요구사항 명세서

IT 채용공고 문서 수집 인프라 구축 및 트렌드 분석

프로젝트 개요와 기술 스택을 활용한 시스템 구조는 다음과 같습니다.

1. 기능 요구사항

1.1 데이터 수집

- 크롤링 대상: 사람인, 잡코리아, 원티드, 인크루트, Linkedin, Indeed 등 국내외 주요 채용 사이트
- 크롤링 주기: 일 단위 (정기적 데이터 수집을 위한 스케줄러 도입)
- 크롤링 방식: 파이썬 스크립트를 사용하여 웹 페이지의 HTML을 파싱하여 데이터 추출, 데이터 수집 실패 시 재시도 로직 포함

1.2 데이터 저장

- 데이터 레이크: AWS S3
    - 저장 데이터: 크롤링한 원시 데이터
- 데이터 웨어하우스/마트: On-Premise MySQL
    - 저장 데이터: 정제 후 데이터 저장, 최종 데이터 저장

1.3 데이터 정제

- 버퍼 시스템: Apache Kafka를 사용하여 데이터 스트리밍 처리
- 데이터 정제 도구: Apache Spark
- 정제 작업: 기사 본문 정제 (HTML 태그 제거, 불필요한 공백 제거 등)

1.4 데이터 분석

- 분석 도구: LLM (Large Language Model)
- 분석 작업: 텍스트 토큰화 및 빈도수 분석

1.5 보고서 생성 및 전송

- 보고서 생성 도구: Apache Airflow
- 보고서 전송 방식: 이메일 (Gmail)

2. 비기능 요구사항

2.1 시스템 배포

- 플랫폼: Kubernetes
- 서비스 모니터링: Prometheus+Grafana

2.2 버전 관리

- 버전 관리 시스템: Git
- 호스팅 플랫폼: GitHub

2.3 DevOps

- **CI/CD** 도구: Jenkins
- 프라이빗 레포지토리: Harbor

3. 기술 스택

- 크롤링: Python
- 배치 스케줄링: Apache Airflow
- 스트리밍 버퍼: Apache Kafka
- 데이터 정제: Apache Spark
- 데이터 저장: AWS S3
- **LLM** 모델: TBD (To Be Decided, 적절한 LLM 모델 선정 필요)
- 보고서 전송: Apache Airflow
- 배포: Kubernetes
- 모니터링: Prometheus + Grafana
- 버전 관리: GitHub
- **CI/CD**: Jenkins
- 프라이빗 레포지토리: Harbor
- **Web Service**: Django, Nginx, Gunicorn

4. 시스템 아키텍처

4.1 데이터 수집

- 크롤링: Python 스크립트가 주요 채용 사이트를 크롤링하여 데이터를 수집
- 스트리밍: 수집된 데이터는 Apache Kafka를 통해 스트리밍됨

4.2 데이터 저장

- 데이터 레이크: Kafka에서 스트리밍된 데이터는 AWS S3에 저장됨
- 데이터 웨어하우스/마트: 정제된 데이터는 On-Premise의 MySQL에 저장됨

4.3 데이터 정제

- 데이터 정제: Apache Spark가 S3에 저장된 데이터를 정제하여 처리.

4.4 데이터 분석

- 데이터 분석: 정제된 데이터는 LLM 모델을 통해 분석됨

4.5 보고서 생성 및 전송

- 보고서 생성: Apache Airflow가 분석된 데이터를 기반으로 보고서를 생성
- 보고서 전송: 이메일을 통해 보고서를 전송

4.6 시스템 배포 및 모니터링

- 배포: 모든 서비스는 Kubernetes 클러스터에 배포됨
- 모니터링: 서비스 상태 및 로그는 Prometheus + Grafana를 통해 모니터링

4.7 CI/CD 및 버전 관리

- **CI/CD**: Jenkins를 통한 자동화된 CI/CD 파이프라인
- 버전 관리: GitHub를 통한 코드 버전 관리
- 이미지 관리: Harbor를 통한 Docker 이미지 관리

이러한 시스템 구조를 통해 IT 채용공고 문서를 효율적으로 수집, 저장, 정제, 분석하여 현재 기업이 필요로 하는 기술 스택을 파악하고, 이를 기반으로 트렌드 보고서를 생성 및 배포할 수 있습니다.

    ○