. 작동하는 코드와 문서가 포함된 ZIP 파일을 제출해주세요.

제출물에는 다음이 포함되어야 합니다:

1. 작동하는 코드

- 모델을 등록, 조회, 나열할 수 있는 Python 스크립트
- 2개의 사전 학습된 오픈소스 모델 (아래 제안 참조)
- 로컬 파일 시스템 저장소
- 기능을 시연하는 예제 스크립트 1개

2. README.md 포함 내용:

- **설치 및 실행 방법:** 설치 및 실행 방법
- **사용 예제:** 레지스트리 사용 방법
- **AI 도구 사용 노트:** 어떤 도구를 어떻게 사용했는지 (아래 가이드라인 참조)

3. SYSTEM_DESIGN.md 포함 내용:

- **배경 조사:** 모델 레지스트리가 해결하는 문제는 무엇인가? 기존 솔루션(MLflow, HuggingFace Hub 등)에서 고려한 디자인 패턴은?
- **아키텍처 다이어그램:** 시스템 구성 요소와 상호작용을 시각적으로 표현
- **핵심 구성 요소:** 각 주요 구성 요소와 그 책임 설명
- **디자인 결정:** 다음을 포함한 접근 방식 설명:
 - 저장 전략 및 버전 관리 체계
 - 기술 선택 (YAML vs SQLite vs 기타)
 - 고려한 트레이드오프
 - 특정 선택을 한 이유
- **데이터 흐름:** 등록 및 조회가 단계별로 어떻게 작동하는지

III. AI 도구 활용 가이드라인

이 과제의 모든 부분에서 AI 도구(ChatGPT, Claude, Copilot 등)를 자유롭게 사용하시기를 권장합니다.

README에 간단히 문서화할 내용: - 어떤 AI 도구를 사용했는지 - 특히 도움이 되었던 프롬프트 2-3개 예시 - AI 제안에 동의하지 않았거나 수정해야 했던 사례 1개

IV. 기술 요구사항

제안 구조 (예시 - 수정 가능):

```
model_registry_assignment/
├── README.md           # 설치, 사용법, AI 활용
├── SYSTEM_DESIGN.md    # 배경, 아키텍처, 디자인 결정
├── requirements.txt     # 의존성
├── registry.py         # 레지스트리 구현
├── example.py          # 데모 스크립트
├── models/             # 2개의 모델 파일
└── registry.yaml       # 선택사항: 설정 파일
```

제안 비전 모델 (2개 선택, 총 <200MB):

일반 컴퓨터 비전: - ResNet-18 (PyTorch) - ~45MB - 이미지 분류 - MobileNetV2 (PyTorch) - ~14MB - 경량 분류 - EfficientNet-B0 (PyTorch) - ~20MB - 효율적인 분류 - SqueezeNet (PyTorch) - ~5MB - 초경량

의료 영상: - MedMNIST 모델 (TensorFlow/PyTorch) - ~5-10MB - 의료 이미지 분류 - U-Net (작은 변형) - ~30-50MB - 의료 이미지 세그멘테이션 - HuggingFace 또는 PyTorch Hub의 작은 의료 영상 모델

참고: 의료 도메인 모델을 권장하지만 필수는 아닙니다. 작은 비전 모델의 조합은 무엇이든 사용 가능합니다.

필수 핵심 기능:

1. **모델 등록** - 메타데이터(이름, 버전, 프레임워크, 메트릭)와 함께 모델 등록, 모델 패밀리 지원
2. **모델 조회** - 이름/버전으로 가져오기, "latest" 쿼리 지원, 저장소 fallback을 통한 경로 조정
3. **메타데이터 관리** - 메타데이터(아키텍처, 입력 shape, 메트릭) 저장, 리니지 추적 (데이터셋, 프레임워크, 날짜)
4. **저장소 백엔드** - 로컬 파일 시스템 필수
5. **버전 관리** - 시맨틱 버저닝 지원, 모델의 모든 버전 나열 (예: ResNet v1, v2, v3)

V. FAQ

Q. GPU나 클라우드 저장소를 사용해야 하나요? A. 아니요. CPU와 로컬 파일 시스템 사용으로 제한합니다.

Q. 어떤 프로그래밍 언어를 사용해야 하나요? A. Python 3.10+

Q. 코드가 즉시 실행되어야 하나요? A. 네. 다음과 같이 실행할 수 있어야 합니다:

```
pip install -r requirements.txt
python example.py
```

Q. 이 과제에 얼마나 시간이 걸려야 하나요? A. 4-6시간 정도의 작업을 목표로 하세요. 과도한 엔지니어링 대신 코어 기능들과 killing feature를 중점으로 고민해주세요.

VI. 평가 기준

- **실행 가능 여부** - 코드를 실행하고 레지스트리를 사용할 수 있나요?
- **코드 품질** - 깔끔하고 가독성이 좋으며 잘 정리되어 있나요?
- **시스템 디자인** - SYSTEM_DESIGN.md의 아키텍처 다이어그램, 디자인 결정, 트레이드오프 분석
- **AI 도구 활용** - AI 도구를 효과적으로 활용하고 문서화했나요?

VII. 제출

- **기한:** 1주일 (연장 가능 - 미리 알려주세요)
- **형식:** [이름]_ModelRegistry.zip 이름의 ZIP 파일
- **문의:** 질문이 있으시면 jung.woojin@airsmed.com으로 연락주세요