**Архитектура на апликацијата**

Овој документ ја претставува концептуалната, извршната и имплементационата архитектура на апликацијата за обработка на историски податоци од Македонската берза. Апликацијата е дизајнирана како хибридна архитектура која ги комбинира следните архитектурни стилови: **цевки и филтри**, **слоевита веб архитектура**, **дистрибуирана архитектура со микросервиси** и **контејнеризација**.

**1. Концептуална архитектура**

Концептуалната архитектура ги дефинира основните функции и протокот на податоци во системот:

**Главни компоненти:**

- **Модул за прибирање податоци:** Автоматско преземање на историски податоци од веб-страницата на Македонската берза.

- **Модул за обработка на податоци:** Примена на цевки и филтри за трансформирање и чистење на податоците.

- **Модул за анализа:** Анализа на обработените податоци, подготвени за визуелизација и извештаи.

- **Модул за складирање:** Чување на трансформираните податоци во структурирана база на податоци.

- **Веб интерфејс:** Обезбедување кориснички пристап до анализите и визуелизациите.

**Проток на податоци:**

1. Податоците се преземаат од изворите.

2. Се обработуваат преку цевки и филтри за чистење и трансформирање.

3. Се складираат во база на податоци.

4. Се визуелизираат преку веб интерфејс.

2. **Извршна архитектура**

Извршната архитектура ги опишува оперативните аспекти и физичката распределба на компонентите:

**Дистрибуирани компоненти:**

**Микросервиси:**

- *Сервис за прибирање податоци:* Локално хостиран сервис кој пристапува до веб-страницата.

- *Сервис за обработка*: Одговорен за трансформации со употреба на цевки и филтри.

- *Сервис за анализа:* Врши статистички анализи и генерира извештаи.

- *Сервис за складирање:* Запишува трансформирани податоци во CSV формат.

- *Сервис за кориснички интерфејс:* Локално хостира веб апликацијата.

**Технологии:**

*- Контејнеризација*: Секој сервис е изолиран во Docker контејнери за лесна управливост.

*- Оркестрација*: Docker Compose се користи за локален развој и управување со сервисите.

**3. Имплементациона архитектура**

Имплементационата архитектура ги опишува специфичните технологии и модули кои ќе бидат користени:

**Технологии и јазици:**

- **Backend**:

- Python со BeautifulSoup

- Pandas за обработка на податоци.

- Java Spring Boot

- **Frontend**:

- HTML, CSS, JavaScript and Thymeleaf.

- Chart.js за визуелизации.

- **База на податоци:** CSV files

**Цевки и филтри:**

- Реализација на секвенцијални процеси за трансформација на податоците.

- Логички чекори:

1. Валидирање на преземените податоци.

2. Чистење на податоците (отстранување на дупликати, пополнување на празни вредности).

3. Трансформација во унифициран формат.

**Контејнеризација:**

- Docker Compose за локален развој.

- Kubernetes YAML датотеки за оркестрација во продукција.

Со оваа хибридна архитектура, апликацијата ќе биде скалабилна, модуларна и лесно управлива. Ова овозможува ефикасно управување со големи количини податоци и флексибилност за понатамошно проширување.