

Документација за Микросервиси - Изработка на Финансиски Систем

Оваа документација го опишува начинот на кој се изработени петте микросервиси кои заедно формираат интегриран систем за анализа на финансиски податоци во реално време. Во оваа документација ќе биде опишан процесот на развој на секој од микросервиси, користените технологии и нивната комуникација.

1. Data Ingestion Service (Сервис за собирање на податоци)

Сервисот за собирање на податоци е одговорен за примање на податоци од различни надворешни извори преку HTTP или WebSocket протоколи. Овој сервис користи API-и како **Yahoo Finance** или **Alpha Vantage** за да добие податоци за акции, валути и други финансиски инструменти.

Развивање на сервисот:

- **API комуникација:** Користевме Java и Spring Boot за креирање на RESTful API кој ги собира податоците од надворешните извори. Комуникацијата со API-ите е реализирана преку HTTP барања.
- **Обработка на податоци:** Потоа, добиените податоци се обработуваат и ги испраќаат во системите за реално-временска обработка (Kafka).

2. Real-Time Processing Service (Сервис за обработка во реално време)

Сервисот за обработка во реално време е одговорен за обработка на податоците кои пристигнуваат преку Kafka во реално време. Користиме **Apache Spark Streaming** за процесирање на податоците.

Развивање на сервисот:

□ **Apache Kafka:** За испраќање на податоците во реално време, користевме Kafka, што овозможи брзо и безбедно преносување на податоци од сервисот за собирање податоци до сервисот за обработка.

□ **Apache Spark Streaming:** За обработка на податоците во реално време, имплементиравме Spark Streaming. Овој систем овозможи агрегирање и филтрирање на податоците во текот на обработката.

3. Data Storage Service (Сервис за складирање на податоци)

Сервисот за складирање на податоци е одговорен за долговечно складирање на податоците во различни бази на податоци. За неструктурирани податоци се користи **Cassandra**, а за структурирани податоци **PostgreSQL**.

Развивање на сервисот:

□ **Интерфејс за бази на податоци:** За интеракција со базите на податоци, користевме Spring Data JPA за PostgreSQL и Spring Data Cassandra за Cassandra.

□ **Процес на складирање:** Податоците што пристигнуваат во сервисот за складирање се конвертираат и складираат во одговарачките бази на податоци, во зависност од нивниот тип (структурирани или неструктурирани).

4. Data Analysis Service (Сервис за анализа на податоци)

Сервисот за анализа на податоци е одговорен за анализа на историските и реално-временските податоци. Користи **машинско учење** и **статистичка анализа** за генерирање на предвидувања и трендови.

Развивање на сервисот:

□ **Машинско учење:** Изградивме и интегриравме модели за машинско учење кои предвидуваат трендови на пазарот.

□ **Статистичка анализа:** Користевме библиотеки како **TensorFlow** и **Scikit-learn** за креирање на модели за анализа на податоците.

- **API за резултати:** Направивме API кој ја враќа анализата и резултатите во JSON формат.

5. Data Visualization Service (Сервис за визуелизација на податоци)

Сервисот за визуелизација на податоци овозможува прикажување на анализираните податоци преку интерактивни графикони и табели користејќи **D3.js** или **Plotly**.

Развивање на сервисот:

- **Динамички графикони:** Користиме **D3.js** и **Plotly** за динамички графикони кои ги прикажуваат резултатите од анализата во интерактивна форма.
- **Интерфејс за корисници:** Веб-интерфејс за визуелизација на податоците е развиен со помош на **JavaScript** и **HTML5**.

Секој од сервисите е развиен со цел да работи независно, но истовремено да комуницира со другите сервисите преку **RESTful API**. За разменување на податоци во реално време се користи **Apache Kafka**, додека за складирање на податоците се користат **PostgreSQL** и **Cassandra**.