



Neumann János Egyetem
Műszaki és Informatikai
Kar

Java alkalmazások

Előadás – Beadandó feladat

Petrik Tibor
GDAZA8

2025

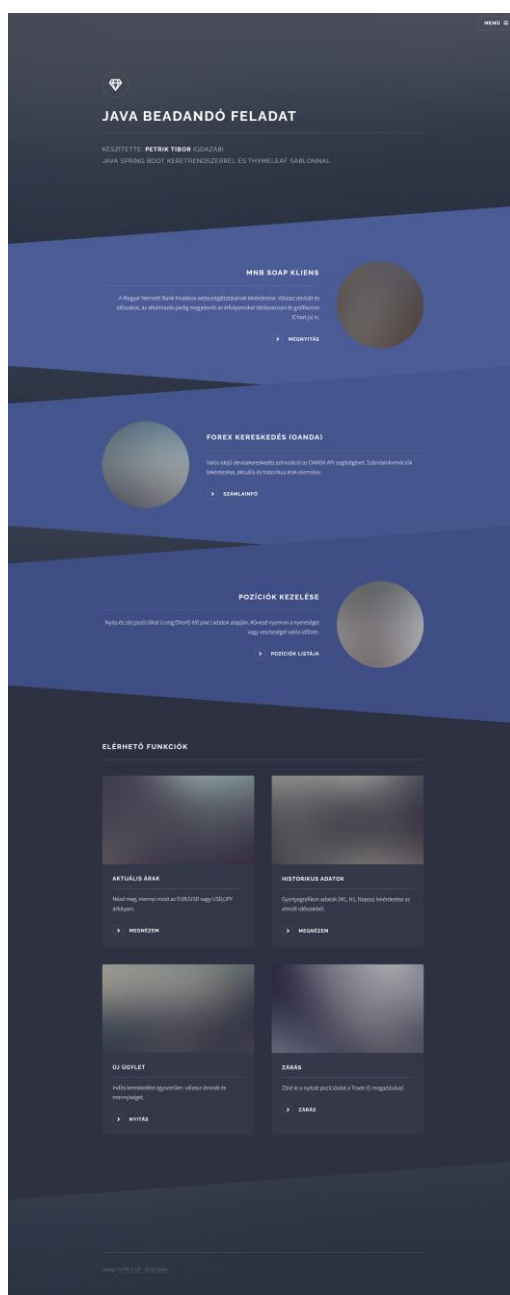
Tartalomjegyzék

BEVEZETÉS	7
1. FELHASZNÁLÓI FELÜLET ÉS DESIGN.....	8
2. MNB SOAP KLIENS MEGVALÓSÍTÁSA.....	10
2.1. MEGVALÓSÍTÁS.....	10
2.2. ADATFELDOLGOZÁS ÉS MEGJELENÍTÉS	10
3. FOREX KLIENS (OANDA) MEGVALÓSÍTÁSA.....	12
3.1. TECHNIKAI HÁTTÉR.....	12
3.2. SZÁMLAINFORMÁCIÓK ÉS ÁRFOLYAMOK.....	12
3.3. POZÍCIÓ NYITÁSA	12
4. FUTTATÁS ÉS KÖRNYEZET	14
5. TECHNIKAI ADATOK.....	15

Bevezetés

A féléves beadandó feladat keretében egy szerver oldali webalkalmazást készítettem Java nyelven, a Spring Boot keretrendszer használatával. A projekt célja egy olyan pénzügyi információs rendszer létrehozása volt, amely képes hivatalos árfolyamokat lekérdezni a Magyar Nemzeti Banktól (SOAP protokollon keresztül), valamint valós idejű devizakereskedést szimulálni egy nemzetközi bróker (OANDA) API-jának segítségével.

A fejlesztéshez a Java 21-es verziót és a Maven build rendszert használtam. A felhasználói felület kialakításához a Thymeleaf sablonkezelőt és a HTML5/CSS3 technológiákat alkalmaztam.



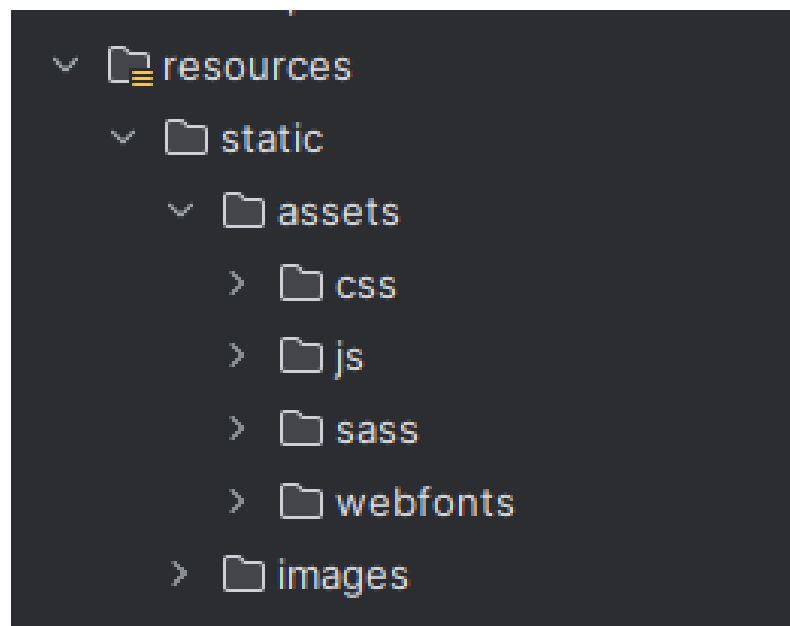
1. Felhasználói felület és design

A feladat kiírásának megfelelően egy reszponzív, modern megjelenésű sablont kerestem. Választásom a HTML5 UP "Solid State" nevű ingyenes sablonjára esett.

Megvalósítás: A letöltött sablon fájljait integráltam a Spring Boot projekt mappaszerkezetébe.

A stíluslapokat (CSS), szkripteket (JS) és képeket a src/main/resources/static/assets mappába helyeztem.

A HTML fájlokat a src/main/resources/templates mappába másoltam és átalakítottam őket Thymeleaf sablonokká.



Főoldal (index.html) kódrészlet: A menürendszert úgy alakítottam át, hogy a Java Controller által kezelt végpontokra mutasson:

```
19 <nav id="menu">
20   <div class="inner">
21     <h2>Menü</h2>
22     <ul class="links">
23       <li><a href="/">Főoldal</a></li>
24       <li><a href="/soap">SOAP (MNB Árfolyam)</a></li>
25       <li><a href="/account_info">Forex - Számla</a></li>
26       <li><a href="/actual_prices">Forex - Aktuális Ár</a></li>
27       <li><a href="/hist_prices">Forex - Historikus Ár</a></li>
28       <li><a href="/positions">Forex - Nyitott Pozíciók</a></li>
29       <li><a href="/open_position">Forex - Nyitás</a></li>
30     </ul>
31     <a href="#" class="close">Close</a>
32   </div>
33 </nav>
```

2. MNB SOAP Kliens megvalósítása

2.1. Megvalósítás

A feladat egyik fő része az MNB hivatalos árfolyamainak lekérdezése volt SOAP webszolgáltatáson keresztül.

Technikai háttér: A kommunikációhoz a jaxws-maven-plugin-t használtam a pom.xml-ben. Ez a plugin a fordítási időben automatikusan legenerálta a szükséges Java osztályokat az MNB WSDL leírása alapján (<http://www.mnb.hu/arfolyamok.asmx?wsdl>).

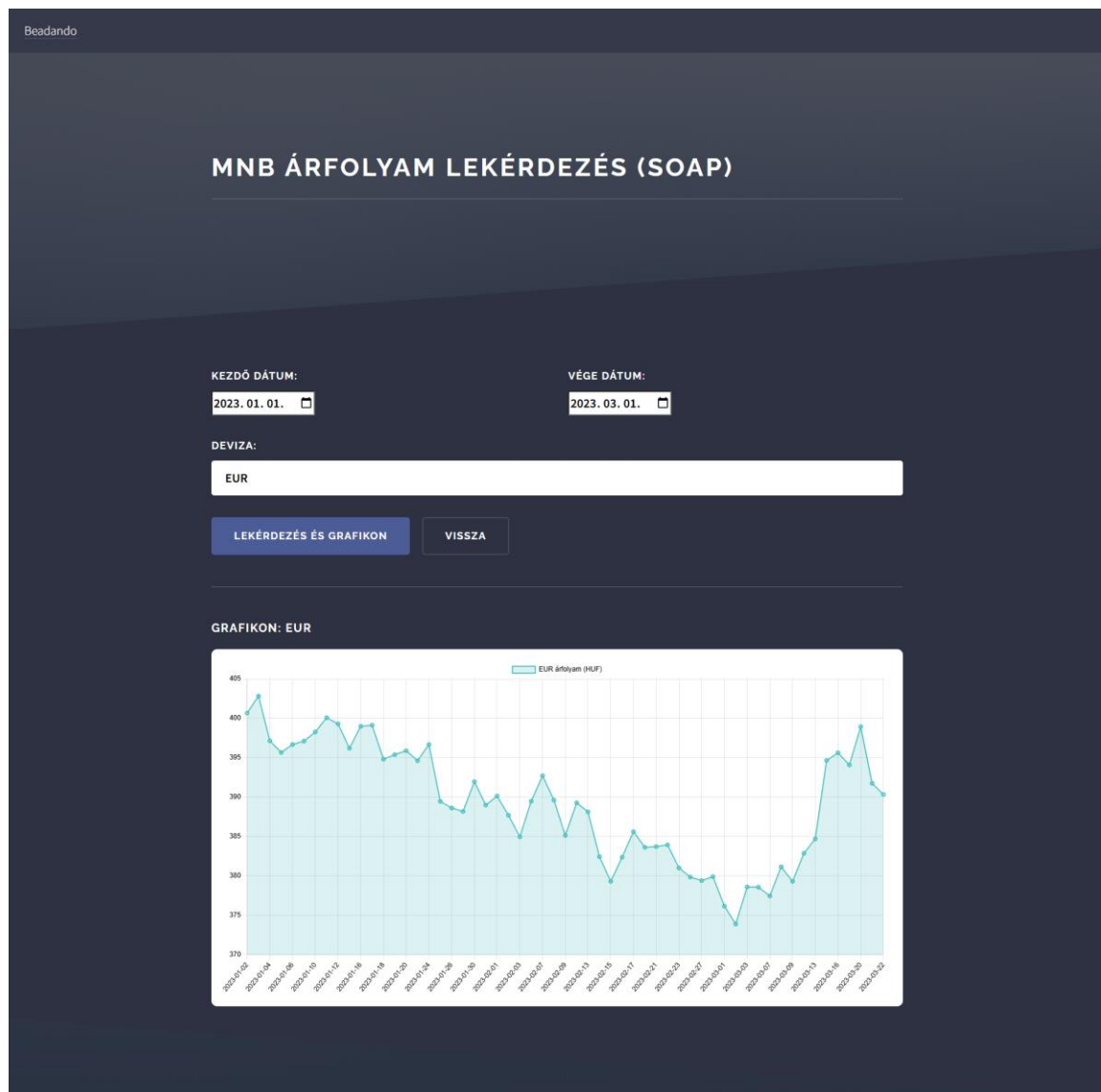
```
53     <build>
54         <plugins>
55             <plugin>
56                 <groupId>org.springframework.boot</groupId>
57                 <artifactId>spring-boot-maven-plugin</artifactId>
58             </plugin>
59
60             <plugin>
61                 <groupId>com.sun.xml.ws</groupId>
62                 <artifactId>jaxws-maven-plugin</artifactId>
63                 <version>4.0.2</version>
64                 <executions>
65                     <execution>
66                         <goals>
67                             <goal>wsimport</goal>
68                         </goals>
69                     </execution>
70                 </executions>
71                 <configuration>
72                     <wsdlUrls>
73                         <wsdlUrl>http://www.mnb.hu/arfolyamok.asmx?wsdl</wsdlUrl>
74                     </wsdlUrls>
75                     <packageName>soapclient</packageName>
76                     <sourceDestDir>src/main/java</sourceDestDir>
77                 </configuration>
78             </plugin>
79         </plugins>
80     </build>
```

2.2. Adatfeldolgozás és megjelenítés

A SoapController osztályban hívtam meg a generált szolgáltatást. Mivel az MNB XML formátumban adja vissza az adatokat, ezeket a Java beépített

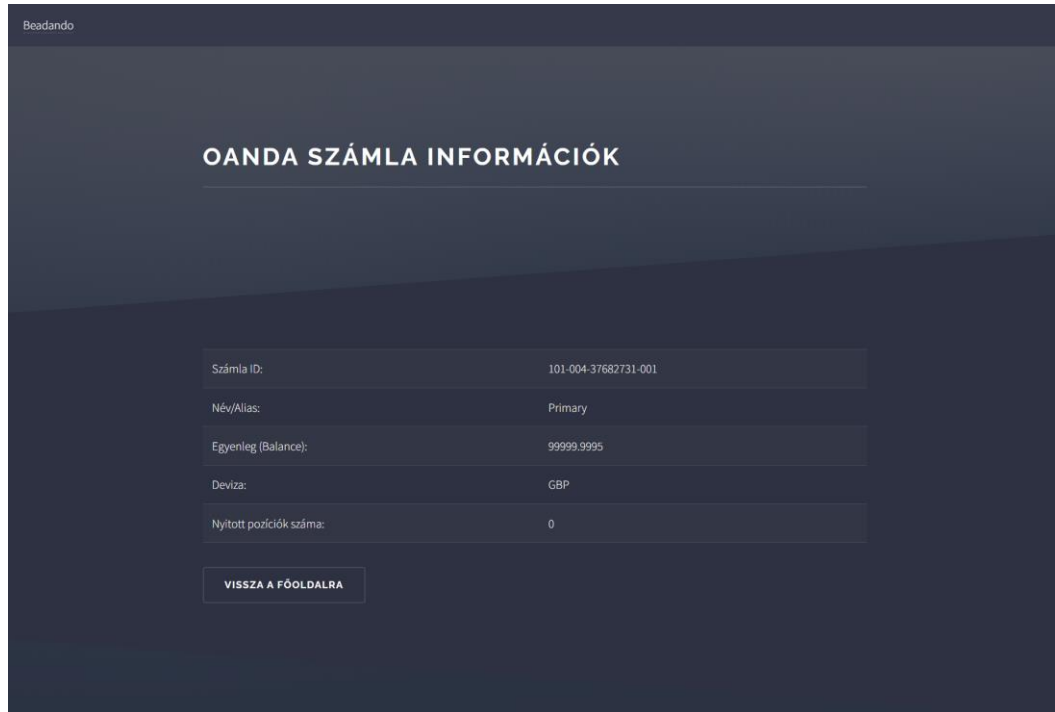
DocumentBuilder osztályával dolgoztam fel (parse-oltam), hogy kinyerjem a dátumokat és az értékeket.

A grafikonos megjelenítéshez a kliens oldalon a Chart.js könyvtárat használtam, amelynek a Controller adja át az adatokat.



3. Forex kliens (OANDA) megvalósítása

A devizakereskedési funkciókat az OANDA bróker API-jának segítségével valósítottam meg.



3.1. Technikai háttér

A projekthez hozzáadtam a v20-java könyvtárat, amely leegyszerűsíti a kommunikációt az OANDA szerverével. A hitelesítéshez szükséges adatokat (Token, Account ID) egy külön Config osztályban tároltam.

3.2. Számlainformációk és Árfolyamok

A ForexController osztály végzi a kommunikációt. A számlaadatok lekéréséhez a `ctx.account.summary()` metódust, az árakhoz pedig a `PricingGetRequest` osztályt használtam.

3.3. Pozíció nyitása

A pozíciók nyitása piaci áras megbízással (Market Order) történik. A felhasználó kiválasztja a devizapárt és a mennyiséget, a program pedig elküldi a kérést.

Beadando

POZÍCIÓ NYITÁS

DEVIZAPÁR:

EUR / USD

MENNYISÉG (EGYSÉG):

10

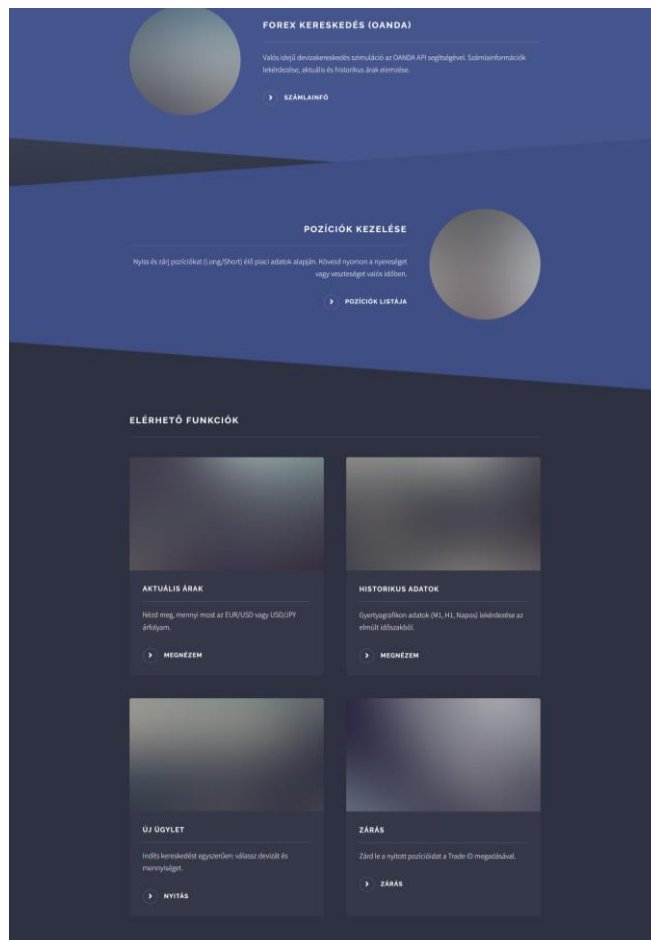
(Pozitív = Long/Vétel, Negatív = Short/Eladás)

NYITÁS

VISSZA

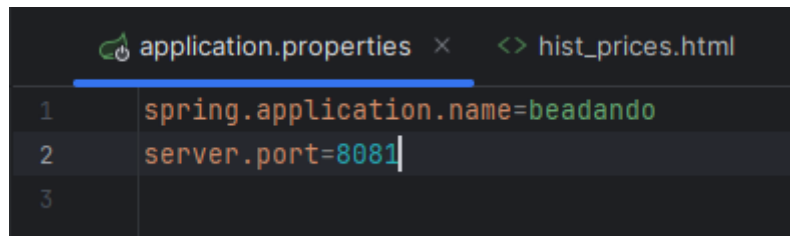
HIBA: A MEGBÍZÁST VISSZAUTASÍTOTTÁK. OK: MARKET_HALTED

A tesztelés: Hétvégén, amikor a devizapiacok zárva tartanak, a rendszer a 'MARKET_HALTED' üzenettel tér vissza. Ez igazolja, hogy a kommunikáció a bróker szerverével sikeres, a kérés eljutott a célba és a válasz feldolgozásra került.



4. Futtatás és környezet

A fejlesztés során felmerült, hogy a 8080-as port foglalt volt, ezért a szerver portját átállítottam 8081-re az application.properties fájlban:



```
application.properties x hist_prices.html
1 spring.application.name=beadando
2 server.port=8081
3
```

A projektből futtatható **JAR fájl készült** (beadando.jar) a Maven package parancsával.

5. Technikai adatok

GitHub Repository

<https://github.com/PetrikTibi/JAVA-alkalmazasok-EA-PT>

OANDA Belépési adatok

nagyjozsef19560115@gmail.com

JavaBeadandoPetrik2025!

Account ID:

101-004-37682731-001

API Token:

0bb22bce1a232d53b941570b2e5b57f0-2284b70ad2c10cdb3a8dd45774f1878c