

OPIS ZADANIA:

W ramach tego zadania należy stworzyć aplikację symulującą giełdę akcji, gdzie zarejestrowani użytkownicy mogą kupować lub sprzedawać wirtualne akcje po aktualnych cenach.

WYMAGANIA BIZNESOWE:

- ▶ Użytkownicy:
 - Aplikacja umożliwia zarejestrowanie się nowego użytkownika oraz zdefiniowanie przez niego swojego portfela (zasoby finansowe w PLN oraz ewentualne pakiety akcji). Nowy użytkownik nie musi posiadać akcji wszystkich spółek będących na tej giełdzie.
 - Do aplikacji jednocześnie może być zalogowanych wielu użytkowników.
 - Każdy użytkownik może na bieżąco sprzedawać akcje ze swojego portfela oraz kupować akcje z giełdy. Nie ma możliwości bezpośredniego handlu pomiędzy różnymi zalogowanymi w tym samym czasie użytkownikami.
 - Przed zakupem/sprzedażą użytkownik powinien potwierdzić daną czynność.
- ▶ Giełda:
 - Giełda nie może sprzedać więcej jednostek danych akcji spółek niż tyle, ile obecnie posiada. Akcje mogą być sprzedawane tylko jako wielokrotność bazowej jednostki (np. $x * 1 \text{ PGB}$, $y * 50 \text{ FPA}$).
 - Aplikacja informuje o dokładnym czasie, z którego pochodzą aktualne kursy akcji.
- ▶ Dodatkowe: (w przypadku, gdy podstawowe funkcjonalności zostaną zaimplementowane)
 - Aplikacja na żądanie użytkownika generuje wykres średnich kursów akcji z ostatnich 20 notowań.
 - Użytkownik może edytować swoje dane oraz portfel.

WYMAGANIA TECHNICZNE:



- ▶ Kursy walut należy pobierać z udostępnionego przez Future Processing serwera. Udostępnia on dane za przy użyciu protokołu HTTP pod adresem <http://webtask.future-processing.com:8068/stocks> oraz poprzez protokół WebSockets pod adresem <ws://webtask.future-processing.com:8068/ws/stocks>
- ▶ Domyślnym formatem zwracanych danych jest JSON. Aby uzyskać dane w innej postaci należy:
 - W przypadku użycia WebSocketów dodać do URL parametr format (format=xml lub format=json).
 - W przypadku połączeń za pomocą protokołu HTTP należy dodać do żądania nagłówek Accept (o wartości application/xml lub application/json).
- ▶ Kursy akcji mogą zmieniać się co 20-30 sekund. Aplikacja zawsze musi wyświetlać aktualne kursy akcji. W przypadku awarii serwera z kursami lub braku możliwości nawiązania połączenia nie powinno być możliwości dokonania operacji finansowych.
- ▶ Rozwiązanie powinno wykorzystywać technologie oparte o AJAX i/lub WebSockets.
- ▶ Dla uproszczenia zarówno giełda jak i użytkownicy operują akcjami tych samych spółek: Future Processing (FP), FP Lab (FPL), FP Coin (FPC), ProgressBar (PGB), FP Adventure (FPA), DeadLine24 (DL24). Walutą rozliczeniową jest polski złoty (PLN) i w tej walucie należy podawać kursy akcji na giełdzie.
- ▶ Początkowa ilość akcji różnych spółek na giełdzie jest definiowana w bazie danych podczas tworzenia aplikacji.
- ▶ Aplikacja powinna być zabezpieczona przed atakami z zewnątrz (np. SQL Injection, XSS itp.).
- ▶ W ramach zadania powinna zostać stworzona baza danych.
- ▶ Całość rozwiązania (tzn. zarówno kod jak i interfejs użytkownika) powinna być w języku angielskim.
- ▶ Do rozwiązania zadania można użyć dowolnej technologii, dowolnego języka programowania i dowolnych narzędzi.
- ▶ W kwestii uruchamiania aplikacji należy spełnić jeden z trzech warunków:
 - Wszystkie niezbędne do uruchomienia aplikacji pliki konfiguracyjne znajdują się w jej kodzie. Jest załączona dokumentacja opisująca manualne kroki wymagane do jej uruchomienia w środowisku developerskim.
 - Uruchamianie i konfigurowanie aplikacji jest zautomatyzowane za pomocą odpowiednich narzędzi, na przykład Vagrant lub Docker. Procedura uruchomienia aplikacji jest udokumentowana.
 - Aplikacja jest zdeployowana w chmurze publicznej (na przykład AWS, Azure, Heroku, Openshift) lub na dowolnym hostingu pod podanym adresem URL.

INTERFEJS UŻYTKOWNIKA

Interfejs użytkownika powinien zostać utworzony zgodnie z poniższym mock-upem (ew. inne strony, okna dialogowe, itp. należy zaprojektować samodzielnie):

Stocks

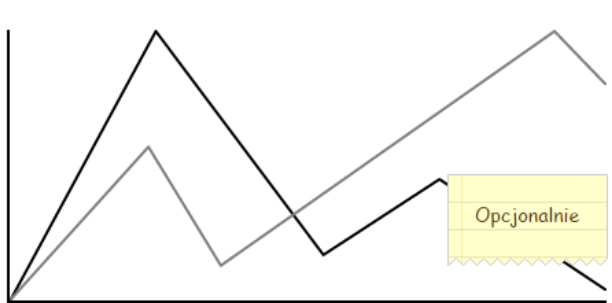
Logged in as John Doe

Edycja profilu użytkownika

Stock prices

Company	Value	Actions
FPA	10.54	<button>Buy</button>
DL24	5.12	<button>Buy</button>
FPL	45.28	<button>Buy</button>
FPC	0.14	<button>Buy</button>



My wallet

Company	Unit price	Amount	Value	Actions
FPA	10.54	100	1054	<button>Sell</button>
DL24	5.12	10	51.2	<button>Sell</button>
FPL	45.28	100	4528	<button>Sell</button>
FPC	0.14	20	2.8	<button>Sell</button>

Available money: 1500 PLN

Program powinien być najwyższej jakości i charakteryzować się zamkniętą funkcjonalnością, czyli wszystkie zaimplementowane funkcjonalności muszą działać od początku do końca.

Program powinien demonstrować rozsądne użycie możliwości języka w zakresie abstrakcji, interfejsów, dziedziczenia, dobrą strukturę klas oraz pokazywać dobre praktyki programistyczne. **Zwracamy uwagę na jakość kodu i projektu.**

Nieprzekraczalny czas na wykonanie zadania: **4 dni**.