StreamBoard

💡 - 스트리밍과 웹 대시보드가 강력한 데이터 플랫폼 'StreamBoard'

주제

📋 **주제: 데이터 플랫폼 벤치마킹을 통한 차세대 데이터 분석 플랫폼 프로토타입 개발**

데이터 엔지니어 직무에서의 전문성을 향상하기 위해, Databricks와 Snowflake와 같은 최신 데이터 플랫폼을 벤치마킹하여 차세대 데이터 분석 플랫폼의 프로토타입을 개발하는 프로젝트를 기획했습니다. 이 프로젝트는 각 플랫폼의 주요 기능과 성능을 분석하고, 이를 바탕으로 데이터 수집, 저장, 처리, 분석 기능을 포함한 최적의 데이터 플랫폼 아키텍처를 설계하는 것을 목표로 합니다. 이를 통해 데이터 엔지니어링 역량을 강화하고, 실무 경험을 쌓아 각 팀원들의 데이터 엔지니어링 직무 역량을 끌어올리고자 합니다.

타겟 유저(페르소나)

- 페르소나 주제 : 데이터 통합 및 분석 플랫폼
- 목표: 실시간 및 배치 데이터를 통합하여 메타 데이터화하고, 사용자(기업CTO, 데이터 엔지니어, 데이터 분석가 등)가 쉽게 데이터 파이프라인을 커스터 마이징하여 구축할 수 있는 환경 제공.

Use Case

- 홈 IoT 제공하는 회사의 데이터 분석가들이 쓰기 좋은 데이터 플랫폼
  - 직책 : 데이터 분석가
  - 배경 : 홈 IoT 서비스를 개발하는 회사에서 데이터 분석가들의 업무 효율성과 타 부서와의 협업을 원활하게 하기 위한 서비스를 찾고 있음.
  - 목표 :

        - 공동생활시설에서 다양하게 발생하는 실시간 IoT 데이터를 효율적으로 수집하고 통합 가능한 기능 제공.
        - 다양한 데이터소스에서 데이터를 수집하고 통합.
        - 향후 서비스 개발 시 데이터 기반 의사결정을 통해 생산성 향상 및 비용 절감.
        - 사용하기 쉬운 대시보드를 통해 실시간 모니터링 가능.
        - 사전 정의된 그룹에 대한 뷰 테이블 정의 및 저장.
            - EX) 평형별, 가족 구성인원별, 건물 타입별
        - 스마트 기기가 장치 고장 및 문제가 발생면 사용자나 관리자에게 알림
        - Iot 스마트 기기 데이터를 실시간으로 수집하여 운영 부서에 배포해주는 파이프라인 구축
        - 분석 목적에 맞는 데이터 마트 구축 및 작성된 스키마에 따른 API 제공 → 분석 툴 연동
            - EX) 태블로, Power BI, BIgquery, Hadoop Ecosystem

고려할 점

- 보안
- 무중단 서비스
- 일정
- 구현 가능성
- 기술적 난이도

기본 기능(서비스)

- 데이터 인풋(스트리밍, 배치 ← MQ를 통한 통합 INPUT Streaming, Airflow)
    - 스트리밍 데이터 입력
    - 배치 데이터 입력
    - 통합 데이터 입력 지점(Kafka)
- 데이터 저장소(레이크, 웨어하우스, 비관계형 데이터베이스, 데이터 마트)
    - 관계형 데이터베이스
        - MySQL, RDS

    - 비관계형 데이터베이스
        - MongoDB
    - 백업 데이터베이스
- 데이터 거버넌스(통합 마스터 데이터셋 구성)
    - 유저별 마스터 데이터셋 정의 기능
- ETL, ELT
    - 데이터 인풋시 변환
    - 데이터 마트 이동시 변환
    - 데이터 호출시 변환
- 유저 관리자 대시보드(데이터 관리 가능한 UI)
    - 리액트 사용
    - 웹 UI를 통한 그래프 표출
    - 외부 서비스 API 통합
- 코어 서버(Quart)
    - 대시보드 관리
    - 데이터 플로우 관리
    - RestAPI 관리
    - 외부 서비스 API 통합
    - 데이터 표출
- 보안, 인증
    - 각 유저별 조회 가능 데이터 제한(서버단)

추가 기능

- ML 지원 서비스(AutoML, 단순 회귀/분류 추론)
- 웹페이지 디자인
- 데이터 조회 페이지 공유

기술 스택

프론트엔드

- React, Next.js(Javascript):
  - 고객용 대시보드를 만들고 고객이 데이터 플랫폼의 기능을 웹UI상에서 원활하게 사용하도록 하고자 함

백엔드

- Quart(Python)
  - 고객들이 플랫폼의 기능을 사용하는데 있어 필요한 기능들과 API 들을 구현하기 위해 사용하고자 함

인프라

- DB
  - MySql
    - 고객 관리나 기타 기능별 저장소와 같은 부분의 저장을 위해 사용하고자 함
  - MongoDB
    - 고객이 넣는 데이터를 수집하는 서버로써 BSON타입으로 다양한 형태의 데이터를 적재할 수 있다는 장점 때문에 Cassandra 대신 선택함
- AWS
  - 서비스 배포를 위해 클라우드 인프라를 사용하고자 함.
  - AWS의 EKS 서비스를 이용해 K8s기반으로 구축한 서비스들을 배포할 예정.
- K8s
  - Container 기반으로 서비스를 배포하고 사용량에 따른 오토 스케일링을 진행하기 위해 사용하고자 함
- NiFi
  - Data-Flow의 제어와 처리를 위한 Rule을 보다 쉽게 만들고자 도입함
- Kafka
  - 전체 서비스를 관통하는 MQ로써 메시지가 원활하게 전달될 수 있도록 하기 위해 도입함
- Kafka Streams
  - Kafka Topic에 쌓인 데이터를 간단한 데이터 변환으로 전처리하고자 도입함
- Airflow

- 데이터 수집을 진행할 때 수집 Job을 관리하는 툴로 사용하기 위해 도입함