**Опис на архитектурните стилови и шаблони**

1. *Концептуална архитектура*

A diagram of a data processing process

Description automatically generated

Овој дијаграм ја претставува концептуалната архитектура на системот.

**Компоненти:**

* **User (Корисник):** Корисникот го користи системот за да пребарува акции и добие предвидувања.
* **AppUI (User Interface):** Графичкиот интерфејс кој му овозможува на корисникот да ги пребарува акциите и да добие предвидувања за нив. Преку оваа компонента, корисникот испраќа барања за акции.
* **Presentation Layer:** Оваа компонента го обработува корисничкиот внес и ги проследува барањата до следната логика во системот. Потоа, врз основа на резултатите, ја враќа информацијата назад до **AppUI**.
* **Data Access Layer:** Овој слој е одговорен за управување со податоците. Првично ќе се обиде да ги најде потребните податоци во локалните CSV фајлови. Ако податоците не се пронајдени, ќе иницира процес за scrapping.
* **Analytics Service:** Оваа компонента ги обработува и анализира податоците за да генерира предвидувања на цените на акциите користејќи алгоритми за машинско учење (на пример, LSTM).
* **Data Collection Service:** Оваа компонента собира податоци од надворешни извори, како што е веб страната Македонската берза. Собраните податоци се зачувуваат во локалните CSV фајлови.
* **CSV Storage:** Локално складиште каде што се чуваат сите податоци во CSV формат.

1. *Извршна архитектура*

*A diagram of a software company

Description automatically generated*

Овој дијаграм ја претставува извршната архитектура на системот.

**Компоненти:**

* **GUI (Graphical User Interface):** Ова е визуелниот интерфејс преку кој корисникот комуницира со системот. **GUI** користи **HTTP барања** за комуникација со backend сервисите.
* **Data Service:** Backend сервис кој ги управува податоците. Главни одговорности:
  1. Проверка на локалните податоци во **CSV Storage**.
  2. Комуникација со **Scraping Service** ако податоците недостасуваат.
  3. Интеракција со **Prediction Service** за генерирање на предвидувања.
* **LSTM Module (Prediction Service):** Ова е модул за предвидување, кој користи алгоритми за машинско учење,за обработка на податоците и генерирање на предвидувања за цените на акциите.
* **CSV Storage:** Локален складиштен простор во CSV формат каде што се чуваат податоците. **Data Service** го користи за проверка и ажурирање на податоците.
* **Scraping Service:** Оваа компонента собира податоци од надворешни извори односно Македонската берза. Ако не се пронајдат податоци во локалниот складиштен простор, **Scraping Service** ги собира потребните податоци и ги враќа назад.
* **Macedonian Stock Exchange (MSE):** Надворешен извор на податоци (веб-страница на Македонската берза) од каде што **Scraping Service** собира податоци.

1. *Имплементациска архитектура*

*A diagram of a computer

Description automatically generated*

Овој дијаграм ја претставува имплементациска архитектура на системот.

**Компоненти:**

* **Frontend (GUI):** Користи HTML, CSS и JavaScript за изградба на корисничкиот интерфејс. Преку HTTP барања, **GUI** комуницира со backend компонентите.
* **Backend (Data Service, Prediction Service, Scraping Service):** Овие сервиси се имплементирани во Python:
  + **Data Service:** Управува со логиката за читање и пишување податоци од/во **CSV Storage** и ги повикува **Scraping Service** и **Prediction Service**.
  + **Prediction Service:** Користи алгоритми за машинско учење LSTM во Python за обработка на податоците и генерирање на предвидувања.
  + **Scraping Service:** Ја користи Python библиотеката BeautifulSoup за да собира податоци од веб-страницата на Македонската берза и ги зачувува во **CSV Storage**.
* **CSV Storage:** Локално складиште во **CSV** формат каде што се чуваат сите податоци.
* **Macedonian Stock Exchange (MSE):** Веб-страница на Македонската берза од каде што **Scraping Service** собира податоци.