**Dokumentáció**

**Web-programozás II.**

**Napfény Tours utazási iroda**

**2023/2024/1**

Taksár Tamás Barnabás

SMQPQ5

Őrsi Bence

RQNJUO

**GitHub elérés:** <https://github.com/Bence22/TravelWebProg.git>

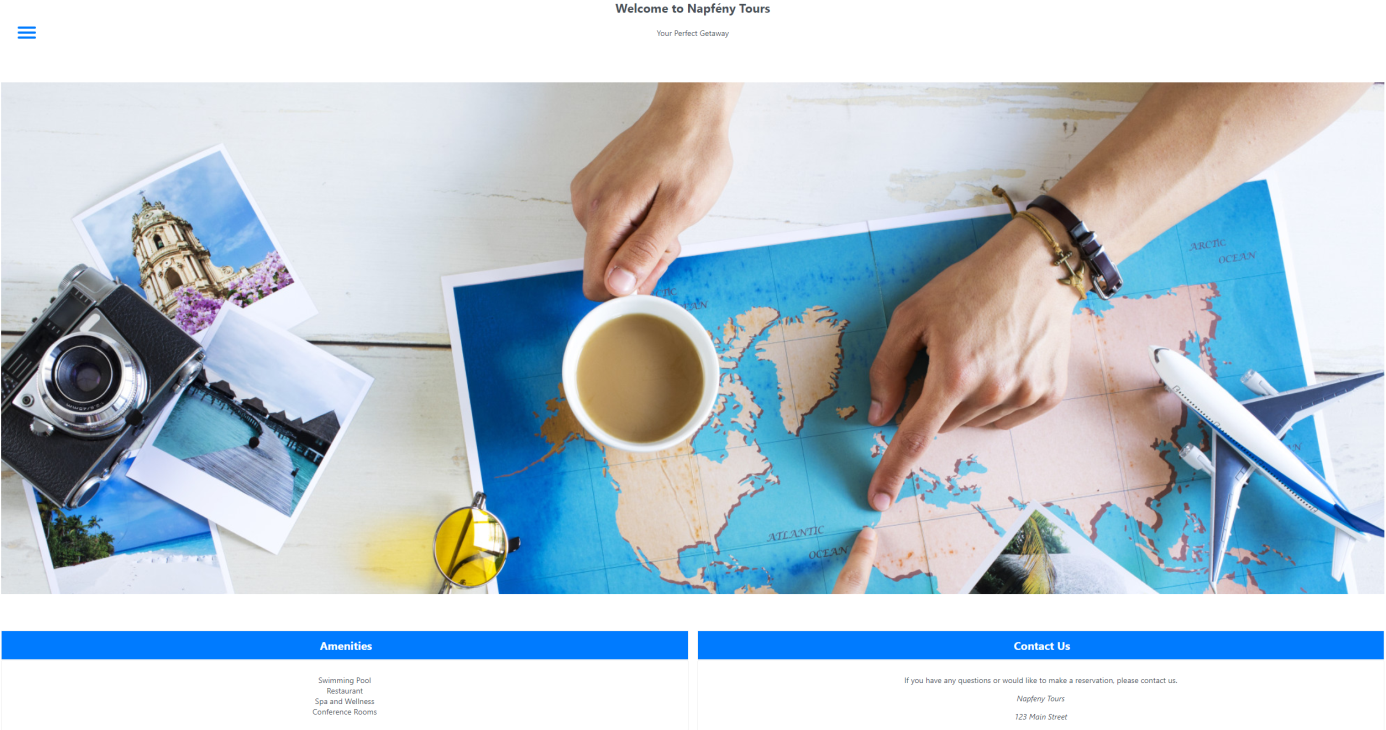
**URL:** <http://taksartamas.nhely.hu/>

**Nethely:**

Felhasználó: [taksar.tomi@gmail.com](mailto:taksar.tomi@gmail.com)

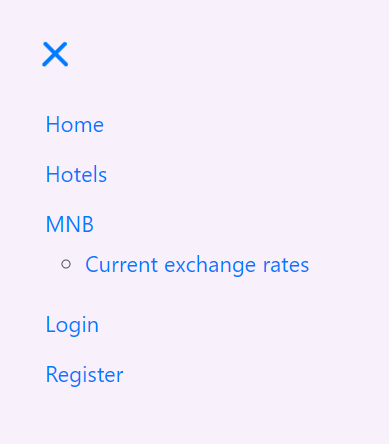
Jelszó: Web2Napfeny

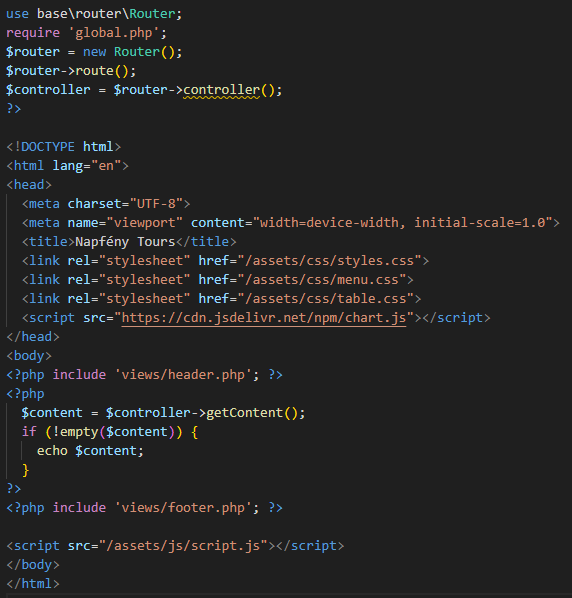
Weblapunk a Napfény Tours utazási irodának készült, amely segítséget nyújt az észak-afrikai és a közel-keleti régióba utazni vágyó felhasználóknak az utazásuk megtervezésében és megszervezésében. A látogatók telefonon, e-mailen vagy személyesen tudnak kapcsolatba lépni az iroda munkatársaival, akik segítenek az utazás megtervezésében és a foglalások lebonyolításában.

Az oldal kezdőlapja:

(1. feladat)

A honlap nyitó oldalán látható a cím és a szlogen, valamint alap információk a szállodák felszereltségéről és a cég elérhetőségéről. A navigálásban egy lenyíló menü segít a látogatóknak, aminek ikonja a bal felső sarokban található, ami jelzi, hogy a menü nyitott vagy zárt állapotban van-e. A menü pozíciója rögzített, ezért az oldalon való görgetés során folyamatosan elérhető.

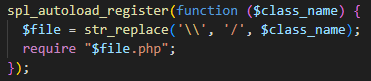


Az oldal reszponzív, tehát a böngésző ablakának mérete változtatása esetén dinamikusan változik a felület, hogy a látogatók mindig megfelelően és kényelmesen lássák az adatokat és az elemeket.

Jelenleg minden kérés az index.php-hoz érkezik, ahol egy routert helyeztünk el. Ennek az a feladata, hogy a beérkező kérés url-je alapján döntse el, hogy milyen controller osztályt kell példányosítani, illetve annak milyen metódusa felel a megjelenítendő tartalomért. Egy-egy lehetséges útvonalat egy reguláris kifejezéssel fedtünk el. 

A BaseController osztályunk gondoskodik arról, hogy minden származtatott kontrollerünknek legyen tartalma (getContent), gyakorlatilag ez fog meghívódni az index.php-ban.

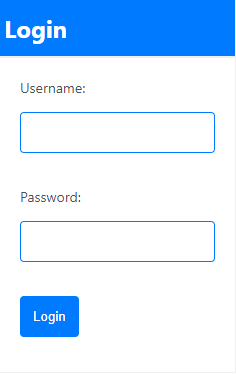
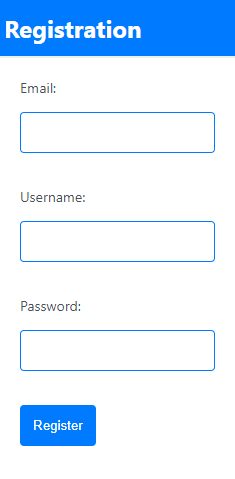
A namespacek gördülékeny használata érdekében bekerül a global.php-ba egy autoloader is, aminek mindössze annyi a feladata, hogy a névtér alapján “feloldja” az adott php fájlt, azaz, hogy ne kelljen manuálisan “berequierolni” minden egyes fájlt.



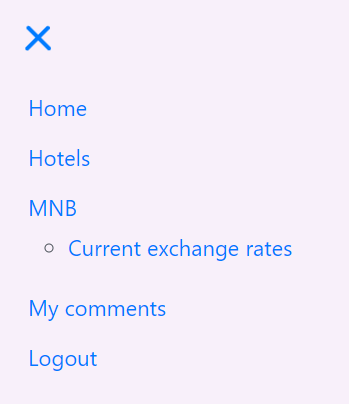
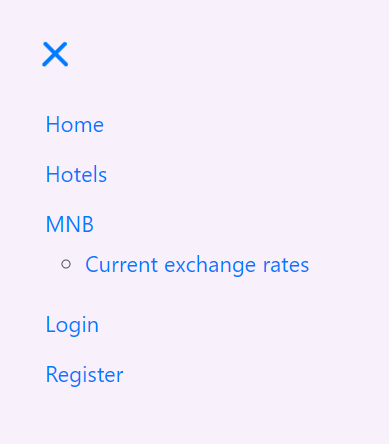
Az iroda által adott adatokat, amelyeket a honlapon kívántak viszont látni egy napfeny\_tours nevű SQL adatbázisban rögzítettük (’helyseg’, ’szalloda’, ’tavasz’ táblákban).

(2. feladat)

A weblapon böngészhetünk anonim látogatóként vagy regisztrált felhasználóként, tehát lehetőség van regisztrációra, bejelentkezésre és kijelentkezésre. Ezért egy ’users’ táblával bővítettük az adatbázist, ahol tároljuk a szükséges regisztrációs információkat [id, email, név, jelszó (titkosítva)]. A regisztrációs adatok elfogadása után nem léptet be automatikusan az oldal, hanem a bejelentkező (login) ablakra ugrik.

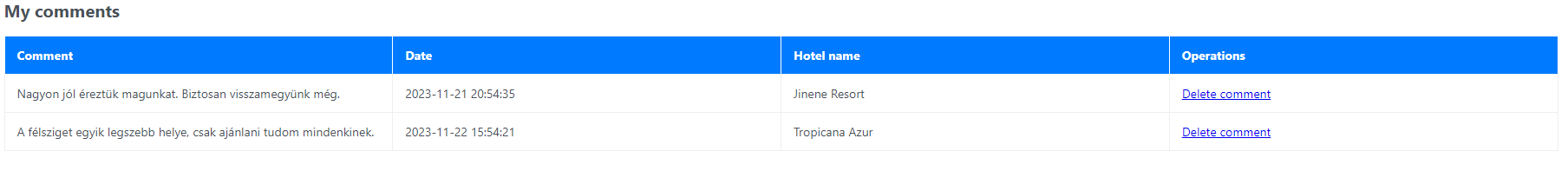


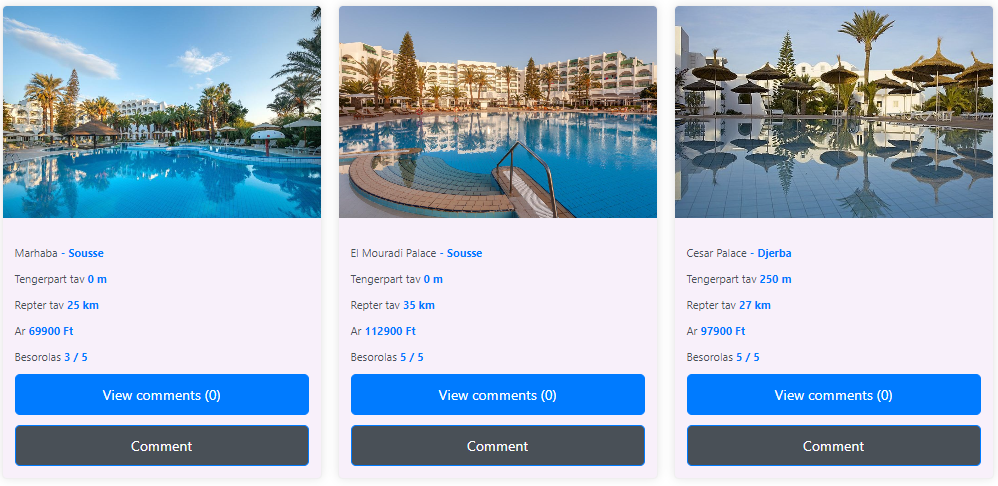
Bejelentkezést követően a fejlécben folyamatosan látható a bejelentkezett felhasználó neve és a legördülő menüben a ’Login’ és a ’Register’ lehetőségeket felváltja a ’Logout’ menüpont. A kijelentkezésre való kattintás után azonnal visszaáll az eredeti állapot.



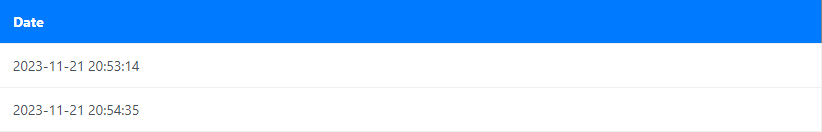
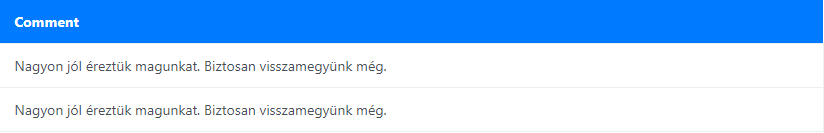
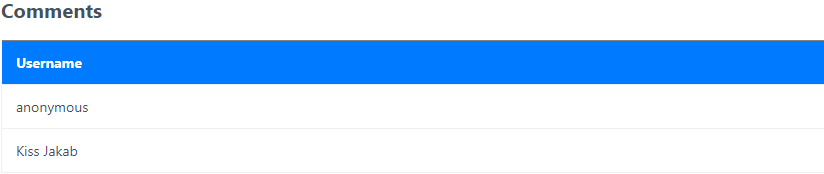
(3. feladat)

Ahogyan az előző képeken is látható a bejelentkezést követően a menüben megjelenik egy ’My comments’ opció, amellyel kilistázhatja a felhasználó a weblapon általa írt összes hozzászólását, illetve lehetősége van törölni őket.

Az adatbázisban szereplő hotelek és a hozzájuk tartozó adatok kártyás elrendezésben jelennek meg a képernyőn a ’Hotels’ menüpontra kattintva, ezen kártyák alsó részén található 2 gomb. Az egyikkel (View comments) a felhasználók és látogatók által írt kommenteket és híreket olvashatjuk az adott szállodával kapcsolatosan, a másikkal (Comment) pedig a hír és vélemény közzététele lehetséges.



A hírek és kommentek az adatbázis ’comments’ táblájában tárolódnak a felhasználó nevével, a hozzászólás idejével és a szálloda azonosítójával együtt, illetve innen hívódnak meg lekérdezéskor. A nem regisztrált látogatók hozzászólásai anonymous néven tárolódnak. Lekérdezés után az információk táblázatba rendezve jelennek meg a képernyőn.



(6. feladat)

A szállodák adatainak biztonságos elérését SOAP protokoll szolgáltatás megvalósításával biztosítottuk. A programunk HotelService osztálya valósítja meg az oldal üzleti logikáját és az adatok lekérdezését, és a HotelModel és HotelView osztályokkal együttműködve kezeli a szállodákkal kapcsolatos funkciókat. A HotelModel osztályt használja az adateléréshez, a HotelView osztály segítségével formázza és jeleníti meg a hotelek adatait.

HotelService metódusai:

* getHotel(string $az): array|false

Visszaadja a megadott azonosítóval rendelkező szálloda adatait.

Paraméterek: $az (string): A szálloda azonosítója.

Visszatérési érték: Szálloda adatainak tömbje vagy false, ha nincs találat.

* listHotels(): string

Visszaadja az összes rendelkezésre álló szálloda listáját formázott szövegként.

* getComments(string $szalloda\_az)

Visszaadja a megadott szállodához tartozó hozzászólásokat.

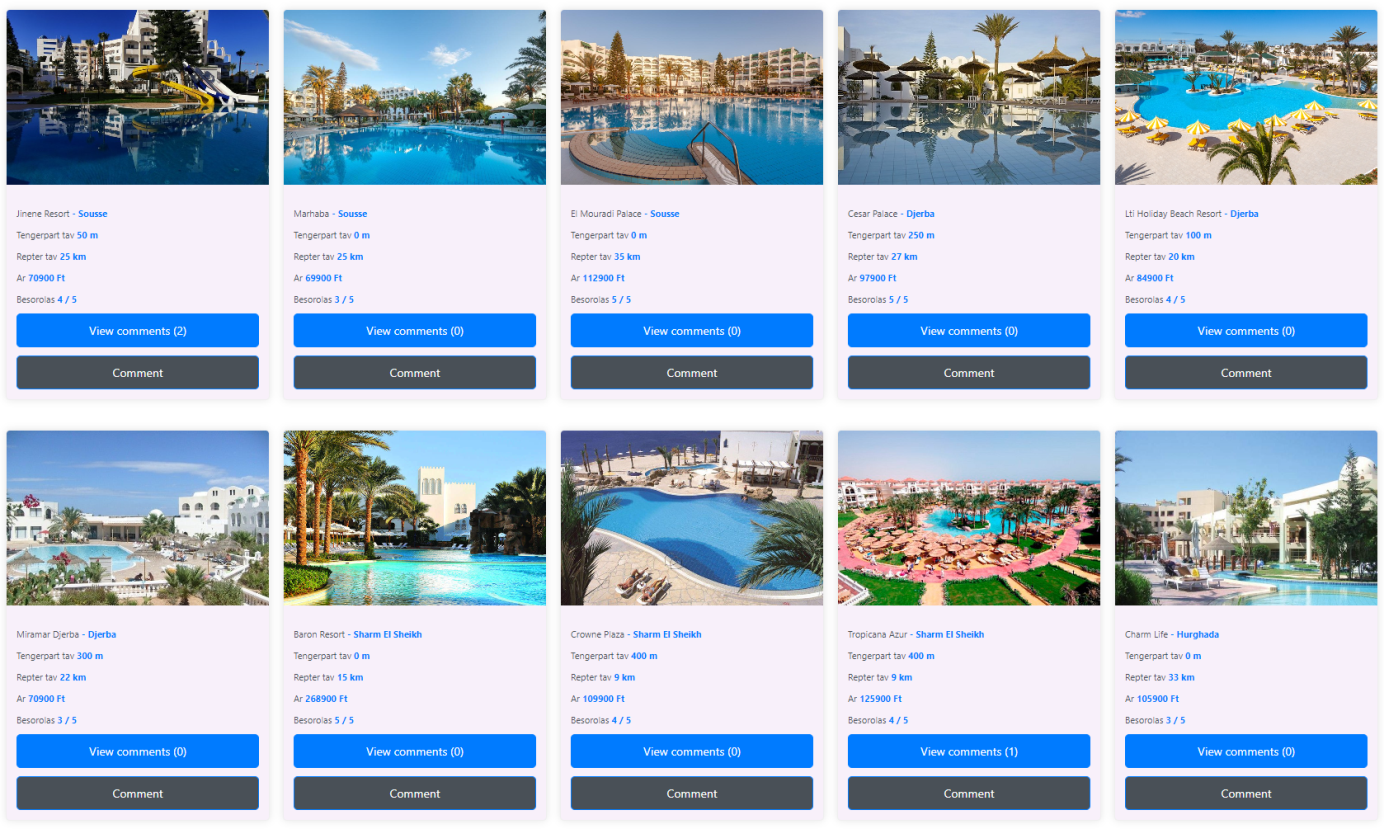
Paraméterek: $szalloda\_az (string): A szálloda azonosítója.



A SoapServer osztály részletesen konfigurálja és inicializálja a szerver objektumot, majd hozzárendeli a szolgáltatás logikáját egy osztállyal.

A kód definiál egy névteret (hotels\server), valamint importálja a hotels\service\HotelService névtérben található osztályt.

Beilleszti a globális global.php fájlt.

Konfigurálja a SOAP szerver opcióit. Létrehoz egy SoapServer objektumot a konfigurációval. Hozzárendeli a szerverhez a szolgáltatás logikáját az osztály (HotelService::class) segítségével. Elindítja a szerver kérések kezelését a handle függvénnyel.

(7. feladat)

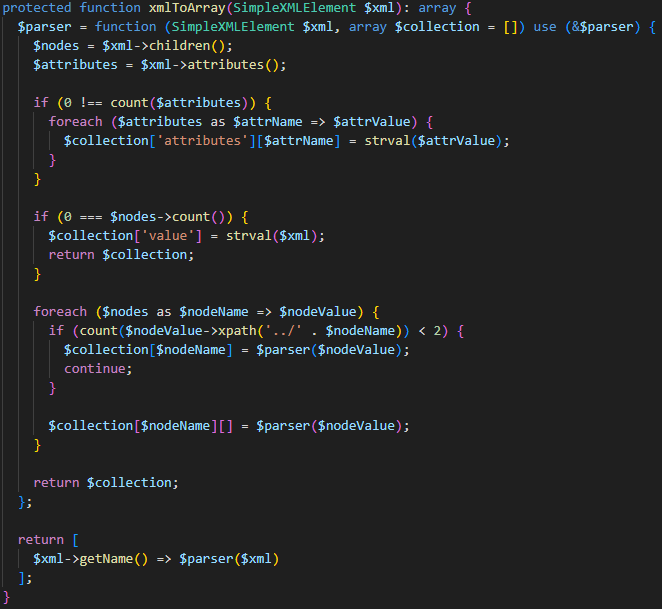
A látogatóknak segítséget nyújtva a külföldi utazásokhoz elhelyeztünk egy valuta árfolyam lekérdezési lehetőséget az oldalon, amit a menüből érhetnek el (Current Exchange Rates). Ezzel a funkcióval kiválaszthatják a számukra érdekes devizapárt a legördülő listában történő kiválasztással, és egy adott időpontot vagy tetszőleges időintervallumot egy kezdeti és egy végdátum megadásával, amelyen belül kíváncsiak az árfolyam változására és aktuális állására.

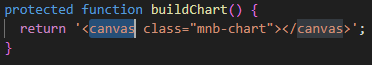
A ’search’ gomb lenyomása után a lekérdező ablak alatt megjelenik az összes választott adat egy táblázatba rendezve (dátum, árfolyam).

A lekérdező ablak felett pedig megjelenik egy grafikon, amelyen az előbb említett aktuális táblázat adatai láthatóak a látványosabb árfolyamváltozás szemléltetése céljából.

A lekérdezési adatok eléréséhez a Magyar Nemzeti Bank Soap szolgáltatását használtuk fel, így valóban mindig az aktuális hivatalos adatok kerülnek listázásra.

Az MNBServiceController-ben kezeljük a SOAP kliens által visszakapott adatokat. Ezeket XML-ben kapjuk meg, ezért szükség van ennek átalakítására a könnyebb használat érdekében. Az egy megjelenített form-mal a feladatban kért mindkét részfeladat egyszerre megvalósítható.



A grafikon nézet JavaScript segítségével valósul meg. Ehhez arra van szükség, hogy behúzzuk a megfelelő javascript fájlt egy külső CDN-ről, a dokumentáció szerint létrehozzunk egy canvas HTML elemet egy id attribútummal, majd a scriptben ezt az id-t megcélozva tudjuk manipulálni az adatokat.

