

## Projekt (Z1\_JG, 2024L)

Celem projektu jest opracowanie oprogramowania służącego do wykrywania anomalii w transakcjach dokonywanych kartami płatniczymi.

1. Projekt oprogramowania powinien zawierać opis poszczególnych komponentów oraz założenia do symulatora anomalii.
2. Oprogramowanie powinno składać się z następujących komponentów:
  - a. Standardowego oprogramowania Kafka i Flink.
  - b. Symulatora transakcji realizowanych za pomocą kart płatniczych, który jednocześnie będzie producentem wiadomości dla Kafki.
  - c. Testowego konsumenta Kafki, który pozwoli na analizę poprawności generowanych danych z wizualizacją danych.
  - d. Detektora anomalii (Aplikacji Flink), który będzie czytał dane z Kafki, wykrywał anomalie i wysyłał do wydzielonego topiku Kafki informacje alarmach.
  - e. Programu do odczytu alarmów, wizualizacji i informowania o alarmach.
3. Założenie do symulatora kart płatniczych:
  - a. Transakcje będą generowane dla 10000 różnych kart.
  - b. Dane generowane powinny zawierać:
    - ID karty.
    - ID użytkownika (jeden użytkownik może mieć więcej niż jedną kartę).
    - Lokalizacja transakcji w formie współrzędnych GPS.
    - Wartość transakcji.
    - Dostępny limit wydatków na karcie.
  - c. Dane powinny być generowane w formacie JSON.
  - d. Centralną częścią symulatora jest generator danych pozwalający na generowanie różnego typu anomalii np. nagła zmiana wartości transakcji, nagła zmiana lokalizacji, częstość transakcji itd. Osoba realizująca projekt ma swobodę doboru typu anomalii.
4. Założenie do detektora anomalii.
  - a. Dane powinny być czytane z Kafki w trybie prawie rzeczywistym
  - b. Metoda/metody detekcji anomalii powinny opierać się na w miarę możliwości algorytmach bazujących na statystykach
  - c. Należy przewidzieć pamięć tymczasową np. do przechowywania częstych lokalizacji
5. Realizujący projekt decyduje o wyborze języka programowania.

Termin oddania projektu: 7 czerwca 2024

### **Forma oddania projektu:**

1. Projekt rozwiązania.
2. Dedykowane oprogramowanie.
3. Wyniki testów oprogramowania.
4. Demonstracja rozwiązania prowadzącemu.