**Vak Bottyán János Katolikus   
Műszaki és Közgazdaság Technikum,   
Gimnázium és Kollégium**

**54 213 05 Szoftverfejlesztő  
szakképesítés**

***Your Fantasy Hair Studio***

**Készítette: Kis Gábor  
2023**

Tartalomjegyzék

[Tartalomjegyzék 2](#_Toc164111005)

[Előszó 4](#_Toc164111006)

[Bevezető 5](#_Toc164111007)

[Megvalósíthatósági elemzés 6](#_Toc164111008)

[Elvárások: 6](#_Toc164111009)

[Megvalósíthatósági elemzés: 6](#_Toc164111010)

[Opcionális (ha nem helyi szerverről fut): 7](#_Toc164111011)

[Alkalmazni kívánt programozási nyelvek 7](#_Toc164111012)

[Követelmények (funkcionális): 7](#_Toc164111013)

[Követelmények (nem funkcionális): 8](#_Toc164111014)

[Működés: 8](#_Toc164111015)

[Felhasználói felület: 8](#_Toc164111016)

[Programterv 8](#_Toc164111017)

[Design terv: 9](#_Toc164111018)

[Felhasználói dokumentáció 10](#_Toc164111019)

[A program célja: 10](#_Toc164111020)

[Környezeti leírás: 10](#_Toc164111021)

[Használati leírás: 11](#_Toc164111022)

[Főoldal felhasználói felülete: 11](#_Toc164111023)

[Regisztráció: 12](#_Toc164111024)

[Bejelentkezés: 13](#_Toc164111025)

[Időpontfoglalás: 14](#_Toc164111026)

[Időpontok: 15](#_Toc164111027)

[Időpontjaim: 16](#_Toc164111028)

[Galéria: 17](#_Toc164111029)

[Áraink: 18](#_Toc164111030)

[Lehetséges hibaüzenetek 19](#_Toc164111031)

[Fejlesztői dokumentáció 21](#_Toc164111032)

[Futási környezet: 21](#_Toc164111033)

[Programnyelvek: 21](#_Toc164111034)

[Követelmények (funkcionális): 24](#_Toc164111035)

[Követelmények (nem funkcionális): 24](#_Toc164111036)

[Adatbázis felépítése 25](#_Toc164111037)

[Táblák: 25](#_Toc164111038)

[Programkód megjelenítése 27](#_Toc164111039)

[Fejlesztői környezet 34](#_Toc164111040)

[Készítő 34](#_Toc164111041)

[Fejlesztési lehetőségek 35](#_Toc164111042)

[Tesztelési dokumentáció 36](#_Toc164111043)

[Tesztelés összegzése 37](#_Toc164111044)

[Fejlesztés után: 37](#_Toc164111045)

[Felhasznált irodalom, eszközök: 38](#_Toc164111046)

[Felhasznált irodalom 38](#_Toc164111047)

[Felhasznált eszközök: 38](#_Toc164111048)

[Ábrajegyzék 39](#_Toc164111049)

[Összegzés 40](#_Toc164111050)

[Köszönetnyilvánítás 40](#_Toc164111051)

[Plágium nyilatkozat 41](#_Toc164111052)

Előszó

A XXI. században egyre több vállalkozás használja az internet által nyújtott lehetőségeket. Ilyenek lehetnek például a saját közösségi oldal, weboldal, mobilalkalmazás stb. Ezek elősegíthetik a cégek fellendítését, hogy minél több emberhez juthassanak el a termékeik vagy szolgáltatásaik. Én is egy ilyen lehetőséget ragadtam meg a témaválasztásomkor. Weboldalt készítettem egy fodrászat számára. Ezen a weboldalon időpontot lehet foglalni a *Your Fantasy Hair Studio* nevű fodrászatba. Az ötlet onnan jött, hogy az én fodrászomhoz, ahova szoktam járni csak telefonon vagy személyesen lehet időpontot foglalni. Ezzel nincs semmi baj, ha akkor szeretnénk időpontot foglalni, amikor éppen nyitva van, de ha például este 10 órakor jut eszünkbe, hogy még nem foglaltunk időpontot, akkor ezt könnyedén megtehetjük egy weboldal használatával, így biztosan nem felejtjük el a mindennapi rohanás során. Csupán pár perc alatt tudunk időpontot foglalni, ezzel időt és energiát is spórolhatunk. A weboldallal arra törekedtem, hogy egy jól átlátható, könnyen használható oldalt hozzak létre, amelyen keresztül mindenki gond nélkül foglalhat időpontot.

Bevezető

A záródolgozat témája egy fodrászat weboldala. A weboldal lehetőséget biztosít arra, hogy az ügyfelek egyszerűen időpontot foglalhassanak egy adott fodrászati szalonban. A felhasználók, miután bejelentkeztek, könnyedén kiválaszthatják a kívánt időpontot, és a foglalásukat később is visszanézhetik, ha szükséges. A fodrászok is láthatják, hogy egy adott nap mely ügyfeleknek és mikorra van időpontjuk.

A weboldal lehetővé teszi a referencia-munkák megtekintését, ezzel segítve a felhasználókat a frizura választásában. Ez egyben egy jó reklám is lehet a cégnek az ügyfelek számára. Továbbá az aktuális árak is fel vannak tüntetve részletesen az oldalon, hogy ne érje meglepetés a vásárlókat a szalonban.

Az oldal szerintem mindenki számára jól használható, az átláthatóság miatt, legyen az idősebb vagy fiatalabb.

Megvalósíthatósági elemzés

Elvárások:

A felhasználó informatikai eszköz segítségével időpontot tud foglalni.

A felhasználó tud regisztrálni és bejelentkezni is már meglévő fiókjába.

Az oldalon a felhasználó megtudja tekinteni lefoglalt időpontját

A fodrász látja, hogy kinek és mikorra van foglalt időpontja

Megfelelő reszponzívitás (akár mobil nézeten is)

Hiba esetén visszajelzés a felhasználónak

Megvalósíthatósági elemzés:

Humán erőforrás: 1 fejlesztő, Óraszám: Kb.120-140

**Hardver erőforrás:**

**Minimum:**

Operációs rendszer: Windows 7, 64 bit

Processzor: 1.8 GHz, 4 magos, 64 bit

Memória: 4 GB

Háttértár: Minimum 800MB – Ajánlott 200 GB

Periféria: Egér, billentyűzet, monitor

**Ajánlott:**

Operációs rendszer: Windows 10, 64 bit

Processzor: 2.4 GHz, 4 magos, 64 bit +

Memória: 8 GB +

Háttértár: 210 GB +

Periféria: Egér, billentyűzet, monitor

**Szükséges szoftverek:**

Visual Studio Code

PHP Server (XAMPP php konfigurációval)

Internetböngésző (HTML 5 támogatással)

Opcionális (ha nem helyi szerverről fut):

A várt felhasználók számához képest egy megfelelő webszerver üzemeltetése

Alkalmazni kívánt programozási nyelvek

* HTML
* CSS
* JS
* PHP
* SQL

Követelmények (funkcionális):

**Főoldal megjelenésekor:**

* Menüpontok közötti váltás
* Bejelentkezési lehetőség megjelenése
* Cikkek megjelenése

**Hatékonyság:**

* Az oldal betöltése nagy felhasználói igény esetén lassulhat.
* Gyors válaszidő
* Pontos visszajelzés

**Megbízhatóság:**

• Rendeltesésszerű használat esetén nem lép fel hiba

Követelmények (nem funkcionális):

**Biztonság:**

Biztonságos adatkommunikáció a helyi szerverrel (XAMPP szerver).

**Felhasználhatóság:**

Könnyen használható, megfelelő instrukciók találhatóak az oldalon

**Környezet:**

Egy megfelelő webböngészőre van szükség.

Működés:

* Gyors működés
* Megbízható használat
* Biztonságos működés
* A hibalehetőségek kizárása

Felhasználói felület:

* Szép design
* Átlátható felhasználói felület
* Könnyen kezelhető

Programterv

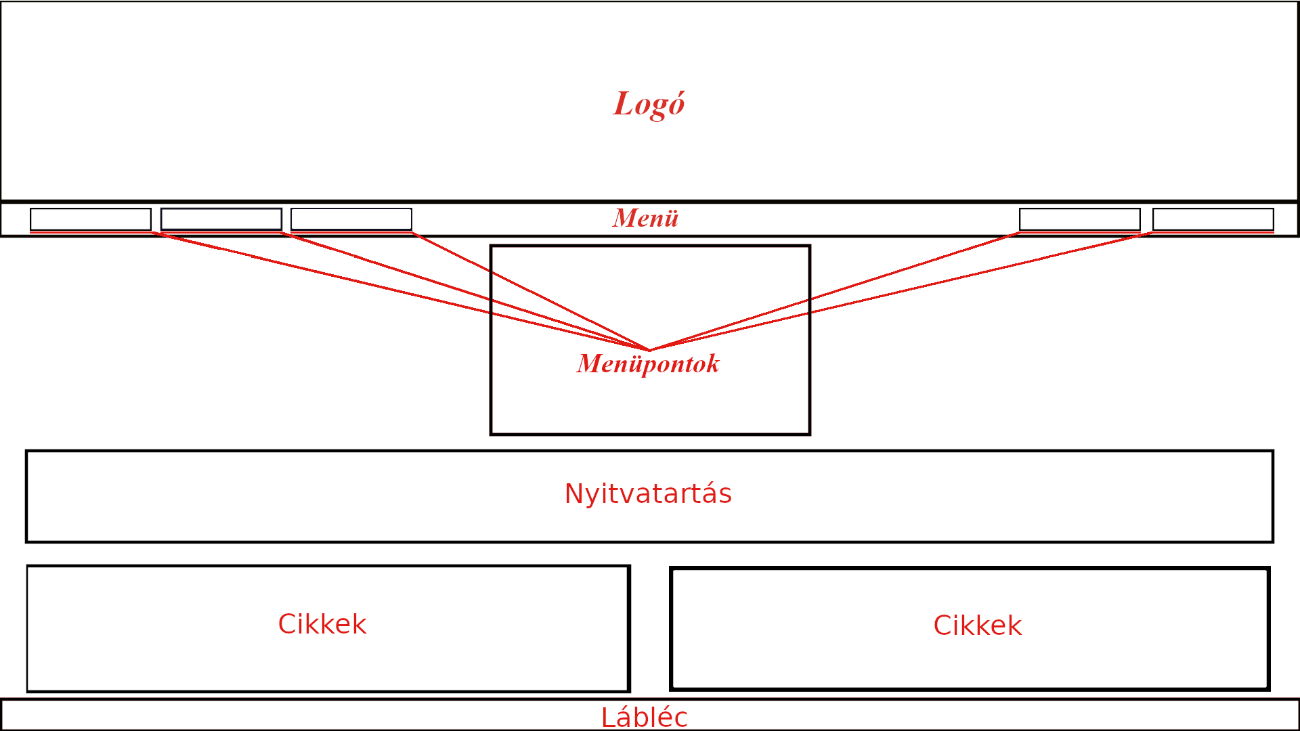
A főoldalon ezek a fülek lesznek elérhetőek:

* Időpontfoglalás
* Áraink
* Galéria
* Belépés
* Regisztráció
* Időpontjaim/Időpontok (felhasználó szintje alapján változik)

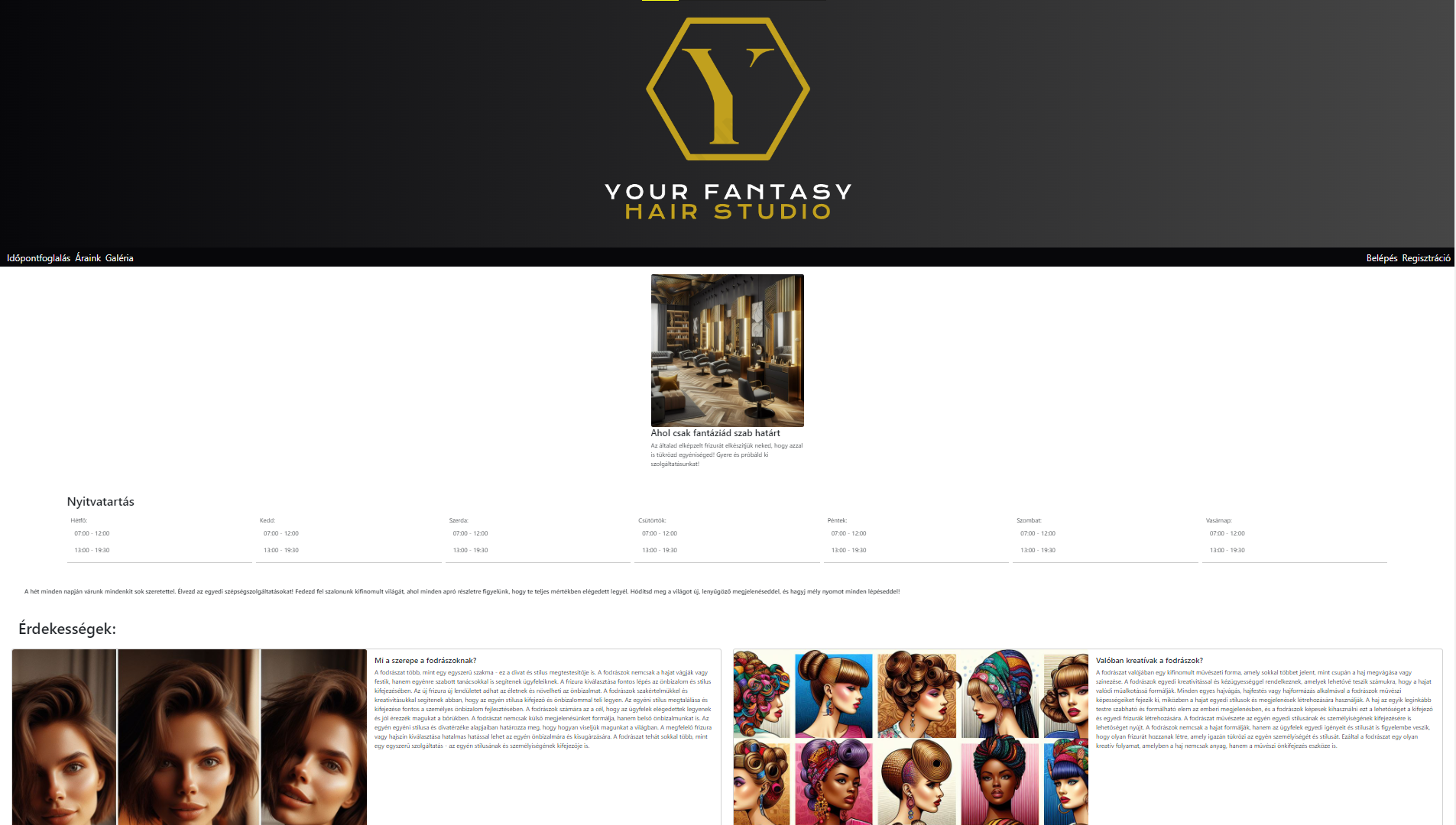
Design terv:

Kezdőlap design nézete:

1. ábra kezdőlap design terv



Kezdőlap végleges kinézete:



2. ábra kezdőlap végleges kinézet

Felhasználói dokumentáció

A program célja:

Ez a weboldal segítség mindenki számára, ugyanis bárki regisztrálhat és foglalhat időpontot a fodrász szalonba. Az oldalon a felhasználónak tetsző időben és tetszőleges napra foglalhat időpontot, ha még van arra az adott napra időpont. Továbbá korlátlan számú időpontot foglalhat le minden felhasználó, így akár egy évre előre is be tudja tervezni, hogy mikor megy fodrászhoz.

Környezeti leírás:

**Hardver erőforrások**

**Minimum:**

Operációs rendszer: Windows 7, 64 bit

Processzor: 1.8 GHz, 4 magos, 64 bit

Memória: 4 GB

Háttértár: Minimum 800MB – Ajánlott 210 GB

Periféria: Egér, billentyűzet, monitor

**Ajánlott:**

Operációs rendszer: Windows 10, 64 bit

Processzor: 2.6 GHz, 8 magos, 64 bit +

Memória: 16 GB +

Háttértár: 210 GB +

Periféria: Egér, billentyűzet, monitor

Használati leírás:

Mivel ez egy webalkalmazás ezért egy egyszerű keresőre lesz szükség, ami által megnyitja a felhasználó az oldalt. Az oldalon ajánlott a sötétmód használata, mert így minden tökéletesen fog megjelenni. Miután betöltődött az oldal szabadon használhatja, de egyes funkciók eléréséhez regisztráció vagy bejelentkezés szükséges. Ahhoz, hogy minden megfelelően működjön, a felhasználónak a számítógépén szüksége lesz Visual Studio Code programra, amiben telepítve van a PHP, valamint XAMPP programra, hogy kommunikálni tudjon a szerverrel amennyiben helyi számítógépről futtatja azt. A phpMyAdmin felületen új adatbázist kell létrehoznia, ehhez pedig csak annyit kell tennie, hogy importálja a YFHS.sql fájlt. Ezután A Visual Studio Code-ban PHP szerverként kell elindítani a main.html oldalt.

Főoldal felhasználói felülete:

3. ábra Főoldal felhasználói felülete

A felhasználói felület létrehozása során próbáltam egy egyszerű, átlátható, mindenki számára könnyen használható felületet létrehozni. A menüsoron 4 opció látható Időpontfoglalás, Áraink, Galéria, Időpontok. Az időpontfoglalás menüpontra kattintva elindul a foglalás folyamata. Ekkor kiválaszthatjuk az időpont napját és idejét majd a kért szolgáltatást. Az időpontot az oldal alján található gomb megnyomásával adhatjuk le.  
Áraink menüpont alatt az aktuális árak listája található. A galériára kattintva megtekinthetjük a fodrászok által készített frizurákat ábrázoló képeket. Az Időpontok menüpontra kattintva a felhasználó megtekintheti a már lefoglalt időpontját.

Regisztráció:

Mielőtt a felhasználó elkezdheti az időpontfoglalást azelőtt regisztrálnia kell. Miután rákattintott a menüsávban a regisztráció menüpontra, ekkor átirányítja az oldal a regisztrációs oldalra a felhasználót. Ekkor meg kell adni a Felhasználónevét, Jelszavát kétszer és a Teljes nevét, majd a regisztráció gombra kattintás után, ha minden jól van kitöltve akkor sikeresen regisztrált. Ha nem egyezik meg a két jelszó, vagy már foglalt a felhasználónév akkor figyelmeztetést kap a problémára, amelyet javíthat. Ezután a bejelentkezés oldalra kerül át.

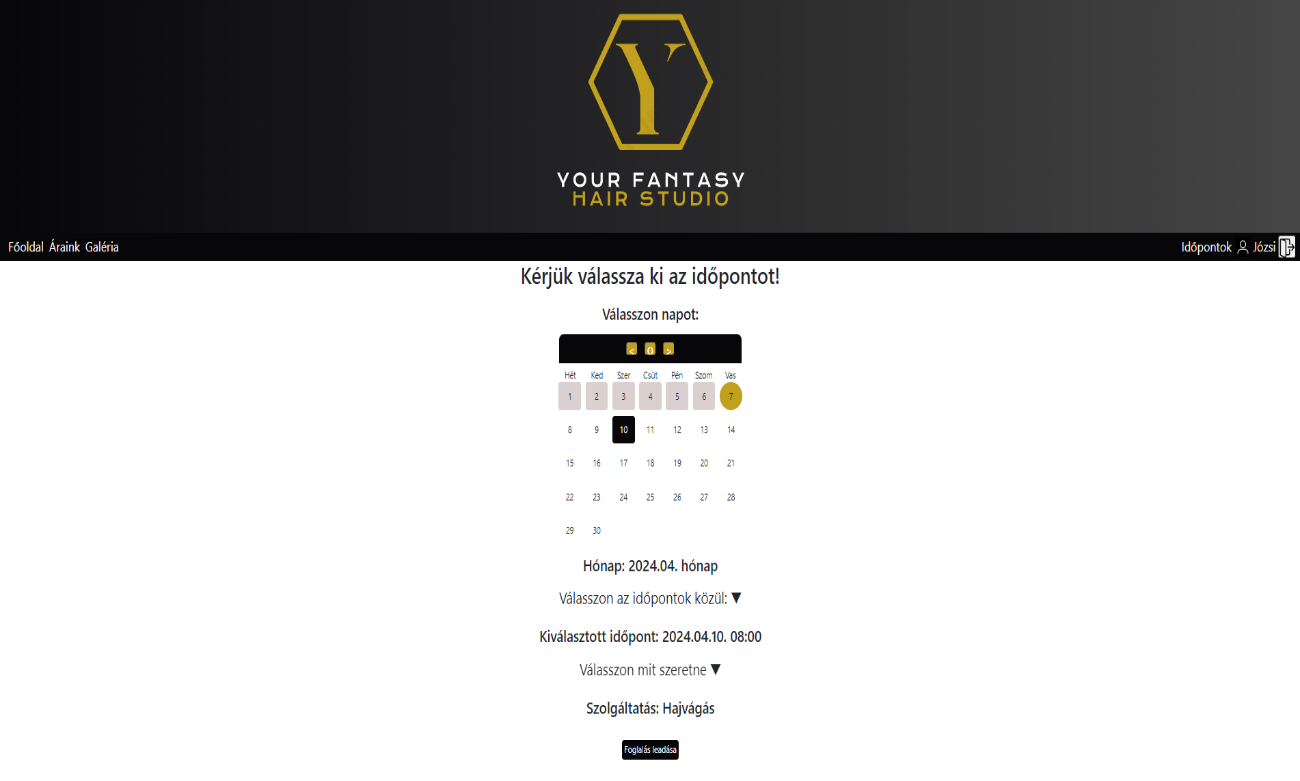
4. ábra Regisztráció oldal

Bejelentkezés:

****A regisztráció során megadott adatok segítségével léphet be, majd végezheti el a számára tetszőleges tevékenységet. Például foglalhat időpontot, vagy már a foglalt időpontját tekintheti meg. Akár tájékozódhat az árakról, előző munkáinkról. Ha regisztráció során megadott adatokat helyesen beírja akkor a bejelentkezés sikeres volt. Sikeres és sikertelen bejelentkezés esetén is értesíti a felhasználót a rendszer.

5**.** ábra Bejelentkezés ablak

Időpontfoglalás:

Bejelentkezés után már meg tudjuk nyitni az időpontfoglalás menüpontot a menüsávon. Itt az első lépés, hogy kiválassza a felhasználó a neki tetsző napot. Ezután meg fog jelenni a következő lehetőség az oldalon, ami a következő: *Válasszon az időpontok közül*. Ekkor lenyithatjuk a még szabad időpontok listáját. Az időpont kiválasztása után ki kell választani, hogy milyen szolgáltatást szeretne igénybe venni, majd az időpont véglegesítéséhez a *Foglalás leadása* gombot megnyomva lesz lehetőség. Sikeres és sikertelen időponfoglalás esetén értesíti a felhasználót a rendszer.

6**.** ábra: Időponfoglalás oldal

Időpontok:

Ez az ablak csak a regisztrált fodrászok számára elérhető. Fodrászként regisztrálni nem lehet, csak az weboldal fejlesztője tud új fodrászt regisztrálni. Az oldalon látható egy naptár, aminek csupán tájékoztató szerepe van. Ami még megtalálható, az a foglalt időpontok. Itt a felhasználó teljes neve, az időpont napja és pontos ideje és a kért szolgáltatás található meg.

7**.** ábra: Időpontok oldal

Időpontjaim:

Ez az ablak a regisztrált felhasználóknak érhető el bejelentkezés után. Bárki regisztrálhat felhasználóként, nincs szükség semmilyen engedélykérésre. Az oldalon látható egy naptár, aminek csupán tájékoztató szerepe van. Ami még megtalálható, az a felhasználó időpontjai. Itt az időpont napja és pontos ideje és a kért szolgáltatás található meg.

8.ábra: Időpontjaim oldal

Galéria:

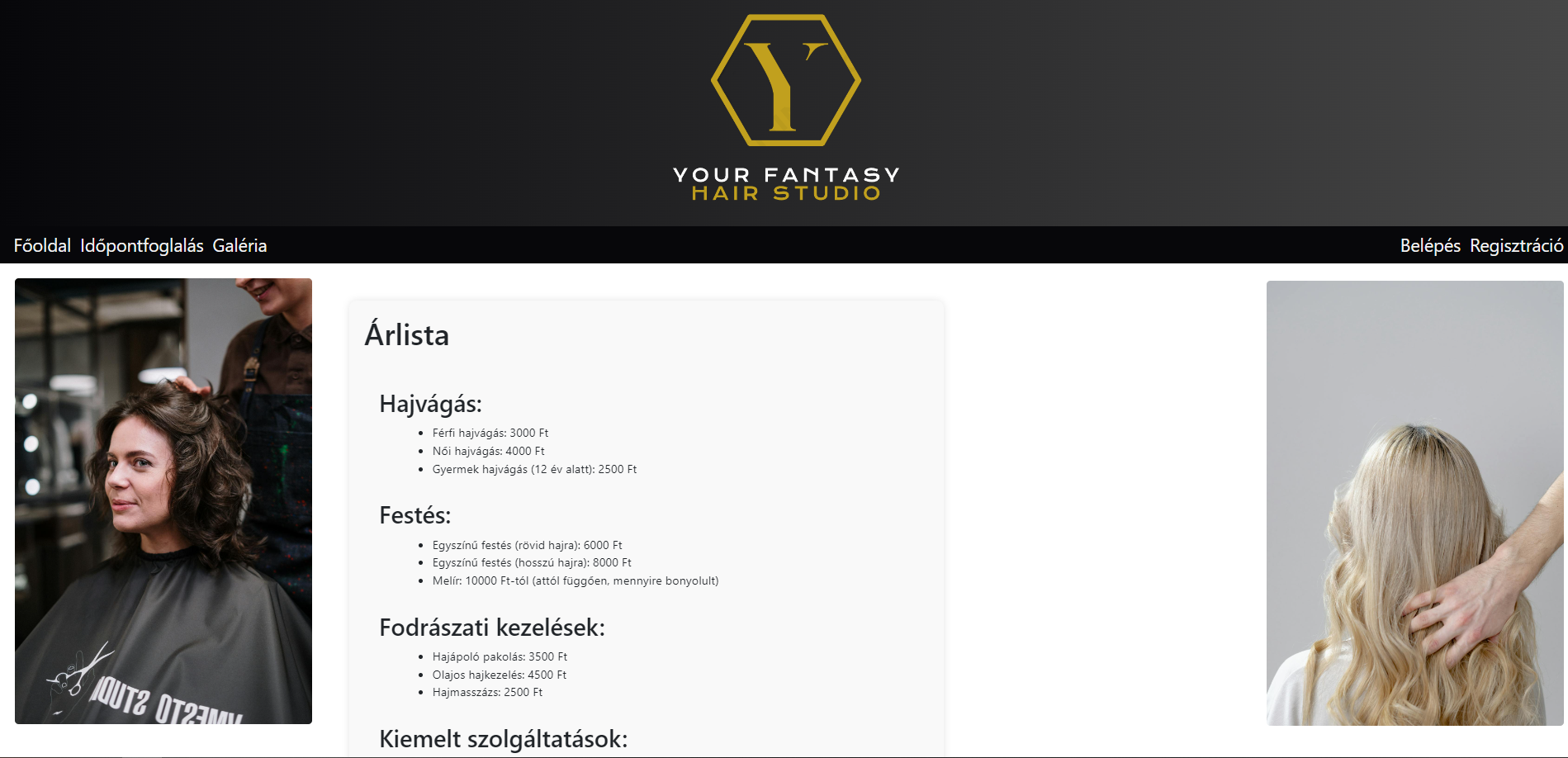
Ez az oldal a fodrászat eddigi munkáit tartalmazza. Mindenki számára elérhető ez az oldal, még bejelentkezés nélkül is megtekinthető. Ennek az oldalnak kiemelt feladata, hogy népszerűsítse a fodrászatot. Egy cég számár fontos a reklám, amely segítségével növelni lehet a kuncsaftok számát. Az itt található képeket a Copilot ai segítségével készítette el. Meglepően tapasztaltam, hogy már milyen minőségű képek elkészítésére képes a mesterséges intelligencia.



9.ábra: galéria oldal

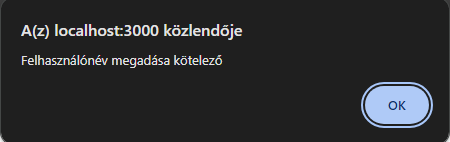
Áraink:

Ez az oldal a fodrászat ár listáját tartalmazza, a fodrászat számára fontos a felhasználók megfelelő tájékoztatása. Ez az oldal nincs regisztrációhoz kötve, mindenki számára elérhető. A főbb menüpontokat tartalmazza az időpontfoglaló oldal szolgáltatások listája.



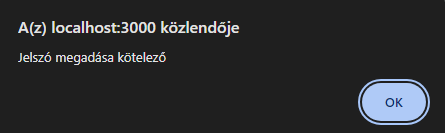
10. ábra: Áraink oldal

Lehetséges hibaüzenetek

**Nincs megadva felhasználónév**

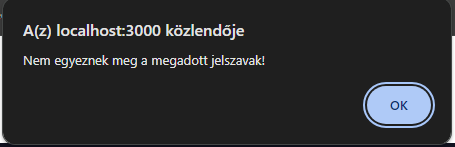
A regisztráció, bejelentkezés lebonyolítása esetén a rendszer jelzi a felhasználónak, ha nem adott meg felhasználónevet.

11. ábra: Felhasználónév nincs megadva

**Jelszó megadása kötelező**

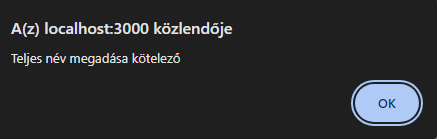
Ha regisztráció során nincs megadva mindkét jelszó akkor a rendszer figyemeztet erre.

12. ábra: Jelszó nincs megadva

**Jelszavak nem egyeznek meg**

Ha a regisztráció során nem egyezik a megadott két jelszó, akkor a rendszer erre figyelmeztet.

13. ábra: Megadott jelszavak nem egyeznek

**Teljes név megadása kötelező**

Ha regisztráció során nincs megadott teljes név, akkor a rendszer erre figyelmeztet.

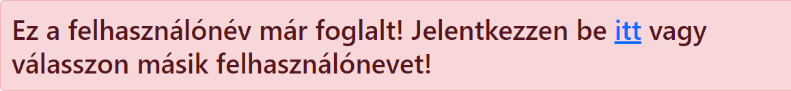
14. ábra: Teljesnév megadása kötelező

**Hiba a bejelentkezés során:**

Ha a bejelentkezéskor (pl. nem megfelelő felhasználónév vagy jelszó, nem található a felhasználó az adatbázisban) ezt a rendszer jelzi a felhasználó felé.

15. ábra: Hibás felhasználónév vagy jelszó

**Hiba a regisztráció során:**

Ha a regisztráció során bármilyen hiba lép fel (pl. már regisztrált felhasználó) ezt a rendszer jelzi a felhasználó felé.

16. ábra: Felhasználónév foglalt

Fejlesztői dokumentáció

Futási környezet:

Mivel a weboldal nem az internet egy szerverén fut, ezért egy saját szerveren kell elindítanunk a programot. Ehhez a XAMPP 8.0.30 verzióját kell letölteni majd telepíteni a saját gépünkre. Ha ez megtörtént, akkor a Visual Studio Code nevű programot kell telepítenünk. Ezután a fájlokat tartalmazó mappát kell megnyitni az előbb a Visual Studio Code-ban, és PHP Server-ként futtatni, hogy tökéletesen tudjon kommunikálni a szerverrel. Ahhoz, hogy a szerver megfelelően tudjon válaszolni a kérésekre, a phpMyAdmin felületen létre kell hozni az adatbázist. Ehhez csak az YFHS.sql nevű fájlt kell importálni a felületen, és már létre is jön az adatbázis automatikusan.

Programnyelvek:

**HTML5:**

Az HTML5 a webfejlesztés legújabb szabványa, amely számos új funkciót és elemet vezetett be a weboldalak fejlesztésének támogatására. Ennek részeként megjelentek új strukturális elemek, mint például a fejléc, navigációs sáv, szakaszok, cikkek, mellettük pedig lehetőség van audio- és videótartalmak beágyazására anélkül, hogy külső pluginokra lenne szükség. A HTML5 bevezette a canvas elemet, amely lehetővé teszi a dinamikus grafikus tartalmak létrehozását JavaScript segítségével. Emellett az HTML5 új lehetőségeket kínál az adattárolásra is, mint például a sessionStorage és localStorage, továbbá adatbáziskezelést is biztosít a böngészőben, lehetővé téve a strukturált adatok tárolását és kezelését a kliens oldalon.

**CSS:**

A Cascading Style Sheets (CSS) egy stílusleíró nyelv, amelyet weboldalak formázására és elrendezésének meghatározására használnak. CSS segítségével lehetőség van megadni a különböző elemek megjelenését, például színeket, betűtípusokat, elhelyezkedést és méreteket. A stíluslapokat külön fájlokban vagy közvetlenül a HTML kódban is lehet definiálni, így lehetőség van a stílusok újrafelhasználására és a kód karbantarthatóságának növelésére. Emellett a CSS alkalmazkodik a különböző eszközök képernyőméreteihez, lehetővé téve a reszponzív tervezést és a felhasználóbarát megjelenést minden eszközön. Továbbá, CSS lehetővé teszi animációk és átmenetek létrehozását, amelyekkel fokozható a weboldalak interaktivitása és vonzereje a látogatók számára.

**JavaScript:**

JavaScript egy programozási nyelv, melyet webfejlesztés során alkalmaznak a weboldalak interaktivitásának és dinamizmusának biztosítására. Ez a nyelv lehetővé teszi a felhasználókkal való aktív kommunikációt, például űrlapok ellenőrzését, események kezelését, valamint animációk és effektek létrehozását. A JavaScript kódot a böngésző futtatja, ami azt jelenti, hogy a böngészőkben végrehajtott kód közvetlenül az ügyfél eszközén fut. Ezen kívül a JavaScript lehetőséget biztosít a dokumentum objektum modell (DOM) manipulációjára, ami által a weboldal tartalma és megjelenése dinamikusan módosítható. A nyelv aszinkron működést is támogat, így a böngésző nem blokkolódik, lehetővé téve a felhasználói interakciók folyamatos kezelését. JavaScript továbbá rendelkezik számos beépített és külső könyvtárral elérhető funkcionalitással, amelyek segítségével fejlett feladatokat is megoldhatunk, mint például adatkezelés vagy aszinkron kommunikáció a szerverrel.

**PHP:**

PHP (Hypertext Preprocessor) egy szerveroldali szkriptnyelv, melyet gyakran használnak webfejlesztés során. A PHP kódot a szerver értelmezi és futtatja, így lehetőséget biztosít a dinamikus tartalmak generálására és a weboldalak interaktivitásának biztosítására. PHP segítségével könnyedén kommunikálhatunk adatbázisokkal, feldolgozhatjuk űrlapokat és kezelhetünk felhasználói munkameneteket. Ezen kívül PHP rendelkezik számos beépített funkcióval és kiterjesztéssel, amelyek lehetővé teszik például fájlkezelést, e-mail küldést vagy képfeldolgozást. PHP kiváló választás a dinamikus weboldalak létrehozásához, mivel könnyen tanulható, rugalmas és széles körben támogatott a webfejlesztői közösség által.

**SQL:**

SQL (Structured Query Language) egy standard adatbáziskezelési nyelv, amelyet az adatbázisok kezelésére és lekérdezésére használnak. SQL segítségével lehetőség van adatok lekérdezésére, beszúrására, módosítására és törlésére az adatbázisokban. A nyelv számos parancsot tartalmaz, például a SELECT-et lekérdezések végrehajtására, az INSERT-et adatok beszúrására, az UPDATE-et adatok módosítására és a DELETE-et adatok törlésére. SQL lehetővé teszi az adatok strukturális és tartalmi manipulációját, valamint az adatok közötti kapcsolatok kezelését, például a JOIN művelet segítségével. Az SQL egy alapvető eszköz az adatok hatékony kezeléséhez és lekérdezéséhez az adatbázisokban, és elengedhetetlen a modern adatbázis-alapú alkalmazások fejlesztése során.

Követelmények (funkcionális):

**Főoldal megjelenésekor:**

* Menüpontok közötti váltás
* Oldal tartalmának betöltődése

**Hatékonyság:**

* A szerver nagy felhasználói igény esetén lassulhat.
* A megjelenített képek kis méretűek és tömörítettek legyenek.
* Gyors válaszidő
* Pontos visszajelzés

**Megbízhatóság:**

* Rendeltesésszerű használat esetén nem lép fel hiba
* A fellépő hibákat jól kezeli, a felhasználót tájékoztatja a hibáról a program

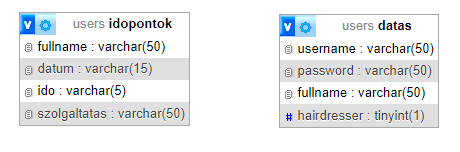
Követelmények (nem funkcionális):

**Biztonság:**

* Biztonságos adatkommunikáció a szerverrel

**Felhasználhatóság:**

* Jól elkülöníthető színek használata
* Egyszerű, átlátható felhasználói felület

Adatbázis felépítése

17. ábra: Adatbázis táblái



18. ábra: Adatbázis táblái 2

Táblák:

**1. Felhasználók tábla (datas)**

Ez a tábla a felhasználók tárolására szolgál. Minden felhasználó saját azonosítóval rendelkezik. A felhasználónév, jelszó, teljes név, és egy logikai változó, ami azt jelzi, hogy az adott személy fodrász-e vagy nem.



19. ábra: Datas tábla szerkezete

20. ábra: Datas tábla példa adatok

**2. Időpontok tábla (idopontok)**

Ez a tábla a foglalt időpontok tárolására szolgál. A teljes név, dátum, idő és szolgáltatás van benne tárolva.



21. ábra: Idopontok tábla szerkezete



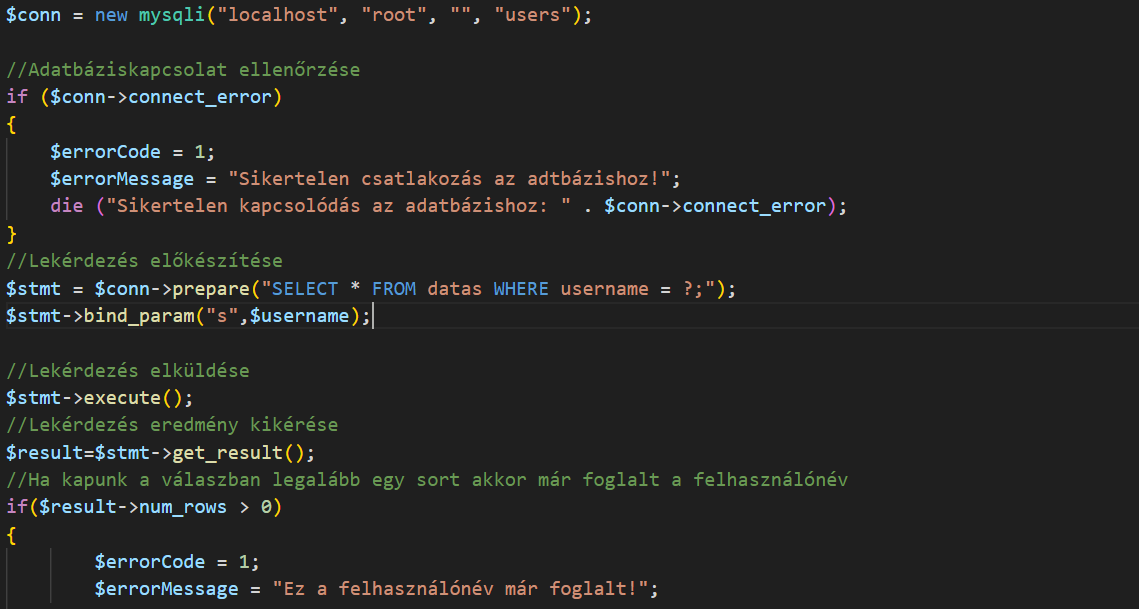
22. ábra: Idopontok tábla példa idopontokkal

Programkód megjelenítése

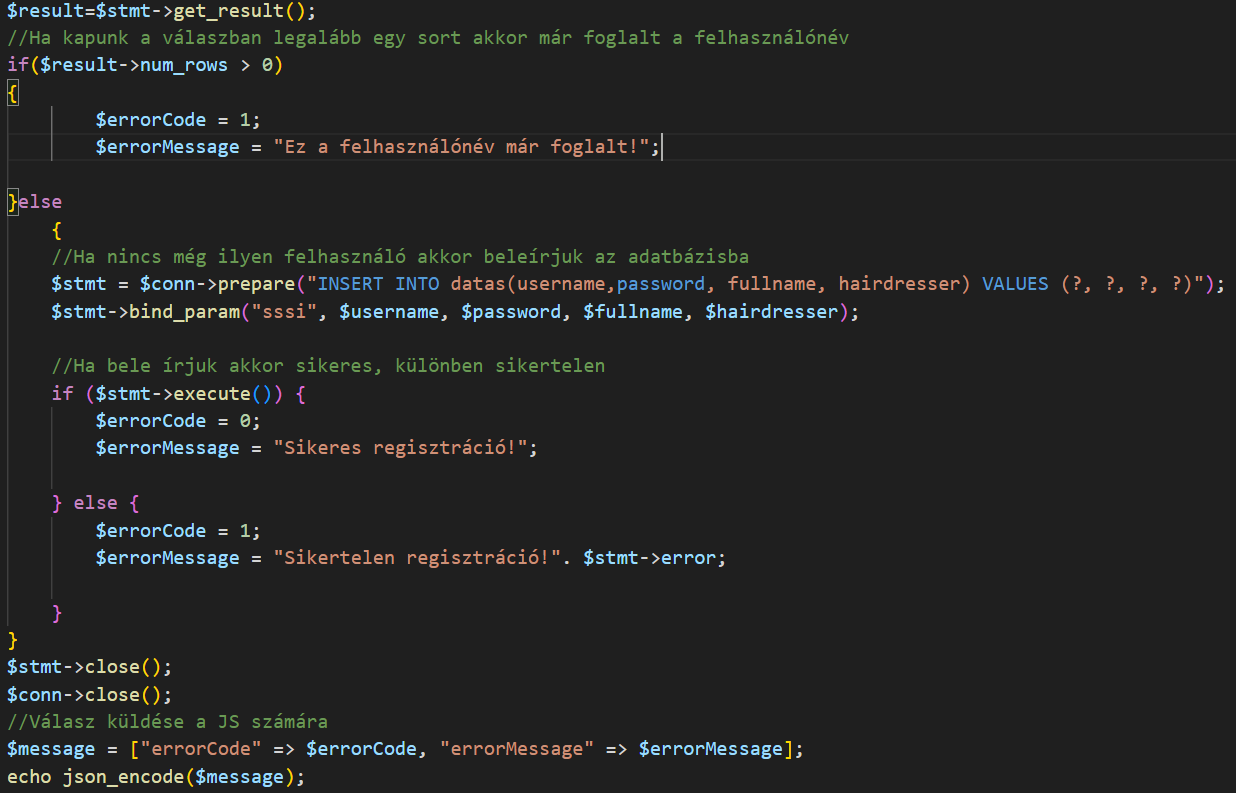
**Session:**

Session, vagy munkamenet, egy olyan folyamat az internetes alkalmazásokban, amely lehetővé teszi az adatok tárolását az egyes felhasználói munkamenetek során. A munkamenetek létrehoznak egy ideiglenes tároló teret a szerveren, amelyben a felhasználóhoz kapcsolódó adatokat tárolhatják, például bejelentkezési információkat vagy kosárba helyezett elemeket. Az alkalmazások általában egyedi azonosítóval látják el a munkameneteket, amelyeket a felhasználó azonosítására használhatnak a későbbi kérelmek során. A munkamenetek jelentősége abban rejlik, hogy lehetővé teszik az alkalmazásoknak, hogy tartsák a felhasználók állapotát és tevékenységeit azáltal, hogy megőrzik az adatokat a felhasználó böngészése során. Fontos azonban ügyelni a biztonságra, mivel a munkamenetek gyakran tartalmaznak érzékeny információkat, ezért a megfelelő titkosítás és védelem elengedhetetlen.

**Regisztráció**

Regisztráció során, mikor a felhasználó beírja az adatait akkor a gomb lenyomása után indul egy kérés az adatbázis felé, amiben lekérdezzük, hogy szabad-e még az adott felhasználónév. A csatlakozáshoz mysqli függvényt használok, ami véd a SQL Injection ellen. Ez egy támadási forma, ami során a támadó egy rosszindulatú kódot juttat az adatbázisba, ezzel manipulálva, vagy elérhetetlenné téve az adatbázis tartalmát.

23. ábra: Adatbázis csatlakozás

Ha nem található meg az adatbázisban még az adott felhasználónév, akkor megkezdődik a regisztráció folyamata. Előlészítünk egy SQL adatfeltöltést a datas tábla rekordjaihoz, és a tartalma a felhasználó által megadott felhasználónév, jelszó, teljes név és a rangja lesz bele írva az adatbázisba. Ha sikeresen belekerültek az adatok az adatbázisba, akkor a regisztráció sikeresen történt meg, különben kiírja a hibakódot a felhasználó számára.

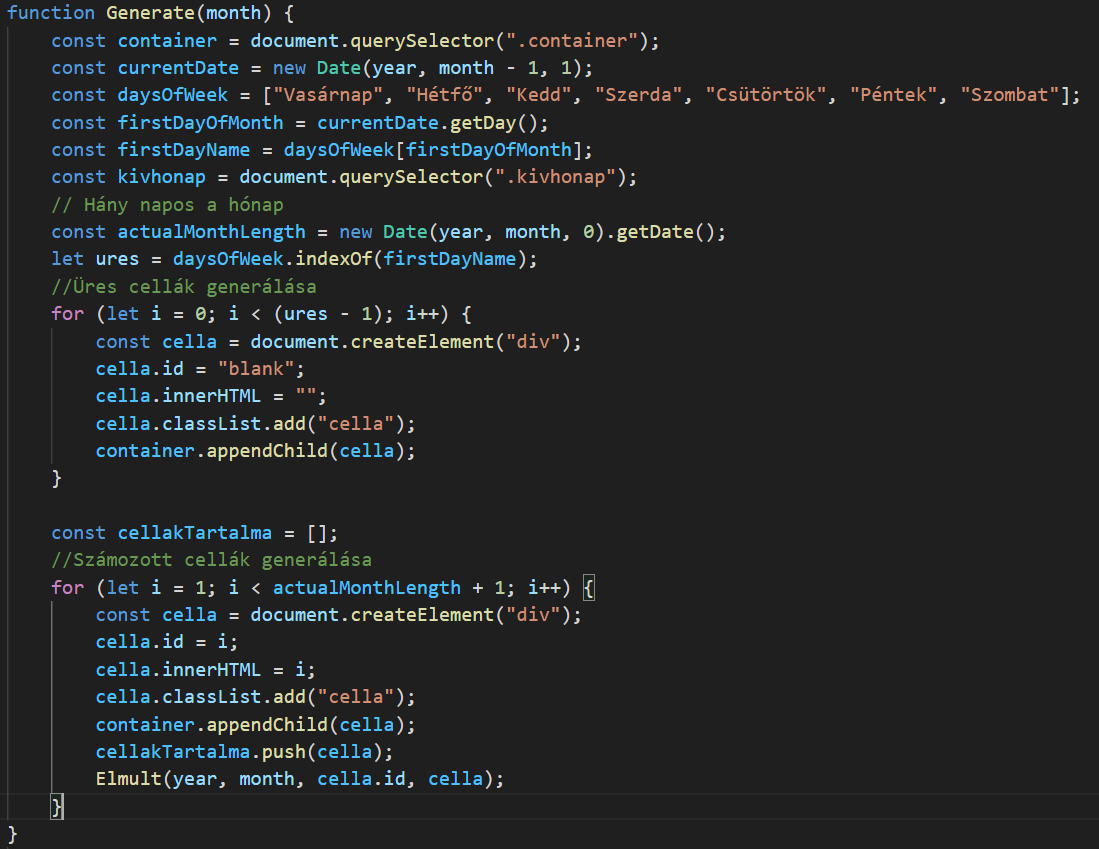
24. ábra: Adatfeltöltése az adatbázisba

**Bejelentkezés**

25. ábra: Bejelentkezés

Amikor a felhasználó sikeresen bejelentkezik akkor a felhasználóneve, teljesneve és a rangja belekerül egy a SESSION munkamenetbe. Ez a folyamat segít a felhasználó azonosításában, és hogy akár minden PHP fájlban tudjuk, hogy ki az adott felhasználó.

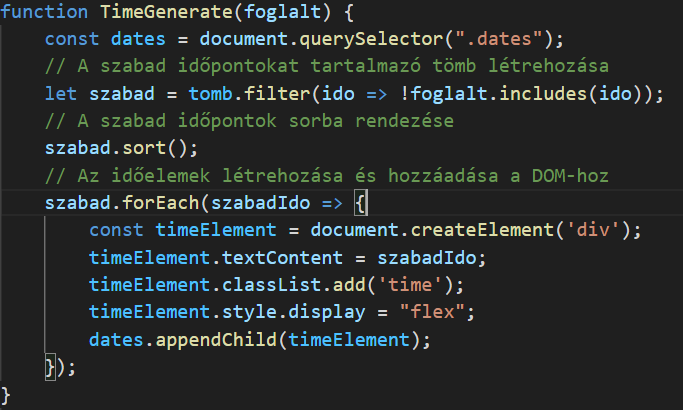
**Időpontfoglalás**

Az időpontfoglaló oldalamon a naptár dinamikusan generálódik. Erre azért van szükség, hogy biztosan csak annyi nap legyen generálva, ahány napból áll a hónap. A JavaScript Date függvényét használtam a munkám során. Lekértem a hónap hosszát, hogy tudjam, az adott hónap hány napból áll, majd azt is, hogy milyen nap 1-je az adott hónapban. A generálás az üres cellák generálásával kezdődik, majd 1-jétől a hónap hosszáig beszámozva generálja le a program a dátumokat. Az elmúlt napokat szürkés háttérrel jelöli, és ezekre a napokra már nem is tud foglalni a felhasználó. Az aktuális napot aranyszínű kör alakú jelzés szimbolizálja.

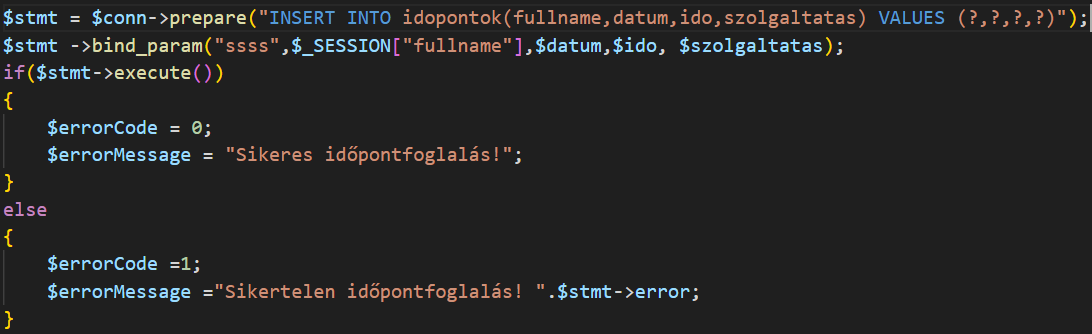
26. ábra: Naptár generálása

A következő programrészlet egy fetch kéréssel lekéri a már foglalt időpontokat az adatbázisból. Egy másik tömbben tárolja a nyitvatartási időt 30 percentként, ezen tömb elemein végig lépkedve eldönti, hogy melyik foglalt már és ezt tárolja és átadja a TimeGenerate function számára.

27. ábra: Foglalt időpontok lekérése

A TimeGenerate function a megkapott adatok alapján a szabad időpontot a filter metódus segítségével kiválasztja a szabad időpontot majd tárolja azokat egy tömbben. Az eltárolt adok alapján készül el a szabad időpontokat tartalmazó legördülő lista.

28. ábra: Szabad időpont

Az időpontok adatbázisba felhasználó teljes neve, a dátum, idő és az előzőleg kiválasztott szolgáltatás kerül eltárolásra. Ha beírásra kerültek az adtok, akkor sikeresen megtörtént a foglalás.

29. ábra: Időpontfoglalás

**Időpontok**

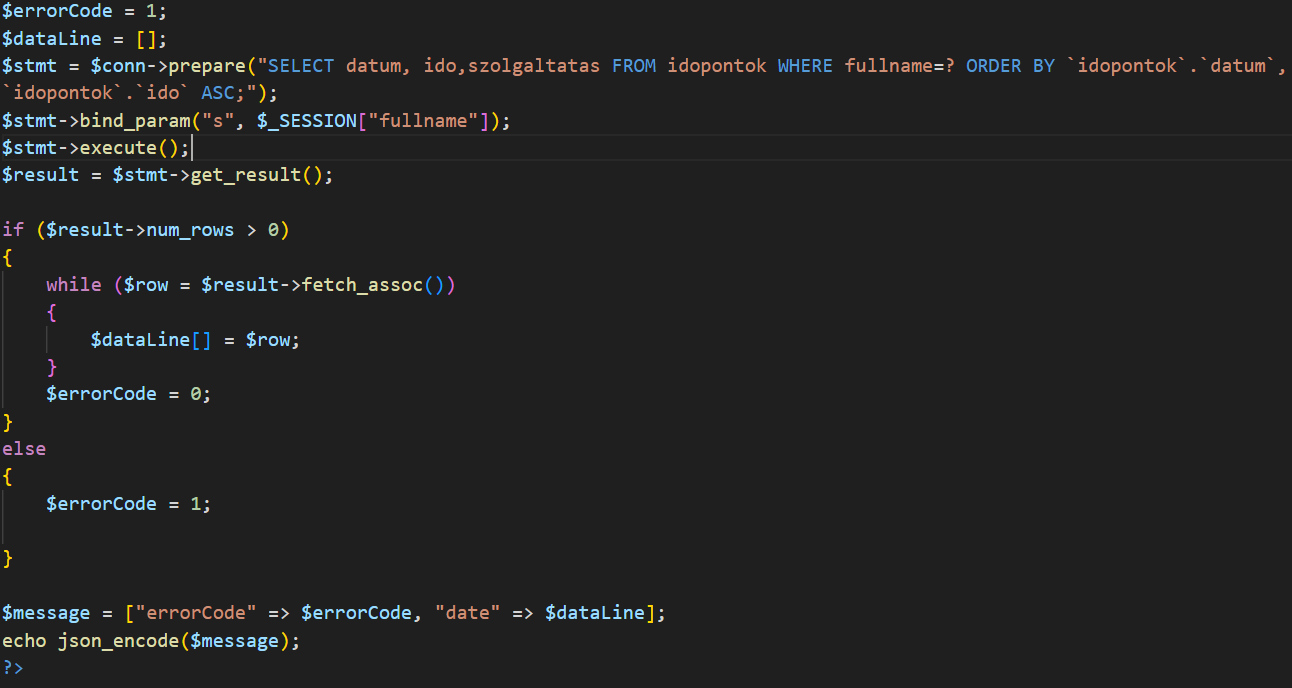
Időpontok oldalon, amelyet csak a fodrász lát, itt minden adatot kiolvasunk az idopontok adatbázisból, majd ezt dinamikusan meg is jeleníti az oldalon.



30. ábra: Fodrász időpntjainak lekérése

**Időpontjaim**

Ezen az oldalon a felhasználó csak a saját időpontjait tekintheti meg. Itt az adatok kikérése az adatbázisból a session-ben lévő teljesnév alapján történik.



31. ábra: Felhasználó időpntjainak lekérése

Fejlesztői környezet

**A fejlesztéshez használt számítógép:**

Alaplap neve Gigabyte GA-A320M-H

CPU típusa AMD Ryzen 5 1400 Quad-Core Processor 3.20 GHz

Memória típusa 8GB DDR4 2400MHz RAM

Videókártya típusa Nvidia GTX 1050 Ti (4GB)

**Szoftverek:**

Windows 10 Home, 22H2

Visual Studio Code ver.1.88.0

GIMP 2.10.36

Készítő

Kis Gábor

Szoftvertervező

Email: gabork706@gmail.com

Tel: +36 70/280-2081

Fejlesztési lehetőségek

**Regisztráció megerősítése emailben**

A regisztráció során meg kell adni egy email- címet, amire küld ezután egy email-t és itt meg kell majd erősíteni, hogy ő regisztrált, és jóvá hagyja azt. Ha ez megtörtént, csak akkor tud majd belépni a felhasználó.

**Email értesítés az időpont esedékességére**

A felhasználó által megadott idővel előbb a rendszer küld egy email-t, hogy például 3 nap múlva 14:00- kor időpontja van a fodrászathoz.

**Időpont lemondása**

Ezzel a funkcióval lehetővé tenném, hogy a felhasználóknak, ha valami közbe jön és nem tud elmenni az foglalt időpontra, akkor egy gomb nyomására le tudja mondani.

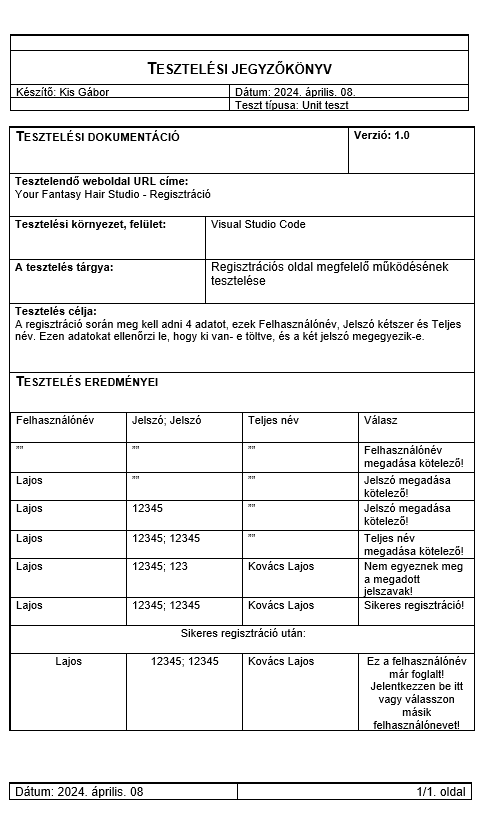
**Jelszó megváltoztatás**

Jelenleg a megadott jelszó megváltoztatására nincs lehetőség ebből adódóan, ha a felhasználó elfelejti a jelszavát, akkor nem tudja megváltoztatni. Csak úgy tud időpontot foglalni, ha készít egy másik felhasználói fiókot.

**A felhasználók értékelhetik a fodrászatot**

Ezzel az új funkcióval lehetővé tenném, hogy az oldalon a felhasználók láthassák egymás véleményét és értékelését. Ezzel akár elősegítik a fodrászat népszerűsítését is.

Tesztelési dokumentáció



32. ábra: Tesztelési jegyzőkönyv

Tesztelés összegzése

A tesztelést többféle módon történhet például statikus vagy dinamikus módon. Statikus tesztelés során a kódot magát vizsgáljuk, például ellenőrizzük, hogy minden sor végén van-e pontosvessző, ez a szintaktikai elemzés része. Emellett figyelhetünk a szemantikára is, ami azt jelenti, hogy logikailag helyesen működik-e a programkód.

A dinamikus tesztelés során futtatjuk a programkódot. Ha rendelkezésre áll a forráskód, akkor fehérdobozos tesztelésről beszélünk, ahol a fejlesztők átnézik és tesztelik a ciklusokat, elágazásokat, hogy megfelelően működjenek.

A fekete dobozos tesztelés során nem nézzük a forráskódot, hanem a specifikáció alapján teszteseteket állítunk fel, és ezeket használjuk a kimenetek vizsgálatához. Ilyenkor például szélsőértékeket is vizsgálunk.

Én a dinamikus tesztelésen belül végeztem fehérdobozos tesztelést a programkódokon, valamint teszteseteket írtam és futtattam le. Emellett több nézőpontból is elfogadási tesztet hajtottam végre az ismerőseim körében. Egyikük sem fejlesztő, de mégis észrevett néhány hibát, amit sikeresen javítottam. Örömmel fogadtam a pozitív visszajelzéseket, mivel rámutattak olyan lehetőségekre, amelyek még nem voltak számukra elérhetőek.

Fejlesztés után:

A projekt végső tesztelése során dinamikus fehérdobozos tesztelést végeztem. Ekkorra már rendelkezésre állt minden forrásfájl működő formában, így lehetővé vált a teljes kód alapos tesztelése és átvizsgálása.

Felhasznált irodalom, eszközök:

Felhasznált irodalom

<https://www.w3schools.com/>

<https://stackoverflow.com/>

Felhasznált eszközök:

**Képek keresése:**

<https://www.pexels.com/hu-hu/>

**Oldal logójának tervezése:**

<https://smashinglogo.com/hu/>

**Login icon:**

https://hu.pinterest.com/pin/785807834990247059/

**Kijelentkezés icon:**

https://www.flaticon.com/free-icon/logout\_1828427?term=logout&page=1&position=1&origin=tag&related\_id=1828427

**Képek generálása:**

https://www.bing.com/images/create?FORM=GENILP

Ábrajegyzék

[1. ábra kezdőlap design terv 9](#_Toc164111054)

[2. ábra kezdőlap végleges kinézet 9](#_Toc164111055)

[3. ábra Főoldal felhasználói felülete 11](#_Toc164111056)

[4. ábra Regisztráció oldal 12](#_Toc164111057)

[5**.** ábra Bejelentkezés ablak 13](#_Toc164111058)

[6**.** ábra: Időponfoglalás oldal 14](#_Toc164111059)

[7**.** ábra: Időpontok oldal 15](#_Toc164111060)

[8.ábra: Időpontjaim oldal 16](#_Toc164111061)

[9.ábra: galéria oldal 17](#_Toc164111062)

[10. ábra: Áraink oldal 18](#_Toc164111063)

[11. ábra: Felhasználónév nincs megadva 19](#_Toc164111064)

[12. ábra: Jelszó nincs megadva 19](#_Toc164111065)

[13. ábra: Megadott jelszavak nem egyeznek 19](#_Toc164111066)

[14. ábra: Teljesnév megadása kötelező 19](#_Toc164111067)

[15. ábra: Hibás felhasználónév vagy jelszó 20](#_Toc164111068)

[16. ábra: Felhasználónév foglalt 20](#_Toc164111069)

[17. ábra: Adatbázis táblái 25](#_Toc164111070)

[18. ábra: Adatbázis táblái 2 25](#_Toc164111071)

[19. ábra: Datas tábla szerkezete 25](#_Toc164111072)

[20. ábra: Datas tábla példa adatok 25](#_Toc164111073)

[21. ábra: Idopontok tábla szerkezete 26](#_Toc164111074)

[22. ábra: Idopontok tábla példa idopontokkal 26](#_Toc164111075)

[23. ábra: Adatbázis csatlakozás 28](#_Toc164111076)

[24. ábra: Adatfeltöltése az adatbázisba 28](#_Toc164111077)

[25. ábra: Bejelentkezés 29](#_Toc164111078)

[26. ábra: Naptár generálása 30](#_Toc164111079)

[27. ábra: Foglalt időpontok lekérése 31](#_Toc164111080)

[28. ábra: Szabad időpont 32](#_Toc164111081)

[29. ábra: Időpontfoglalás 32](#_Toc164111082)

[30. ábra: Fodrász időpntjainak lekérése 33](#_Toc164111083)

[31. ábra: Felhasználó időpntjainak lekérése 33](#_Toc164111084)

[32. ábra: Tesztelési jegyzőkönyv 36](#_Toc164111085)

Összegzés

Összegezve a kitűzött feladatot, amit előre elképzeltem sikerült elkészíteni. Van még néhány ötletem, amit a fejlesztés közben találtam már ki. Nagyon élveztem a feladat elkészítését, mert ezzel nagyon sokat tanultam, és még meg is szerettette velem a webprogramozást. Néha egy kicsit már sok volt, amikor már 5 órája egy problémát kerestem, és annyi volt a hiba, hogy egy idézőjel rossz helyen volt, de összességében nem bántam meg, hogy ezt választottam.

Köszönetnyilvánítás

Ebben az évben nagyon sok új dolgot tanulhattam meg elméleti és gyakorlati téren is. A megtervezés és a projekt megírása folyamata során is nagyon sok ötletet kaptam tanáraimtól és diáktársaimtól is egyaránt. A projekt elkészítése során szerintem nagyon jól összekovácsolódott az osztály is, ugyanis követtük egymás munkáinak alakulását, és mindig tudtunk egymásnak valami új módszert mutatni egy esetleges hiba megoldására. Ezennel szeretném megköszönni Hammer Péter tanárúr munkáját, ugyanis nem számított neki, hogy mikor kerestem meg egy problémával, mindig nagyon szívesen segített.  
A képek szerkesztése során Gimp-et használtam, amire Tüskéné Dombi Zsuzsanna tanárnő tanított meg, az ő munkáját is nagyon szépen köszönöm.

Plágium nyilatkozat

**Nyilatkozat a záródolgozat készítésére vonatkozó szabályok betartásáról**

Alulírott: Kis Gábor jelen nyilatkozat aláírásával

kijelentem, hogy a

**Your Fantasy Hair Studio**

című záródolgozatönálló munkám, a záródolgozatkészítése során betartottam a szerzői jogról szóló 1999. évi LXXVI. törvény vonatkozó rendelkezéseit, valamint az intézmény által előírt, a záródolgozatkészítésére vonatkozó szabályokat, különös tekintettel a hivatkozások és idézetek tekintetében.

Jelen nyilatkozat aláírásával tudomásul veszem, hogy amennyiben bizonyítható, hogy a záródolgozatotnem magam készítettem, vagy a záródolgozattalkapcsolatban szerzői jogsértés ténye merül fel, az intézmény megtagadja a záródolgozatbefogadását és ellenem fegyelmi eljárást indíthat.

A záródolgozatbefogadásának megtagadása és a fegyelmi eljárás indítása nem érinti a szerzői jogsértés miatti egyéb (polgári jogi, szabálysértési jogi, büntetőjogi) jogkövetkezményeket.

Gyöngyös, 2024.04.07.

vizsgázó aláírása