

VÁCI SZAKKÉPZÉSI CENTRUM
BORONKAY GYÖRGY
MŰSZAKI TECHNIKUM ÉS GIMNÁZIUM

VIZSGAREMEK

Fedor Benjámín

Kertész Kornél

Turza Norbert

2024.

VÁCI SZAKKÉPZÉSI CENTRUM
BORONKAY GYÖRGY
MŰSZAKI TECHNIKUM ÉS GIMNÁZIUM



VIZSGAREMEK

Frissfutár

Konzulens:

Készítette: Fedor Benjámin

Kertész Kornél

Turza Norbert

Hallgatói nyilatkozat

Alulírottak, ezúton kijelentjük, hogy a vizsgaremek saját, önálló munkánk, és korábban még sehol nem került publikálásra.

Fedor Benjámin

Turza Norbert

Kertész Kornél

Konzultációs lap

Vizsgázók neve: Fedor Benjámín - Turza Norbert - Kertész Kornél

Vizsgaremek címe: Frissfutár

Program nyújtotta szolgáltatások:

- regisztráció
- bejelentkezés
- kijelentkezés
- rendelés
- szűrés ételfajták alapján
- adminisztrátori funkciók

Sorszám	A konzultáció időpontja	A konzulens aláírása
1.	2023.10.17.	
2.	2023.11.21.	
3.	2023.12.13.	
4.	2024.01.16	
5.	2024.02.20.	
6.	2024.03.12	

A vizsgaremek beadható:

Vác, 2024.

.....

Konzulens

A vizsgaremek átvettem:

Vác, 2024.

.....

A szakképzést folytató
intézmény felelőse

Tartalomjegyzék

Hallgatói nyilatkozat	3
Konzultációs lap	4
Tartalomjegyzék	5
Témaválasztás	7
1. Fejlesztői dokumentáció	8
1.1 Ismertetés	8
1.2 Fejlesztői környezet, Nyelvek és Technológiák	8
1.2.1 Visual Studio Code	8
1.2.2 Visual Studio 2022	8
1.2.3 PHP	8
1.2.4 CSS	8
1.2.5 C#	9
1.2.6 Javascript	9
1.2.7 HTML	9
1.3 Adatszerkezet	9
1.3.1 Adattáblák	10
1.3.2 Táblák közötti kapcsolatok	11
1.3.3 A „telepulesek” tábla	12
1.3.4 A „varmegyek” tábla	13
1.3.5 A „szallitasi_cim” tábla	14
1.3.6 A „felhasznalo_adatok” tábla	15
1.3.7 A „rendeles” tábla	16
1.3.8 A „fizetesi_modok” tábla	18
1.3.9 A „rendeles_statusza” tábla	19
1.3.10 A „rendeles_tetelei” tábla	19
1.3.11 A „termek” tábla	21
1.3.12 Az „afa_kulcs” tábla	22
1.3.13 A „mennyisegi_egysegek” tábla	23
1.3.14 A „termek_csoport” tábla	24
1.3.15 A „termek_kategoria” tábla	25
1.3.16 A „termek_csomag_reszletei” tábla	26
1.3.17 A „termek_csomag” tábla	27
1.3.18 Az „admin” tábla	28
1.4 Használati esetmodell, szerepkörök	29

1.5	Modulok ismertetése	30
1.5.1	Webes alkalmazás	30
1.5.2	Asztali alkalmazás	35
1.6	Tesztelések.....	39
1.6.1	Reszponzivitás.....	39
1.6.2	Téma váltás.....	40
1.6.3	Adatok kezelése.....	41
1.6.4	Adatbázis	42
1.6.5	Admin applikáció tesztelése	44
1.7	Továbbfejlesztési lehetőségek.....	48
1.7.1	Keresés funkció megvalósítása	48
1.7.2	Fizetési módok kezelése	48
1.7.3	Rendelés visszaigazolás, email küldése	48
1.7.4	Rendelések megtekintése külön webes felületen (admin)	48
1.7.5	Adatbázis szerveren történő működtetése	48
1.7.6	Bug-ok javítása, optimalizálás, karbantartás.....	48
1.7.7	Kiemelt ajánlatok kezelése	48
1.7.8	Admin alkalmazás optimalizálása	48
2	Felhasználói dokumentáció.....	49
2.1	Ismertetés	49
2.2	Rendszerkövetelmények.....	49
2.3	Alkalmazás telepítése, beállítása	49
2.4	Alkalmazás használata	50
2.4.1	Webes alkalmazás használata	50
2.4.2	Asztali alkalmazás használata	54
3	Irodalomjegyzék	55
1.5	Internetes források	55
2.	Mellékletek.....	56
2.1	Forrásállományok.....	56
2.2	Adatbázis.....	56
2.3	Xampp	56
2.4	Dokumentáció.....	56
2.5	Teszteléshez szükséges adatok	56

Témaválasztás

A mesterremekünk témájául egy rendkívül egyszerű, de sokrétű és felhasználó barát online étel rendelési oldalt készítettünk.

A platformunk célja, hogy megkönnyítse az emberek számára az ételrendelést és a házhozszállítást, minimalizálva ezzel az adminisztratív terheket és a várakozási időket.

Az oldalunk intuitív felülete és felhasználóbarát tervezése lehetővé teszi, hogy a felhasználók könnyen böngézhessék az éttermek és ételek széles választékát, és egyszerűen kiválaszthassák a számukra legmegfelelőbb lehetőségeket.

A rendszerünk rugalmas és testre szabható, így mindenki megtalálhatja az igényeinek megfelelő lehetőséget, legyen szó diétás étkezésről, allergiák figyelembevételéről vagy akár különleges ételkérésekről.

A célunk, hogy a felhasználók számára kényelmes és problémamentes élményt nyújtsunk az ételrendelésben, és hozzájáruljunk a mindennapi életük egyszerűsítéséhez és élvezetesebbé tételéhez.

1. Fejlesztői dokumentáció

1.1 Ismertetés

A Frissfutár egy étel - és italrendelésre használatos reszponzív weboldal (PC-n, tableten, telefonon egyaránt kényelmesen használható), asztali admin alkalmazással, ami teljeskörű karbantartást biztosít.

1.2 Fejlesztői környezet, Nyelvek és Technológiák

1.2.1 Visual Studio Code

A Visual Studio Code egy ingyenes, nyílt forráskódú fejlesztői környezet, amelyet a Microsoft fejlesztett ki. Számos programozási nyelvet támogat, és rendelkezik számos funkcióval, amelyek segítenek a fejlesztőknek hatékonyabban dolgozni. Ezek közé tartozik a kódszerkesztő, a beépített Git integráció, kiterjesztési lehetőségek, hibakeresési eszközök, és sok más.

1.2.2 Visual Studio 2022

A Visual Studio 2022 egy integrált fejlesztői környezet (IDE) a Microsofttól, amely kifejezetten a fejlesztőknek készült. Ez a szoftvercsomag számos programozási nyelvet és platformot támogat, például C#, C++, JavaScript, Python és még sok más. A Visual Studio 2022 számos fejlett funkcióval rendelkezik, például kódszerkesztő, hibakeresés, verziókezelés (pl. Git), valamint kiterjesztési lehetőségek, amelyek segítségével testreszabhatod a fejlesztői környezetet az igényeidnek megfelelően.

1.2.3 PHP

A PHP egy szkriptnyelv, amely kifejezetten webfejlesztésre lett tervezve. A rövidítés "PHP: Hypertext Preprocessor"-ra utal, ami egy visszafelé rekuzív akronym. Dinamikus, szerveroldali nyelvként működik, ami azt jelenti, hogy a PHP kód a webserveren fut, és generálja a dinamikus weboldalak tartalmát.

1.2.4 CSS

A CSS (Cascading Style Sheets) egy nyelv, amelyet a weboldalak stílusának meghatározására használnak. A CSS segítségével formázhatod a weboldalak HTML elemeit, például megadhatod a színeket, a betűtípusokat, a méreteket, a térközöket és az

elrendezést. A CSS stílusok elválasztják a tartalmat (HTML) a megjelenéstől, ami lehetővé teszi a jobb karbantarthatóságot és a rugalmasabb tervezést. A CSS-t a böngészők értelmezik és alkalmazzák, így a weboldalak könnyen testreszabhatók, hogy igényeidnek és tervezési elképzeléseidnek megfeleljenek.

1.2.5 C#

A C# (C-sharp) egy erőteljes, modern programozási nyelv, amelyet a Microsoft fejlesztett ki. A C# a .NET platform részeként szolgál, és széles körű alkalmazások fejlesztésére alkalmas, beleértve a Windows asztali alkalmazásokat, webalkalmazásokat, mobilalkalmazásokat és játékokat is.

1.2.6 Javascript

A JavaScript egy dinamikus szkriptnyelv, amelyet elsősorban webfejlesztésre használnak. Gyakran használják interaktív webes alkalmazások és weboldalak fejlesztésére, amelyeknek célja a felhasználói élmény javítása és a weboldalak dinamizálása.

Lehetővé teszi a weboldalak elemzését, módosítását és azok viselkedésének kezelését a böngészőben futtatva. Ezáltal interaktív funkciókat, animációkat, űrlapok ellenőrzését, adatkezelést és még sok másat valósíthatunk meg.

1.2.7 HTML

A HTML (HyperText Markup Language) az internetes tartalom alapvető építőeleme, amely lehetővé teszi a strukturált szöveg megjelenítését böngészők segítségével. A HTML elemek, attribútumok és tartalom széles választékát kínálja, amelyek lehetővé teszik a tartalom testreszabását és a weboldalak funkcióinak bővítését. A HTML folyamatosan fejlődik, és újabb és újabb lehetőségeket kínál a webfejlesztőknek a weboldalak interaktívabbá és vonzóbbá tételéhez.

1.3 Adatszerkezet

Adatbázis neve: frissfutar

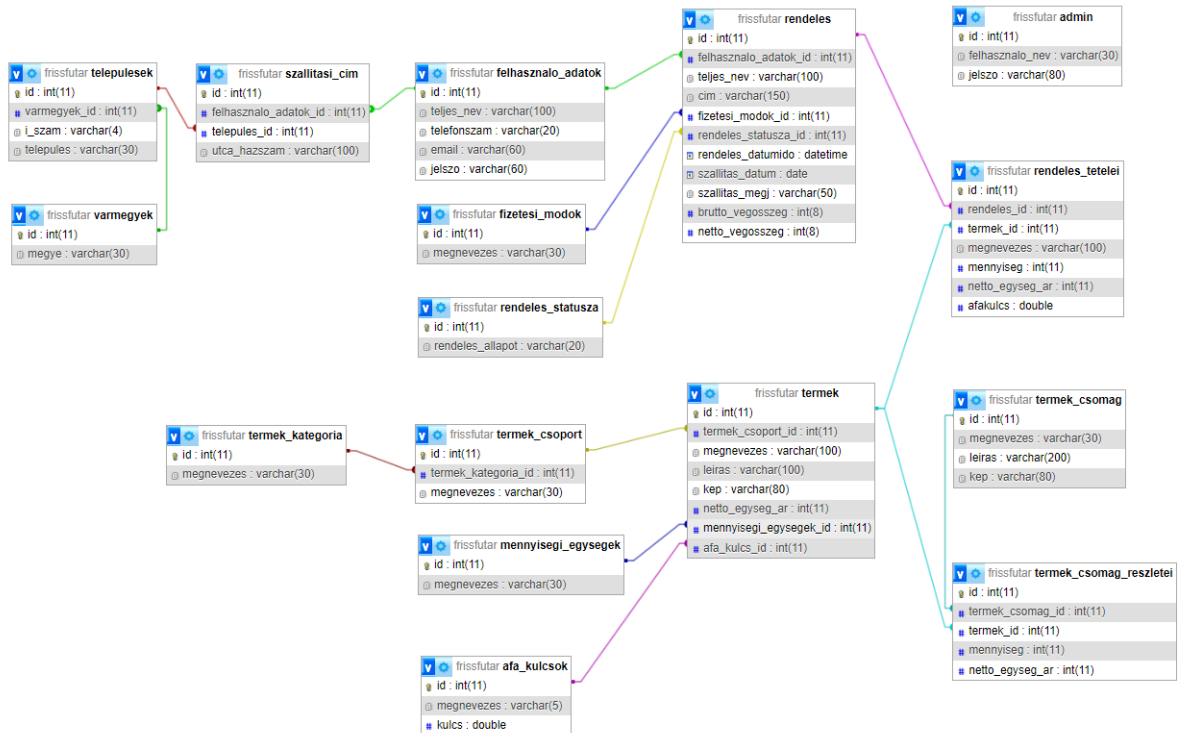
Tárolómotor: InnoDB

Alapértelmezett illesztés: utf8_hungarian_ci

1.3.1 Adattáblák

- telepulesek
- varmegyek
- szallitasi_cim
- felhasznalo_adatok
- rendeles
- fizetesi_modok
- rendeles_statusza
- rendeles_tetelei
- termék
- afa_kulcs
- mennyisegi_egysegek
- termék_csoport
- termék_kategoria
- termék_csomag_reszletei
- termék_csomag
- admin

1.3.2 Táblák közötti kapcsolatok



Az adatbázis megtervezése: Turza Norbert, Kertész Kornél és Fedor Benjámin

Az adatbázist, a táblákat és a kapcsolatot létrehozta: Kertész Kornél

A táblák feltöltése tesztdatokkal: Turza Norbert és Kertész Kornél

A forrásgyűjtést, valamint az adatok feltöltését végrehajtotta: Turza Norbert

1.3.3 A „telepulesek” tábla

Ez a tábla tartalmazza településeket.

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 varmegyek_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 i_szam	varchar(4)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 telepules	varchar(30)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a települések azonosítója (elsődleges kulcs)
- varmegyek_id: a vármegye azonosítója, amelyikhez tartozik az adott település
- i_szam: a települések irányítószáma
- telepules: a települések neve

```
CREATE TABLE `telepulesek` (  
  `id` int(11) NOT NULL,  
  `varmegyek_id` int(11) NOT NULL,  
  `i_szam` varchar(4) NOT NULL,  
  `telepules` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `telepulesek`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `telepulesek_ibfk_1` (`varmegyek_id`);
```

```
ALTER TABLE `telepulesek`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3579;
```

```
ALTER TABLE `telepulesek`  
  ADD CONSTRAINT `telepulesek_ibfk_1` FOREIGN KEY (`varmegyek_id`)  
  REFERENCES `varmegyek` (`id`);
```

1.3.4 A „varmegyek” tábla

Ez a tábla tartalmazza vármegyéket.

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 megye	varchar(30)	utf8mb4_general_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a vármegyék azonosítója (elsődleges kulcs)
- megye: a vármegye megnevezése

```
CREATE TABLE `varmegyek` (  
  `id` int(11) NOT NULL,  
  `megye` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```
ALTER TABLE `varmegyek`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `varmegyek`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=20;
```

1.3.5 A „szallitasi_cim” tábla

Ez a tábla tartalmazza a szállítási címek adatait

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 felhasznalo_adatok_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 telepules_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 utca_hazszam	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: szállítási cím azonosítója (elsődleges kulcs)
- felhasznalo_adatok_id: a felhasználói adatok azonosítója, ez az idegen kulcs a felhasznalo_adatok táblához
- telepules_id: a települések id-je, ez az idegen kulcs a telepulesek táblához

```
CREATE TABLE `szallitasi_cim` (  
  `id` int(11) NOT NULL,  
  `felhasznalo_adatok_id` int(11) NOT NULL,  
  `telepules_id` int(11) NOT NULL,  
  `utca_hazszam` varchar(100) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `szallitasi_cim`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `felhasznalo_adatok_id` (`felhasznalo_adatok_id`),  
  ADD KEY `telepules_id` (`telepules_id`);
```

```
ALTER TABLE `szallitasi_cim`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
```

```
ALTER TABLE `szallitasi_cim`  
  ADD CONSTRAINT `szallitasi_cim_ibfk_1` FOREIGN KEY (`felhasznalo_adatok_id`)  
  REFERENCES `felhasznalo_adatok` (`id`),  
  ADD CONSTRAINT `szallitasi_cim_ibfk_2` FOREIGN KEY (`telepules_id`)  
  REFERENCES `telepulesek` (`id`);
```

1.3.6 A „felhasznalo_adatok” tábla

Ez a tábla a tartalmazza a felhasználók adatait

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 teljes_nev	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 telefonszam	varchar(20)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 email	varchar(60)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	5 jelszo	varchar(60)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- **id**: a felhasználó adatok azonosítója (elsődleges kulcs)
- **teljes_nev**: a felhasználó teljes neve (vezetéknév, keresztnév), (regisztrációhoz)
- **telefonszam**: a felhasználó telefonszáma (regisztrációhoz)
- **email**: a felhasználó email címe (bejelentkezéshez, regisztrációhoz)
- **jelszo**: a felhasználó jelszava (bejelentkezéshez, regisztrációhoz)

```
CREATE TABLE `felhasznalo_adatok` (  
  `id` int(11) NOT NULL,  
  `teljes_nev` varchar(100) NOT NULL,  
  `telefonszam` varchar(20) NOT NULL,  
  `email` varchar(60) NOT NULL,  
  `jelszo` varchar(60) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `felhasznalo_adatok`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `felhasznalo_adatok`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;
```

1.3.7 A „rendeles” tábla

Ez a tábla tartalmazza a rendeléshez szükséges adatokat

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 felhasznalo_adatok_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 teljes_nev	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 cim	varchar(150)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	5 fizetesi_modok_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	6 rendeles_statusza_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	7 rendeles_datumido	datetime			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	8 szallitas_datum	date			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	9 szallitas_megj	varchar(50)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	10 brutto_vegosszeg	int(8)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	11 netto_vegosszeg	int(8)			Nem	Nincs			Módosítás Eldobás Több

- id: a rendelések azonosítója (elsődleges kulcs)
- felhasznalo_adatok_id: a felhasználói adatok azonosítója, a regisztrált felhasználók adatainak elérésére szolgál
- teljes_nev: a nem regisztrált felhasználó teljes neve (vezetéknév, keresztnév), bejelentkezett felhasználóknál nem kell megadni
- cim: a nem regisztrált felhasználó szállítási címe, bejelentkezett felhasználónál nem kell megadni, az eddig elmentett címekből tud választani a felhasználó(lenyíló menü, alaphelyzetben a legutóbbi címmel tölti ki, plusz gombbal lehet megadni másik címet, melyet rendelés után szintén elment)
- fizetesi_modok_id: a fizetési módok azonosítója
- rendeles_statusza_id: a rendelési státusz azonosítója, ez az idegenkulcs a rendeles_statusza táblához
- rendeles_datumido: egy adott felhasználó rendelés feldolgozásának időpontja
- szallitas_datum: akkor kell megadni, ha a felhasználó máskorra kéri a szállítást
- szallitas_megj: a felhasználó megjegyzése a rendeléshez
- brutto_vegosszeg: a fizetendő végösszeg (mennyiség * netto_egyseg_ar * afakulcs)
- nettó_vegosszeg: az áfa nélküli végösszeg (mennyiség * netto_egyseg_ar)

```
CREATE TABLE `rendeles` (
  `id` int(11) NOT NULL,
  `felhasznalo_adatok_id` int(11) NOT NULL,
  `teljes_nev` varchar(100) NOT NULL,
  `cim` varchar(150) NOT NULL,
  `fizetesi_modok_id` int(11) NOT NULL,
```



```

`rendeles_statusza_id` int(11) NOT NULL,
`rendeles_datumido` datetime NOT NULL,
`szaallitas_datum` date NOT NULL,
`szaallitas_megj` varchar(50) NOT NULL,
`brutto_vegosszeg` int(8) NOT NULL,
`netto_vegosszeg` int(8) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;

```

```

ALTER TABLE `rendeles`
ADD PRIMARY KEY (`id`),
ADD KEY `felhasznalo_adatok_id` (`felhasznalo_adatok_id`),
ADD KEY `fizetesi_modok_id` (`fizetesi_modok_id`),
ADD KEY `rendeles_statusza_id` (`rendeles_statusza_id`);

```

```

ALTER TABLE `rendeles`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

```





```

ALTER TABLE `rendeles`
ADD CONSTRAINT `rendeles_ibfk_1` FOREIGN KEY (`felhasznalo_adatok_id`)
REFERENCES `felhasznalo_adatok` (`id`),
ADD CONSTRAINT `rendeles_ibfk_2` FOREIGN KEY (`fizetesi_modok_id`)
REFERENCES `fizetesi_modok` (`id`),
ADD CONSTRAINT `rendeles_ibfk_3` FOREIGN KEY (`rendeles_statusza_id`)
REFERENCES `rendeles_statusza` (`id`);

```

1.3.8 A „fizetesi_modok” tábla

Ez a tábla tartalmazza a rendelésnél kiválasztható fizetési módokat

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id 	int(11)			Nem	Nincs		AUTO_INCREMENT	 Módosítás  Eldobás Több
<input type="checkbox"/> 2	megnevezes	varchar(30)	utf8_hungarian_ci		Nem	Nincs			 Módosítás  Eldobás Több

- id: a fizetési módok azonosítója (elsődleges kulcs)
- megnevezes: fizetési módok megnevezése(bankkártya, SZÉP-kártya, készpénz)

```
CREATE TABLE `fizetesi_modok` (  
  `id` int(11) NOT NULL,  
  `megnevezes` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `fizetesi_modok`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `fizetesi_modok`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
```

1.3.9 A „rendeles_statusza” tábla

Ez a tábla tartalmazza a rendelés állapotát

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/> 2	rendeles_allapot	varchar(20)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a rendelés státuszának az azonosítója (elsődleges kulcs)
- rendeles_allapot: a rendelés állapota (megkaptuk-készítjük, elkészült-szállítjuk, kiszállítva)

```
CREATE TABLE `rendeles_statusza` (
  `id` int(11) NOT NULL,
  `rendeles_allapot` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `rendeles_statusza`
ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `rendeles_statusza`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
```

1.3.10 A „rendeles_tetelei” tábla

Ez a tábla tartalmazza a felhasználó rendelésének tételeit és azoknak a paramétereit (árát, mennyiségét, nevét) ebből lesz a blokk

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/> 2	rendeles_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 3	termek_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 4	megnevezes	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 5	mennyiseg	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 6	netto_egyseg_ar	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 7	afakulcs	double			Nem	Nincs			Módosítás Eldobás Több

- id: a rendelés tételeinek az azonosítója (elsődleges kulcs)
- rendeles_id: a rendelés azonosítója, ez az idegenkulcs a rendeles táblához
- termék_id: a termék azonosítója, ez az idegenkulcs a termék táblához
- megnevezes: a rendelésben szereplő termékek tételenkénti megnevezése
- mennyiség: a rendelt termékek tételenkénti mennyisége
- netto_egyseg_ar: a termékek tételenkénti nettó ára
- afakulcs: az adott termékekhez tartozó áfakulcs

```
CREATE TABLE `rendeles_tetelei` (
  `id` int(11) NOT NULL,
  `rendeles_id` int(11) NOT NULL,
  `termek_id` int(11) NOT NULL,
  `megnevezes` varchar(100) NOT NULL,
  `mennyiseg` int(11) NOT NULL,
  `netto_egyseg_ar` int(11) NOT NULL,
  `afakulcs` double NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `rendeles_tetelei`
  ADD PRIMARY KEY (`id`),
  ADD KEY `rendeles_id` (`rendeles_id`),
  ADD KEY `termek_id` (`termek_id`);
```

```
ALTER TABLE `rendeles_tetelei`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `rendeles_tetelei`
  ADD CONSTRAINT `rendeles_tetelei_ibfk_1` FOREIGN KEY (`rendeles_id`)
REFERENCES `rendeles` (`id`),
  ADD CONSTRAINT `rendeles_tetelei_ibfk_2` FOREIGN KEY (`termek_id`)
REFERENCES `termek` (`id`);
```

1.3.11 A „termek” tábla

Ez a tábla tartalmazza az oldalon lévő termékek adatait

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/> 2	termek_csoport_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 3	megnevezes	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 4	leiras	varchar(100)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 5	kep	varchar(80)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 6	netto_egyseg_ar	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 7	mennyisegi_egysegek_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 8	afa_kulcs_id	int(11)			Nem	Nincs			Módosítás Eldobás Több

- id: a termék azonosítója (elsődleges kulcs)
- termék_csoport_id: : a termék csoportok azonosítója, ez az idegenkulcs a termék_csoport táblához
- megnevezes: az oldalon szereplő termékek teljes neve
- leiras: az oldalon szereplő termékek részletes leírása
- kep: az oldalon szereplő termékekről készült illusztráció
- netto_egyseg_ar: az oldalon szereplő termékek nettó ára
- mennyisegi_egysegek_id: a mennyiségi egységek azonosítója, ez az idegenkulcs a mennyisegi_egysegek táblához
- afa_kulcs_id: az áfakulcsok azonosítója, ez az idegenkulcs a afa_kulcs táblához

```
CREATE TABLE `termek` (
  `id` int(11) NOT NULL,
  `termek_csoport_id` int(11) NOT NULL,
  `megnevezes` varchar(100) NOT NULL,
  `leiras` varchar(100) NOT NULL,
  `kep` varchar(80) NOT NULL,
  `netto_egyseg_ar` int(11) NOT NULL,
  `mennyisegi_egysegek_id` int(11) NOT NULL,
  `afa_kulcs_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `termek`
  ADD PRIMARY KEY (`id`),
  ADD KEY `mennyisegi_egysegek_id` (`mennyisegi_egysegek_id`),
```

```
ADD KEY `termek_csoport_id` (`termek_csoport_id`),
ADD KEY `afa_kulcs_id` (`afa_kulcs_id`);
```

```
ALTER TABLE `termek`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=82;ALTER TABLE `termek`
ADD CONSTRAINT `termek_ibfk_1` FOREIGN KEY (`mennyisegi_egysegek_id`)
REFERENCES `mennyisegi_egysegek` (`id`),
ADD CONSTRAINT `termek_ibfk_2` FOREIGN KEY (`termek_csoport_id`)
REFERENCES `termek_csoport` (`id`),
ADD CONSTRAINT `termek_ibfk_3` FOREIGN KEY (`afa_kulcs_id`) REFERENCES
`afa_kulcsok` (`id`);
```

1.3.12 Az „afa_kulcs” tábla

Ez a tábla tartalmazza a termékekhez kapcsolt áfakulcsokat

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/> 2	megnevezes	varchar(5)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 3	kulcs	double			Nem	Nincs			Módosítás Eldobás Több

- id: az áfa kulcsok azonosítója (elsődleges kulcs)
- megnevezes: az áfakulcsok megnevezése
- kulcs: az áfakulcsok értéke (pl.: 0.27)

```
CREATE TABLE `afa_kulcsok` (
`id` int(11) NOT NULL,
`megnevezes` varchar(5) NOT NULL,
`kulcs` double NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `afa_kulcsok`
ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `afa_kulcsok`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;
```

1.3.13 A „mennyisegi_egysegek” tábla

Ez a tábla tartalmazza a termékekhez használt mennyiségi egységeket (db, gramm, stb.)

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id 	int(11)			Nem	Nincs		AUTO_INCREMENT	 Módosítás  Eldobás Több
<input type="checkbox"/>	2 megnevezes	varchar(30)	utf8_hungarian_ci		Nem	Nincs			 Módosítás  Eldobás Több

- id: a mennyiségi egységek azonosítója (elsődleges kulcs)
- megnevezes: a mennyiségi egységek megnevezése

```
CREATE TABLE `mennyisegi_egysegek` (  
  `id` int(11) NOT NULL,  
  `megnevezes` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `mennyisegi_egysegek`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `mennyisegi_egysegek`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;
```

1.3.14 A „termek_csoport” tábla

Ez a tábla tartalmazza a termékek csoportjait (étel, ital)

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 megnevezés	varchar(30)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 leírás	varchar(200)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 kép	varchar(80)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a termék csoportok azonosítója (elsődleges kulcs)
- termék_kategoria_id: : a termék kategóriák azonosítója
- megnevezés: a termék csoportok megnevezése

```
CREATE TABLE `termek_csoport` (  
  `id` int(11) NOT NULL,  
  `termek_kategoria_id` int(11) NOT NULL,  
  `megnevezes` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `termek_csoport`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `termek_kategoria_id` (`termek_kategoria_id`);
```

```
ALTER TABLE `termek_csoport`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=17;
```

```
ALTER TABLE `termek_csoport`  
  ADD CONSTRAINT `termek_csoport_ibfk_1` FOREIGN KEY (`termek_kategoria_id`)  
  REFERENCES `termek_kategoria` (`id`);  
COMMIT;
```


1.3.15 A „termek_kategoria” tábla

Ez a tábla tartalmazza a termékek kategóriáit (üdítők, szeszes italok, pizzák stb.)

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 megnevezes	varchar(30)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a termék kategóriák azonosítója (elsődleges kulcs)
- megnevezes: a termék kategóriáknak a megnevezése

```
CREATE TABLE `termek_kategoria` (  
  `id` int(11) NOT NULL,  
  `megnevezes` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `termek_kategoria`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `termek_kategoria`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;
```

1.3.16 A „termek_csomag_reszletei” tábla

Ez a tábla tartalmazza termék csomagoknak a tulajdonságait (árát, mennyiségét)

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/> 1	id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/> 2	termek_csomag_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 3	termek_id	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 4	mennyiseg	int(11)			Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/> 5	netto_egyseg_ar	int(11)			Nem	Nincs			Módosítás Eldobás Több

- id: a termék csomagok részleteinek az azonosítója (elsődleges kulcs)
- termék_csomag_id: : a termék csomagok azonosítója
- termék_id: : a termékek azonosítója
- mennyiseg: a termék csomagban lévő termékek mennyisége
- **netto_egyseg_ar: most ez nemtom minek a nettó ára

```
CREATE TABLE `termek_csomag_reszletei` (
  `id` int(11) NOT NULL,
  `termek_csomag_id` int(11) NOT NULL,
  `termek_id` int(11) NOT NULL,
  `mennyiseg` int(11) NOT NULL,
  `netto_egyseg_ar` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `termek_csomag_reszletei`
  ADD PRIMARY KEY (`id`),
  ADD KEY `termek_csomag_id` (`termek_csomag_id`),
  ADD KEY `termek_id` (`termek_id`);
```

```
ALTER TABLE `termek_csomag_reszletei`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `termek_csomag_reszletei`
  ADD CONSTRAINT `termek_csomag_reszletei_ibfk_1` FOREIGN KEY
  (`termek_csomag_id`) REFERENCES `termek_csomag` (`id`),
  ADD CONSTRAINT `termek_csomag_reszletei_ibfk_2` FOREIGN KEY (`termek_id`)
  REFERENCES `termek` (`id`);
```

1.3.17 A „termek_csomag” tábla

Ez a tábla tartalmazza a termék csomagoknak a tulajdonságait (leírás, kép)

#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 megnevezes	varchar(30)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 leiras	varchar(200)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	4 kep	varchar(80)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: a termék csomagok azonosítója (elsődleges kulcs)
- megnevezes: a termék csomagok megnevezése
- leiras: a termék csomagok leírása
- kep: a termék csomagról készült illusztráció

```
CREATE TABLE `termek_csomag` (  
  `id` int(11) NOT NULL,  
  `megnevezes` varchar(30) NOT NULL,  
  `leiras` varchar(200) NOT NULL,  
  `kep` varchar(80) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `termek_csomag`  
  ADD PRIMARY KEY (`id`);
```

```
ALTER TABLE `termek_csomag`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;
```

1.3.18 Az „admin” tábla

Ez a tábla tartalmazza az admin alkalmazáshoz tartozó adatokat

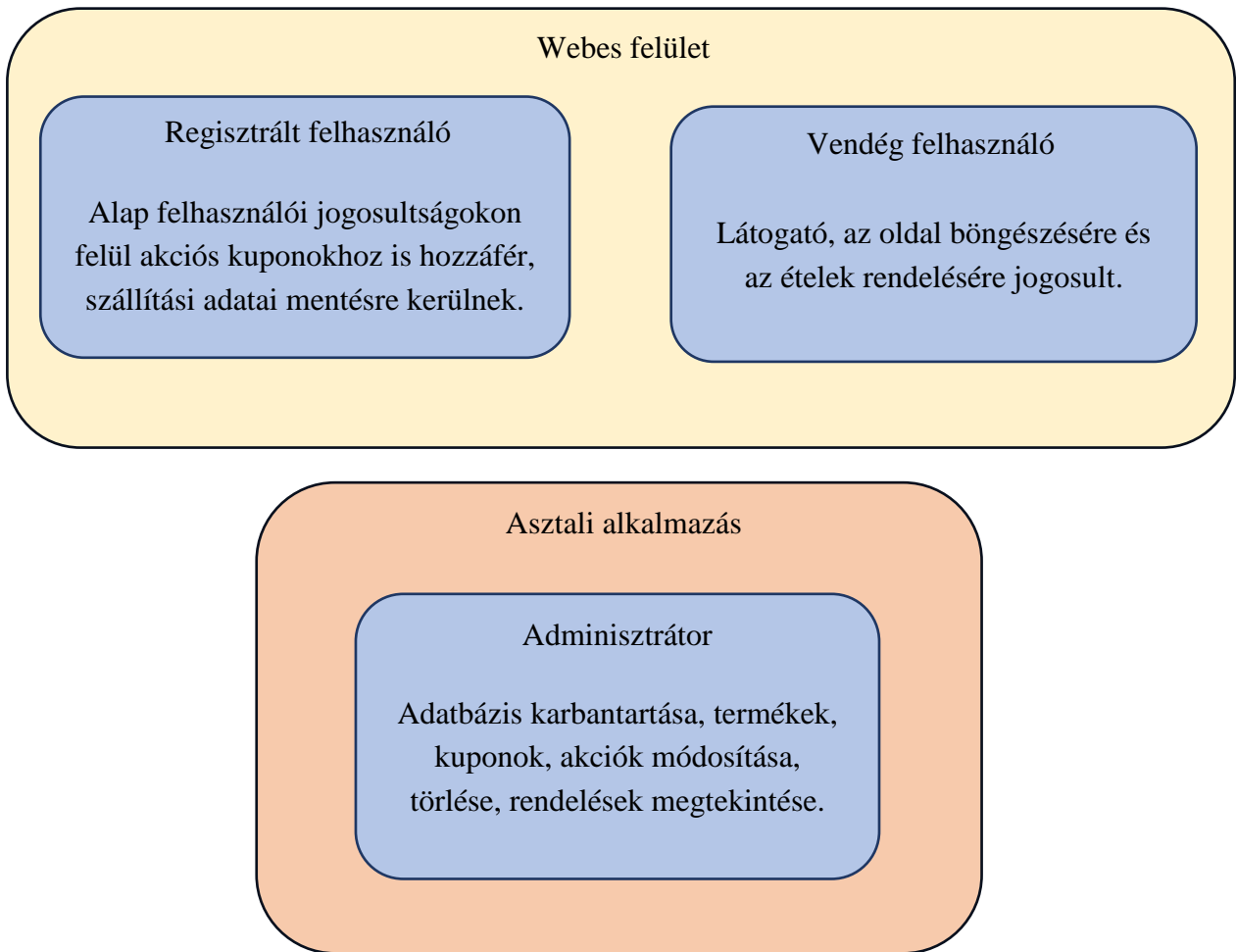
#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra	Művelet
<input type="checkbox"/>	1 id	int(11)			Nem	Nincs		AUTO_INCREMENT	Módosítás Eldobás Több
<input type="checkbox"/>	2 felhasznalo_nev	varchar(30)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több
<input type="checkbox"/>	3 jelszo	varchar(80)	utf8_hungarian_ci		Nem	Nincs			Módosítás Eldobás Több

- id: az admin azonosítója (elsődleges kulcs)
- felhasznalo_nev: a bejelentkezéshez szükséges felhasználónév
- jelszo: a bejelentkezéshez szükséges jelszó

```
CREATE TABLE `admin` (  
  `id` int(11) NOT NULL,  
  `felhasznalo_nev` varchar(30) NOT NULL,  
  `jelszo` varchar(80) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_hungarian_ci;
```

```
ALTER TABLE `admin`  
  ADD PRIMARY KEY (`id`);  
ALTER TABLE `admin`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
```

1.4 Használati esetmodell, szerepkörök



1.5 Modulok ismertetése

1.5.1 Webes alkalmazás

1.5.1.1 Főoldal

Nem bejelentkezett felhasználó számára:



Bejelentkezett felhasználó számára:



A termékek megjelenítésére szolgáló metódus:

```
function termeketMegjelenit($kategoria,$termek_csoport_adatok,$adatok)
{
    $n = count($termek_csoport_adatok);
    for ($i=0; $i < $n; $i++) {
        if ($termek_csoport_adatok[$i]['kategoria']==$kategoria){
            $csoport = $termek_csoport_adatok[$i]['megn'];
            echo
            "<h3 id='$csoport'$>$csoport</h3>"
            <div class='menu-item-container' >
                <div class='lapoz'><button class='atlatso-gomb' id='$csoport-elozo' onclick='balra('$csoport')'><i class='fa-solid
            echo "<div class='oldal-kartya-hidden' id='id_$csoport'>";
            //az aktuális csoport termékei
            $n2 = count($adatok);
            for ($j=0; $j < $n2; $j++) {
                if ($adatok[$j]['csoport']==$csoport) {
                    $id = $adatok[$j]['id'];
                    $kep = $adatok[$j]['kep'];
                    $megnevezes = $adatok[$j]['megn'];
                    $leiras = $adatok[$j]['leiras'];
                    $brutto_ar = $adatok[$j]['ar'];
                    echo"<div class='oldal-kartya_tartalom ok_$csoport'>
                        <div class='oldal-kartya-header'>
                            <img class='img-fluid kartya-img' src='img/$kep'alt=''>$megnevezes</div>
                            <div class='oldal-kartya-body'>$leiras</div>
                            <div class='oldal-kartya-footer'>$brutto_ar Ft";
                    echo ' <button class="kosarhoz" onclick="HozzaAd('.$adatok[$j]['id'].',1)">Rendelés<i class="fa-solid fa-plu
                    echo"</div>";
                }
            }
            echo "</div>";
            echo "<div class='lapoz'><button class='atlatso-gomb' id='$csoport-kovetkezo' onclick='jobbba('$csoport')'><i class='fa
            echo "</div>";
```

A termékek megjelenítését megvalósította: Fedor Benjámín

1.5.1.2 Regisztráció

A regisztrálási űrlap 3 részre van bontva, melyek közt lapozni lehet, az adatok tovább lépés előtt ellenőrzésre kerülnek, és csak helyes adat esetén engedi tovább a felhasználót.

Az utolsó oldalon lévő „Regisztráció” gomb megfelelő adatok megadása esetén regisztrálja a felhasználót, és a Bejelentkezés oldalt jeleníti meg. Nem megfelelő adatok megadásakor értesíti a felhasználót a sikertelen regisztrációról és a regisztrációs űrlap elejére irányítja.

Sikeres regisztráció esetén az adatok az adatbázisban eltárolásra kerülnek, a felhasználó pedig értesítést kap a regisztráció sikerességéről.

1.5.1.3 Bejelentkezés

„Bejelentkezés” gomb helyes adatok megadása esetén a Főoldalra irányítja a bejelentkezett felhasználót. Helytelen adatok esetén értesíti a felhasználót a sikertelen bejelentkezésről.

A bejelentkezett felhasználó e-mail címe SESSION-ben kerül tárolásra, a továbbiakban ez alapján minden fontos adata lekérdezhető az adatbázisból. (pl: szállítási cím, telefonszám)

A regisztrációs és bejelentkezési folyamatot megvalósította: Turza Norbert és Kertész Kornél.

Az oldal megjelenéséért és használatáért felelős: Fedor Benjámin

1.5.1.4 Kosár:

A kosár tartalma a főoldalon és a kosár oldalon is módosítható, A kosár adatai SESSION-ben kerülnek eltárolásra.

Egy termék hozzáadása a kosárhoz Javascriptben jQuery segítségével:

```
function HozzaAd(id,mennyiseg)
{
    console.log(id+ ' <=> ' + mennyiseg);
    document.getElementById('termek').value = id;
    document.getElementById('darab').value = mennyiseg;
    $.post
    (
        "kosarba_tesz.php",
        $("#frm_kosar").serialize(),
    )
    .done(function(data,status){
        if(status == 'success')
        {
            var ret = JSON.parse(data);
            if(ret['hiba_kod'] == 0)
            {
                Kosar_Frissit();
            }
        }
    })
}
```

A kosár oldalt megvalósította: Fedor Benjámin

1.5.1.5 Adatok megadása

Adatok megadása

Teljes név
Teszt Elek

Telefonszám
+36301234567

Település
Vác

Cím
|

Fizetési mód
bankkártya

Megjegyzés
Ide írja a megjegyzését a rendeléshez...

← →

Ezen oldalon a felhasználó a rendeléshez szükséges adatokat tudja megadni, regisztrált és bejelentkezett felhasználó esetén a szükséges adatok automatikus kitöltésre kerülnek, ezt azonban a felhasználó szabadon megváltoztathatja.

Az adatok megadására szolgáló oldalon lehetősége van a felhasználónak a vissza nyíllra kattintva visszalépni a Kosár oldalra, illetve a tovább nyíllra kattintva sikeres adatmegadás esetén a Rendelés véglegesítésére szolgáló oldalra lépni.

Az adatok megadása oldal backendjét készítette: Kertész Kornél.

A Frontendet készítette: Fedor Benjámin.

1.5.1.6 Rendelés véglegesítése

The screenshot shows a mobile application interface for finalizing an order. It is divided into two main sections: 'Szállítási adatok' (Shipping data) and 'Rendelés tételei' (Order items).

Szállítási adatok

Név	Teszt Elek
Telefonszám	+36301234567
Település	Vác
Cím	Kossuth tér 2

Rendelés tételei

Megnevezés	Egységár	Mennyiség
Gyros tál	2650 Ft	1 db
Sajtleves	1590 Ft	2 db
Cordon Bleu	3220 Ft	1 db

Fizetési Mód: készpénz

Végösszeg: 9050 Ft

At the bottom, there are two yellow buttons: 'Vissza' (Back) and 'Rendelés' (Order).

A rendelési folyamat szempontjából a végső oldal, a felhasználó megtekintheti az általa megadott rendeléshez kapcsolódó adatokat, valamint a rendelés tételeit, és minden hozzátartozó információt.

A „Rendelés” gombbal véglegesítheti rendelését, mely ezután feldolgozásra, és az adatbázisban való rögzítésre kerül. A felhasználó visszajelzést kap a rendelésének feldolgozásáról.

A Rendelés feldolgozásáért felelős: Kertész Kornél

A Rendelés véglegesítése weboldal megjelenéséért felelős: Fedor Benjámin

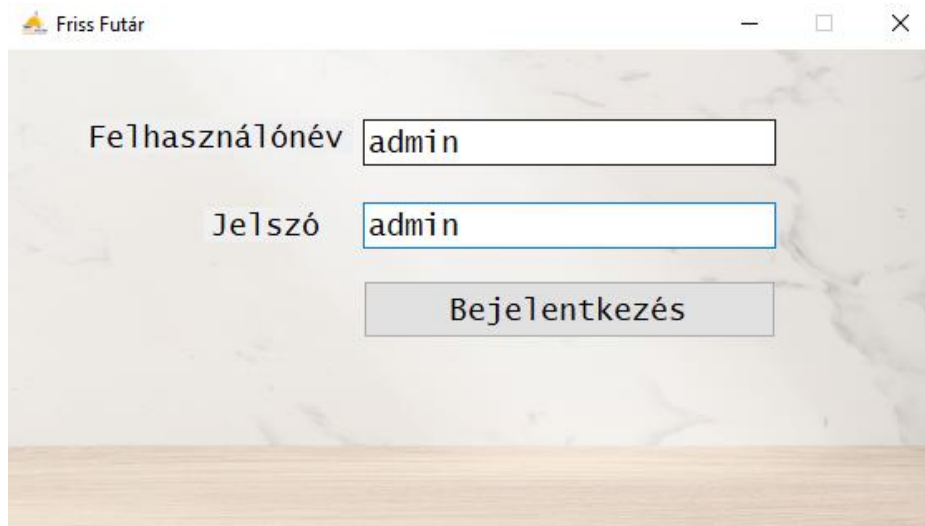
1.5.1.7 Felhasználói adatok módosítása

A bejelentkezett felhasználónak lehetősége van megváltoztatni telefonszámát, e-mail címét, illetve szállítási címét. Az oldal a főoldalról nyitható meg, a jobb felső sarokban található fiók ikon segítségével.

Az adatok módosításáért felelt: Kertész Kornél

1.5.2 Asztali alkalmazás

1.5.2.1 Bejelentkezés



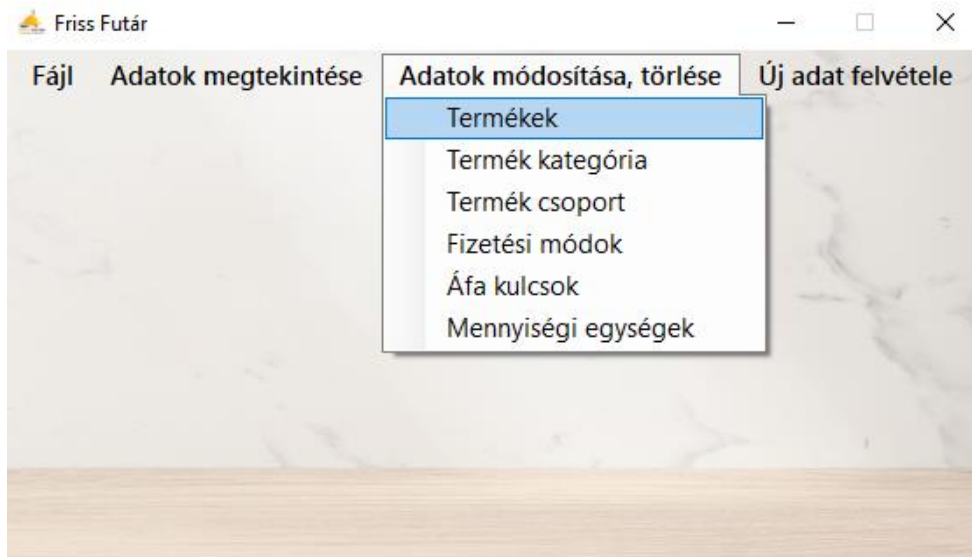
```
StringBuilder sb = new StringBuilder();
MD5 md5 = MD5.Create();

byte[] hash_ertek = md5.ComputeHash(Encoding.UTF8.GetBytes(jelszo_tb.Text));
foreach (byte betu in hash_ertek)
{
    sb.Append($"{betu:x2}");
}
string md5_jelszo = sb.ToString();
if (md5_jelszo == DB.SelectOneValue("SELECT jelszo FROM `admin` WHERE felhasznalo_nev = '" + felhasznalo_nev + "' ;"))
{
    MessageBox.Show("Sikeres belépés!");
}
```

A „Bejelentkezés” gomb helyes adatok megadása esetén, az adatbázisban tárolt adatok alapján a főablakra irányítja a bejelentkezett admin. Helytelen adatok esetén értesítést küld.

A jelszó titkosítására MD5-öt használunk.

1.5.2.2 Főablak



A „Fájl” menüpont alatt található a „Kilépés” amivel bezárul az applikáció. Az eseménykezelések és a program működéséért felelős kódok nagy része a MainForm-ben található.

1.5.2.3 Adatok megtekintése

The screenshot shows the 'Szállítási címek' (Shipping addresses) window. It contains a table with the following data:

ID	teljes_nev	telefonszam	email	jelszo
1	Nem Regisztrált	00000000000	nem@regisztralt.hu	87b524b4689c0a64177...
2	Gipsz Jakab	06301234567	dodo@gmail.com	5db5ef897b70844afbd...
3	Teszt Elek	hjhsdhjgj	turza.norbert2002...	58ae1f97879ab574348...
4	Jo Elek	hjhsd54	jozsi435@gmail.com	5db5ef897b70844afbd...
*				

Az „Adatok megtekintése” menüpont alatt megtekinthetjük az adatbázis összes táblájának az adatait egyenként.

1.5.2.4 Adatok módosítása, törlése

The image displays two screenshots of a software window titled "Termék módosítás" (Product Modification). The top screenshot shows the initial state with three empty dropdown menus and three buttons: "Módosít" (Modify), "Töröl" (Delete), and "Mégse" (Cancel). The bottom screenshot shows the form after clicking "Módosít". It contains fields for "Termék csoport" (Frissensültek), "Megnevezés" (Rostonsült Csirkemell), "Leírás" (Omlós, szaftos, sült csirkemell. Allergént nem tartalmaz.), "kép feltöltés" (csirkemell_roston.jpg), "Nettó ár" (1900), "Mennyiségi egységek" (adag), and "Áfakulcs" (15%). There are "OK" and "Mégse" buttons on the right.

Az „Adatok módosítása, törlése” segítségével módosíthatunk és törölhetünk, a jövőben szükséges változtatásokat végezhetjük el az adatbázis bizonyos tábláin. Bármelyik lehetőségre kattintva megnyílik egy új form amin 3 gomb lesz elérhető:

„Módosít”:

A gomb megnyomásakor az előtte megadott adatok alapján egy következő formot nyit meg, ahol az adatbázisból betöltött adatokat külön-külön lehet változtatni, az „OK” gombbal elfogadja és az „UPDATE” SQL paranccsal feltölti az adatokat az adatbázisba - ha minden ki van töltve - a „Mégse” gomb bezárja ezt a formot és az előző formra irányít, megtartva az eddigi adatokat.

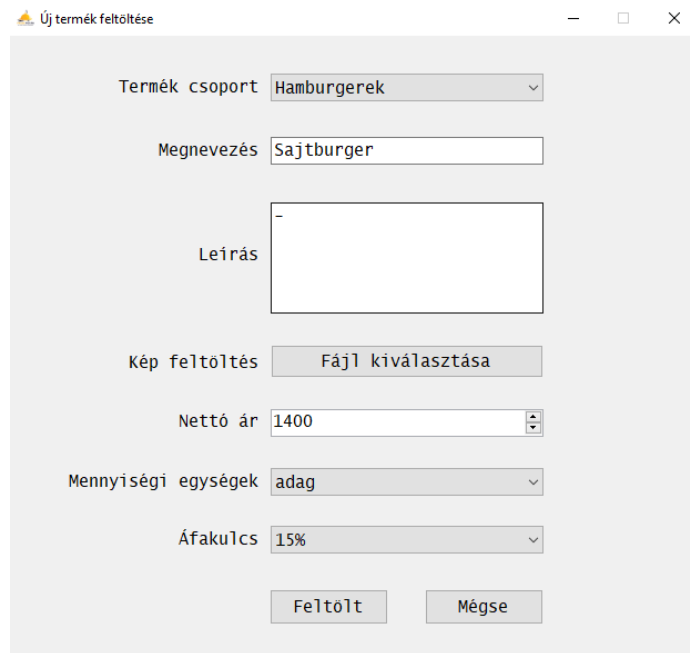
„Töröl”:

A kiválasztott adatot a „DELETE” SQL paranccsal véglegesen törli az adatbázisból, de csak akkor, ha az adott táblához alárendelt táblában nincsenek adatok.

„Mégse”:

Bezárja ezt a formot és a főformra irányít.

1.5.2.5 Új adat felvétele



Új termék feltöltése

Termék csoport: Hamburgerek

Megnevezés: Sajtburger

Leírás:

Kép feltöltés: Fájl kiválasztása

Nettó ár: 1400

Mennyiségi egységek: adag

Áfakulcs: 15%

Feltölt Mégse

Az „Új adat felvétele” a megadott táblákat új adatokkal tölti fel. Bármelyik menüpontra kattintva megnyit egy új formot amin 2 gomb található:

„Feltölt”:

Az új adatokat feltölti az adatbázisba.

„Mégse”:

Bezárja a formot és visszairányít a főformra.

Az admin applikációt megvalósította: Kertész Kornél és Turza Norbert

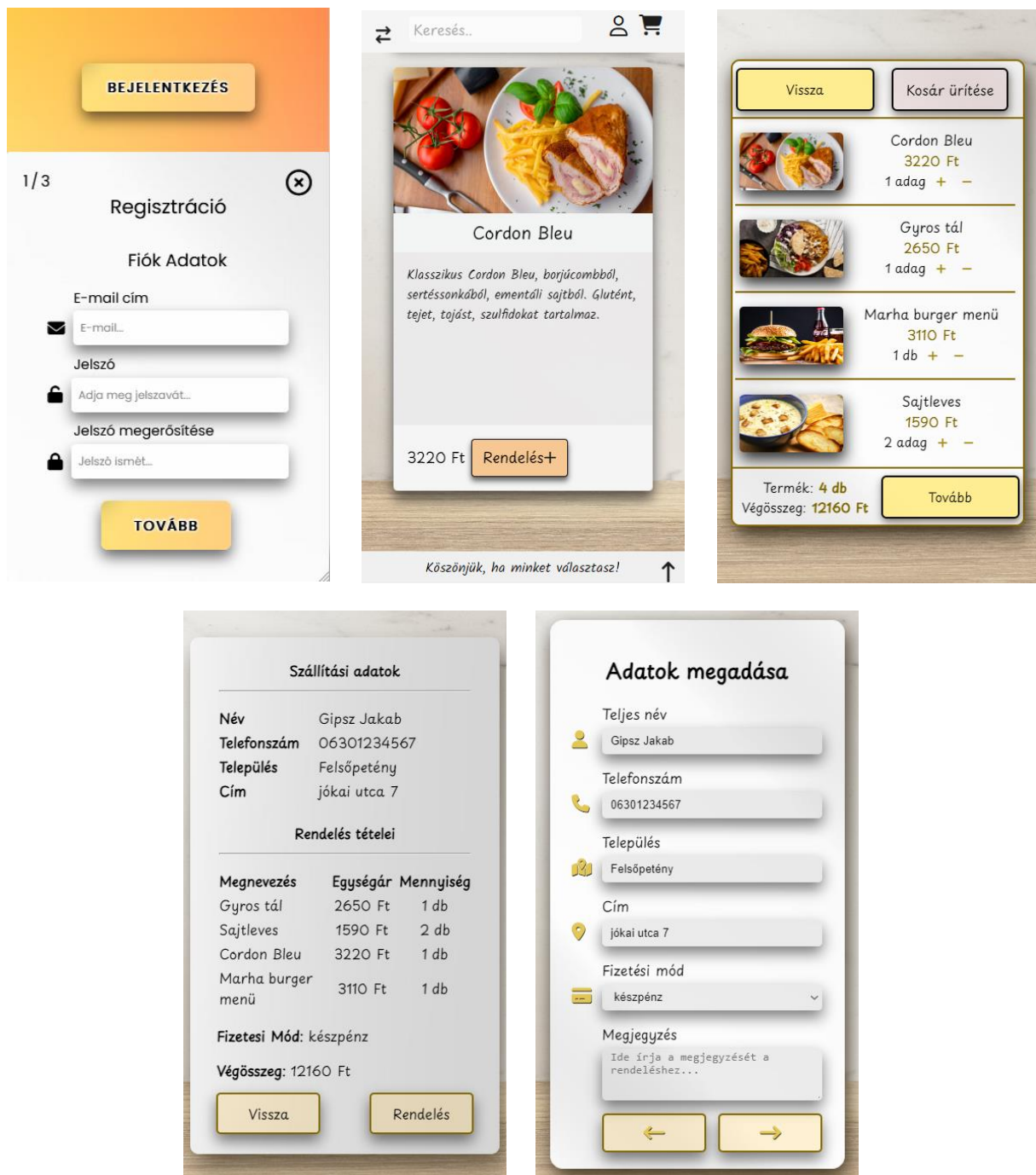
1.6 Tesztelések

1.6.1 Reszponzivitás

Minden oldal reszponzív, asztali gépen, telefonon, tableten tesztelve.

Hibák:

Néhány bug előfordul a különböző böngészőknél, amikor visszalép a felhasználó, nem mindig a kijelző méretének megfelelően jelenik meg az oldal, ezt a hibát a telefon elfordítása, majd visszafordítása helyreállítja.



Egy -a reszponzivitás megvalósításáért felelő kódrészlet javascriptben:

```
window.addEventListener("resize", () => {  
  if (document.documentElement.clientWidth <= 700) {  
    content.style.marginLeft = "0px";  
    content.style.width = "100vw";  
    sidebar.style.backgroundColor = "var(--sidebar_hatter)";  
  }  
  else {  
    sidebar.style.backgroundColor = "var(--sidebar_hatter-transparent)";  
    let currentVW = 100-convertPXTovW(320);  
    let VW = parseFloat(currentVW);  
    //sidebar ki és behúzása  
    //content szélesítése 320px-el  
    if (clickSzam % 2 == 1) {  
      content.style.marginLeft = "0px";  
      content.style.width = "100vw";  
      footer_content.style.marginLeft = '0px';  
    }  
    //content zsugorítása 320px-el  
    else {  
      content.style.marginLeft = "320px";  
      // az aktualis 100vw -320px érték megadása  
      content.style.width = VW+"vw";  
      footer_content.style.marginLeft = '320px';  
    }  
  }  
});
```

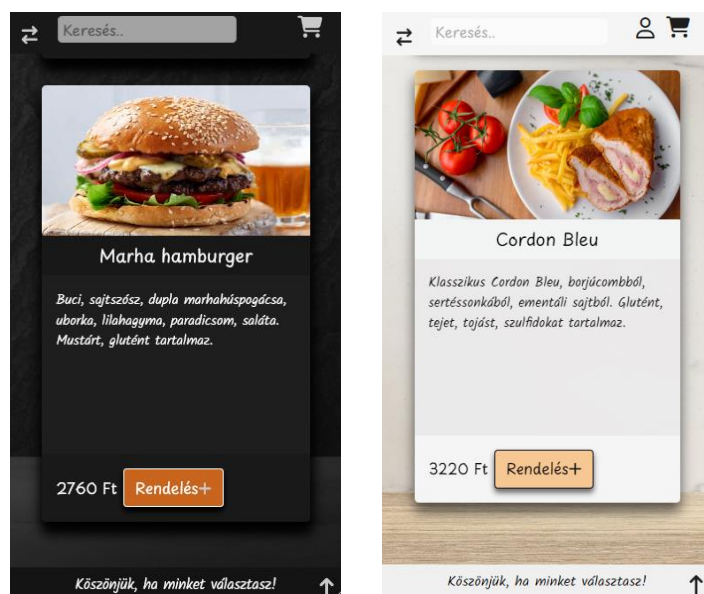
Az oldal tartalmának szélessége dinamikusan változik átméretezéskor, ha egy kritikus szélesség alá csökken, az oldalsó menü is behúzásra kerül (telefonoknál alapértelmezett)

Az oldal betöltésekor is hasonló vizsgálat zajlik le.

A weboldal reszponzivitását megvalósította: Fedor Benjámín

Tesztelte: Turza Norbert és Kertész Kornél

1.6.2 Téma váltás



A főoldalon lehetősége van a felhasználónak a sötét és világos tema közt váltani, az éppen használt témát a program eltárolja, és az oldal legközelebbi látogatása esetén ezt használja, valamint a további oldalak megnyitásakor is a felhasználó által beállított témát használja.

Hibák:

Google Chrome-nál előforduló hiba, hogy az aloldalak megnyitásakor, az alapértelmezett témát használja az oldal, ennek okára még nem jöttünk rá.

A fenti hibán kívül a funkció megfelelően működik.

Egy kódrészlet a téma mentésére, majd későbbi beállítására Javascriptben:

```
if (theme.getAttribute('href') == 'css/indexLight.css') {  
    //mentes  
    localStorage.setItem('theme_pref', 'css/indexDark.css');  
    //téma beállítás  
    const theme_prefered = localStorage.getItem('theme_pref');  
    if (theme_prefered != null) {  
        theme.setAttribute('href', theme_prefered);  
    }  
    else{  
        theme.setAttribute('href', 'css/indexLight.css');  
    }  
}
```

A témaváltásért, és az oldal megjelenését megvalósította: Fedor Benjámin

1.6.3 Adatok kezelése

A felhasználó visszajelzést kap az általa megadott adatok sikeres feldolgozásáról, illetve arról is, ha az adatok nem megfelelőek.

1.6.3.1 Bejelentkezésnél

Helytelen jelszó vagy e-mail cím megadása esetén a felhasználó értesítést

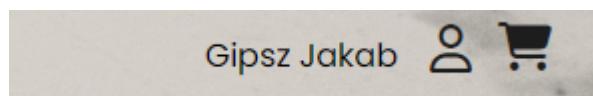


kap:

Sikeres bejelentkezést követően a főoldalon többek között a felhasználó neve kerül megjelenítésre.

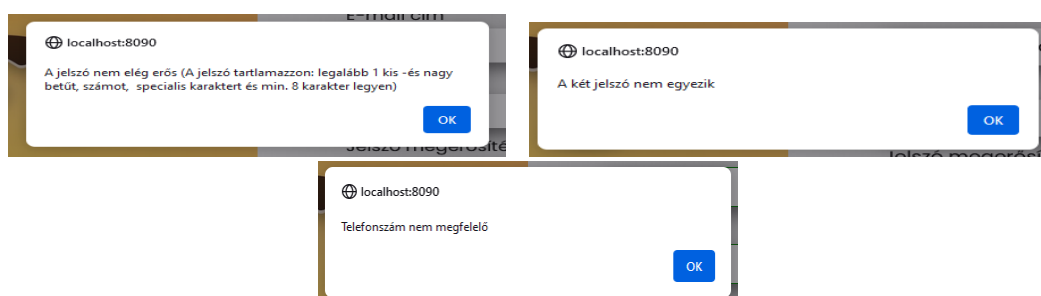
1.6.3.2 Regisztrációnál

A regisztrációs folyamat 3 részre van bontva, az adatok ellenőrzése részenként



történik, ezáltal helytelen adatmegadás esetén a felhasználónak csak az aktuális modul adatait kell újra megadnia, ezzel elősegítve a gördülékenyebb felhasználói élményt.

Néhány visszajelzés a felhasználó számára, nem megfelelő adat megadása után



Visszajelzés sikeres regisztráció esetén:



A regisztráció és bejelentkezés backendje: Kertész Kornél és Turza Norbert

A regisztráció és bejelentkezés folyamatának tesztelése: Kertész Kornél és Fedor Benjámín

1.6.3.3 Rendelési adatok

A rendelési adatok megadása esetén is kap visszajelzést a felhasználó,

amennyiben az általa megadott adat nem megfelelő.

1.6.4 Adatbázis

1.6.4.1 Kapcsolat az adatbázissal

Az adatbázishoz történő kapcsolódás php-ban PDO-val történik.

```
function kaps()  
{  
    $host = "localhost";  
    $user = "root";  
    $pwd = "";  
    $dbname = "frissfutar";  
  
    $s = "mysql:host=$host;dbname=$dbname;charset=utf8;port:3306";  
    $db = new PDO($s, $user, $pwd);  
    return $db;  
}
```

1.6.4.2 Adatok feltöltése az adatbázisba

A project elején a termékek feltöltése php eljárásokon keresztül történt.

```
function termék_csoporthoz_töltés()  
{  
    $termek_nevek = array("Hamburgerek", "Frissensültek", "Tészták", "Levesek", "Gyrosok", "Pizzák", "Mártások", "Saláták", "Köretetek",  
        $ital_nevek = array("Szénsavas üdítők", "Gyümölcslevek", "Energiaitalok", "Szeszes italok", "Teák", "Ásványvizek");  
  
    for ($i = 0; $i < count($termek_nevek); $i++) {  
        $id = $i + 1;  
        $t_k_id = 1;  
        $megnevezes = $termek_nevek[$i];  
        $sql = "INSERT INTO `termek_csoporthoz` (`id`, `termek_kategoria_id`, `megnevezes`) VALUES ('$id', '$t_k_id', '$megnevezes')";  
        $db = mysqli_connect($host, $user, $password, $db_name);  
        $stmt = $db->prepare($sql);  
        $stmt->execute();  
        $db = null;  
    }  
    for ($i = 0; $i < count($ital_nevek); $i++) {  
        $id = NULL;  
        $t_k_id = 2;  
        $megnevezes = $ital_nevek[$i];  
        $sql = "INSERT INTO `termek_csoporthoz` (`id`, `termek_kategoria_id`, `megnevezes`) VALUES ('$id', '$t_k_id', '$megnevezes')";  
        $db = mysqli_connect($host, $user, $password, $db_name);  
        $stmt = $db->prepare($sql);  
        $stmt->execute();  
        $db = null;  
    }  
}
```

Adatok feltöltése rendelés leadása esetén.

```
$rendelés_töltés = $db->query("INSERT INTO `rendelés` (`id`, `felhasznalo_adatok_id`, `teljes_nev`, `cim`, `fizetesi_modok_id`,  
    `rendelés_statusza_id`, `rendelés_datumido`, `szallitas_datum`, `szallitas_megj`, `brutto_vegosszeg`, `netto_vegosszeg`)  
    VALUES ('NULL', '$fh_id', '$teljes_nev', '$cim', '$fizetesi_modok_id', '$rendelés_statusza_id', '$rendelés_datumido',  
        $szallitas_datum', '$megjegyzes', '$vegosszeg_brutto', '$vegosszeg_netto')");  
$eredmeny_rendelés_töltés = $rendelés_töltés->fetch(PDO::FETCH_ASSOC);
```

A(z) localhost:8090 közlendője

Rendelését sikeresen rögzítettük!

OK

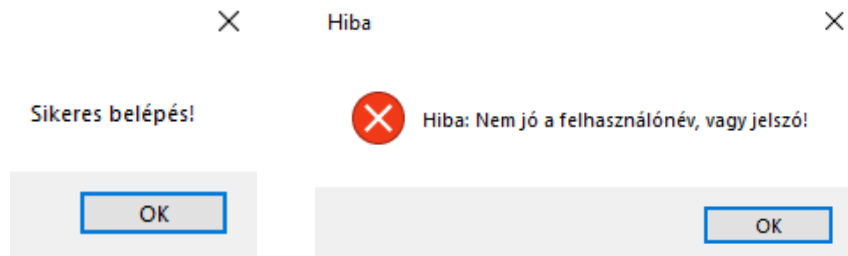
1.6.4.3 Adatok lekérdezése

Az adatbázisban lévő, a felhasználó számára minden lényeges adat lekérdezésre kerül a főoldal megnyitásakor, és php tömbökben kerül eltárolásra (pl. termékek, termék kategóriák stb..). Ennek segítségével jelenítjük meg a termékeket, képeket, menüpontokat az oldalon.

1.6.5 Admin applikáció tesztelése

1.6.5.1 Bejelentkezés

Visszajelzés sikeres és sikertelen belépés esetén:



1.6.5.2 Adatbázis

```
private void Connect()
{
    try
    {
        MyINI myINI = new MyINI("connect.ini");
        string host = myINI.ReadString("CONNECT", "HOST", "localhost");
        string user = myINI.ReadString("CONNECT", "USER", "root");
        string pwd = myINI.ReadString("CONNECT", "PWD", "");
        string database = myINI.ReadString("CONNECT", "DATABASE", "frissfutar");
        db = new MyDB(host, user, pwd, database);
    }
    catch (Exception err)
    {
        MessageBox.Show("Nem sikerült csatlakozni az adatbázishoz! Az alkalmazás bezáródik!\n" +
            err.Message, "Hiba", MessageBoxButtons.OK, MessageBoxIcon.Error);
        Application.Exit();
    }
}
```

Az adatbázishoz való kapcsolódás a connect.ini fájlban tárolt adatok segítségével történik, ami egyszerűbbé teszi a jövőben esedékes változtatások beállítását.

1.6.5.3 Adatok feltöltése adatbázisba

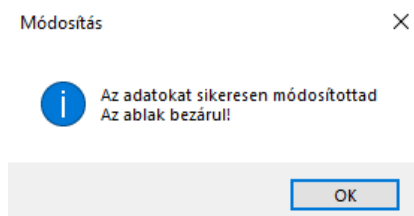
```
string id = "NULL";
string megn = this.ujTermekKategoriaFrm.TKMegnTb.Text;
string[,] feltoltendo_adatok =
{
    {"@id", id },
    {"@megn", megn },
};
string sql = "INSERT INTO `termek_kategoria` (`id`, `megnevezes`) VALUES(@id, @megn);";
this.DB.Query(sql, feltoltendo_adatok);
MessageBox.Show("Az adatokat sikeresen feltöltötted", "Feltöltés"
    , MessageBoxButtons.OK, MessageBoxIcon.Information);
}
catch (Exception err)
{
    MessageBox.Show("Adatokat nem sikerült eltárolni!\nHiba: " +
        err.Message, "Hiba", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
```

Az adatokat kétdimenziós tömbökben tárolunk, amit később az SQL kódban felhasználunk az adatbázis feltöltéséhez.

1.6.5.4 Módosítás

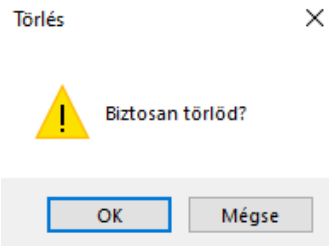


Ha vannak kötelezően kitöltendő mezők, akkor az alkalmazás hibaüzenetet küld és nem hajtja végre a változtatásokat.

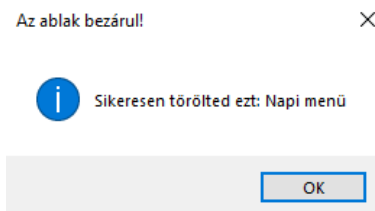


Sikeres módosítás esetén visszajelzést kap a felhasználó a programtól.

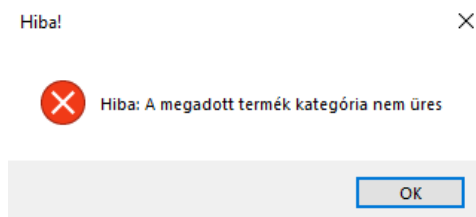
1.6.5.5 Törlés



A program megerősítést kér a törlésről, ezzel elkerülve a véletlen történő adatvesztést.

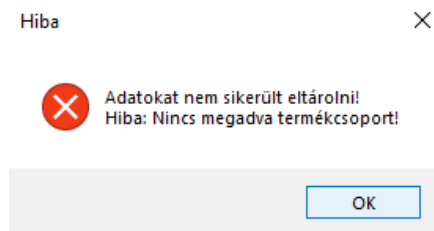


Sikeres törlésnél visszajelzést küld.

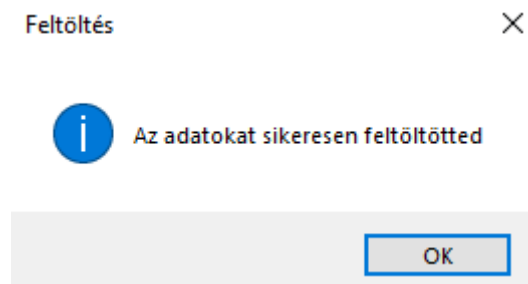


A program nem minden esetben enged törölni, ilyenkor hibaüzenetet kap az admin.

1.6.5.6 Új adatok



Üresen hagyott mezők esetén, az alkalmazás hibaüzenetet küld és nem tölti fel az új adatot az adatbázisba.



Sikeres feltöltés esetén azonban visszajelzést küld a program a felhasználónak a végbement adatfeltöltésről.

Az admin alkalmazást készítette: Turza Norbert és Kertész Kornél

Az admin alkalmazás tesztelése: Turza Norbert

1.7 Továbbfejlesztési lehetőségek

1.7.1 Keresés funkció megvalósítása

A főoldalon a bal felső keresés modullal lehessen keresni bizonyos ételre/italra, illetve típusra.

1.7.2 Fizetési módok kezelése

Az Adatok megadásánál rendeléskor miután a megfelelő fizetési mód kiválasztásra került, ennek függvényében jelenjen is meg a megfelelő online fizetési platform.

1.7.3 Rendelés visszaigazolás, email küldése

A rendelés leadásakor küldjön visszaigazoló emailt a felhasználónak a rendelés felvételéről és a várható kiszállítási időtartamról.

1.7.4 Rendelések megtekintése külön webes felületen (admin)

Egy független webes felület készítése, melyen a leadott rendelések tekinthetők meg, azoknak minden adatával együtt. Szűréssel, kereséssel, A futár számára szükséges információkat tartalmazza.

1.7.5 Adatbázis szerveren történő működtetése

Az adatbázis szerveren történő működtetése, annak érdekében, hogy a weboldal bárholonnan elérhető legyen forrásállományok és alkalmazások telepítése nélkül.

1.7.6 Bug-ok javítása, optimalizálás, karbantartás

További tesztelések elvégzése, jelenleg ismert, és a jövőben tudomásunkra jutó hibák javítása, egyéb optimalizációk elvégzése, folyamatos karbantartás.

1.7.7 Kiemelt ajánlatok kezelése

A főoldal elején megjelenő kiemelt ajánlatok konfigurálhatók legyenek, az admin beállíthatja az aktuális ajánlatokat, amelyek megjelennek az oldal tetején.

1.7.8 Admin alkalmazás optimalizálása

Az admin applikáció optimalizálása, az adatbázis gyorsabb és hatékonyabb karbantartása érdekében.

2 Felhasználói dokumentáció

2.1 Ismertetés

A Frissfutár egy étel - és italrendelésre használatos weboldal, ami számítógépen, tableten, telefonon egyaránt elérhető bárki számára, aki rendelkezik internetkapcsolattal.

2.2 Rendszerkövetelmények

Minimum rendszerkövetelmények:

Telefon/Tablet: Android 8.0, iOS 12

Stabil internetkapcsolat

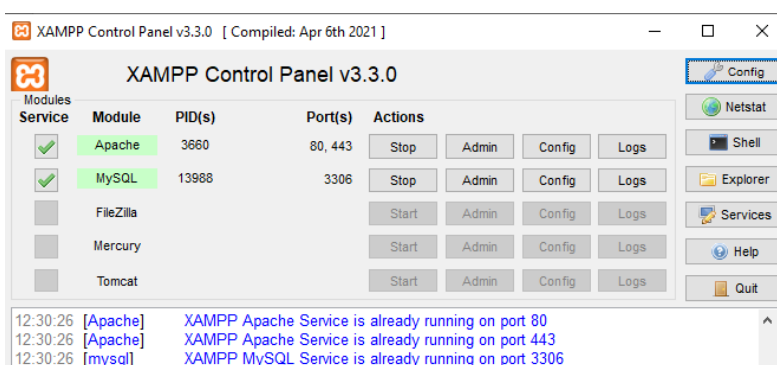
Ajánlott operációs rendszer:

Windows 10, Windows 11

2.3 Alkalmazás telepítése, beállítása

A weboldalunk használatához, a xampp telepítése után (mellékletben található link) a böngészőben a következő link: <https://github.com/fedorbenjamin02/Vizsgaremek2024> bemásolásával a nyilvános github oldalunkon keresse meg a zöld háttérű „Code” lenyíló menüt, majd kattintson rá. Ezután kattintson a „Download Zip” gombra, majd az előugró ablakban navigáljon el a C:/xampp/htdocs/ mappájába és mentse ide a zip fájlt.

Következő lépésként csomagolja ki a Vizsgaremek2024-main.zip-et a jelenlegi mappába (C:/xampp/htdocs/). Miután megvan indítsa el a xampp-ot, a xampp-on belül pedig indítsa el az Apache-ot és a MySQL-t a mellettük található Start gombbal!



A böngésző keresőjébe a localhost/ Vizsgaremek2024-main/app/ -ot beírva éri el a weboldalt.

2.4 Alkalmazás használata

2.4.1 Webes alkalmazás használata

2.4.1.1 Főoldal

A bal felső sarokban található oldalsó menü előhívó és eltüntető gomb.

Keresés modul a bal felső sarokban.

A jobb felső kosár gomb megnyitja a kosár előnézetét.

A „Kosár ürítése” gomb törli a kosár teljes tartalmát.

A „Tovább a rendeléshez” gomb átirányít a kosár oldalra.

A „Fiók” menüpont alatt található

Mindkét menüpont a bejelentkezés/regisztráció oldalra irányít át.

Az „Étlap” menüpont alatt található

„Ételek” menüpont megjeleníti az összes étel kategóriát.

„Italok” menüpont az ital típusokat jeleníti meg.

A bizonyos étel és ital kategóriákra kattintva az adott típusú ételhez vagy italhoz navigálja a felhasználót.

A „Kapcsolat” menüpont az elérhetőségi adatokhoz irányítja a felhasználót.

A „Kuponok” menüpont bejelentkezett felhasználók számára elérhető, és az akciós kuponokhoz navigál.

A kosár gomb mellett található fiók ikon bejelentkezett felhasználóknak jelenik meg, és az alábbi menüpontokat tartalmazza:

Kijelentkezés – A felhasználó kijelentkeztetése.

Adatok módosítása – Az adatmódosító aloldalra irányítja a felhasználót, ahol szállítási, illetve bejelentkezési adatainak módosítására van lehetősége.

A jobb alsó sarokban található felfelé mutató nyíl az oldal tetejére irányítja a felhasználót.

A termékekhez tartozó „Rendelés” gomb hozzáad a kosárhoz a választott termékből egyet.

Az oldalsó menü utolsó menüpontjával a sötét és világos téma közt lehet váltani, mely választott témát a későbbiekben a többi aloldal is alkalmazza.

2.4.1.2 Bejelentkezés és Regisztráció

Az oldal tetején található „x” ikonra kattintva a Főoldalra irányítja a felhasználót. A „Bejelentkezés” és „Regisztráció” gombokkal a regisztráció és a bejelentkezés űrlap között lehet váltani.

Regisztráció:



A „Tovább” gomb megfelelő adatok megadása esetén a regisztrációs űrlap következő oldalára lapoz.

A „Vissza” gomb az előző oldalra lapoz.

Az utolsó oldalon lévő „Regisztráció” gombbal tudja véglegesíteni a regisztrációját a felhasználó.

Bejelentkezés:



2.4.1.3 Kosár oldal



A „Vissza” gomb a Főoldalra irányítja a felhasználót.

A „Kosár ürítése” gomb törli a kosár tartalmát, és visszalépteti a felhasználót a Főoldalra.

A „Tovább” gomb a rendelési adatok megadására szolgáló oldalra irányít.

A kosárban lévő termékek mennyisége mellett található „ + ” és „ - ” gombok segítségével növelhető illetve csökkenthető a megrendelni kívánt termék mennyisége.

2.4.1.4 Rendelési adatok megadása

A rendeléshez szükséges adatok megadására szolgáló oldal.

Bejelentkezett felhasználónál az adatok automatikus kitöltésre kerülnek, azokat természetesen változtatni lehet.

2.4.1.5 Rendelés befejezése



Szállítási adatok		
Név	Teszt Elek	
Telefonszám	+36301234567	
Település	Vác	
Cím	Kossuth tér 2	

Rendelés tételei		
Megnevezés	Egységár	Mennyiség
Gyros tál	2650 Ft	1 db
Sajtleves	1590 Ft	2 db
Cordon Bleu	3220 Ft	1 db

Fizetési Mód: készpénz
Végösszeg: 9050 Ft

Vissza Rendelés

A rendelés leadására szolgáló oldal, melyen a felhasználó áttekintheti a rendelni kívánt tételeket, és minden fontos információt.

A „Vissza” gombbal a felhasználó visszaléphet az Adatok megadása oldalra.

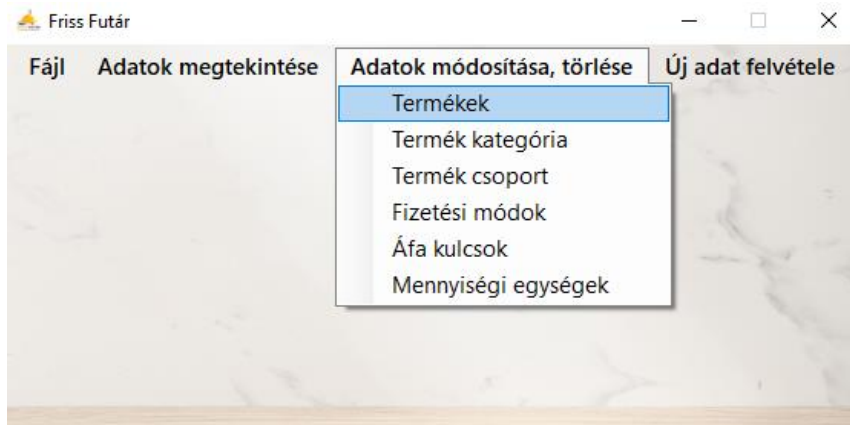
A „Rendelés” gombbal véglegesítheti rendelését.

2.4.1.6 Felhasználói adatok módosítása

A bejelentkezett felhasználónak lehetősége van megváltoztatni telefonszámát, e-mail címét, illetve szállítási címét. Az oldal a főoldalról nyitható meg, a jobb felső sarokban található fiók ikonra, majd a lenyíló menüben belül a beállítások menüpontot választva.



2.4.2 Asztali alkalmazás használata



A „Fájl” menüpont alatt található a „Kilépés”, amivel bezárul az applikáció.

Az „Adatok megtekintése” menüpont alatt megtekinthetjük az adatbázis összes táblájának az adatait egyenként.

Az „Adatok módosítása, törlése” segítségével módosíthatunk és törölhetünk a jövőben szükséges változtatásokat végezhetjük el az adatbázis bizonyos tábláin. Bármelyik lehetőségre kattintva megnyílik egy új form amin 3 gomb lesz elérhető:

„Módosít”:

A gomb megnyomásakor az előtte megadott adatok alapján egy következő formot nyit meg, ahol külön-külön változtathatók a tábla adatai, az „OK” gombbal elfogadja és feltölti az adatokat az adatbázisba, a „Mégse” gomb bezárja ezt a formot és az előző formra irányít.

„Töröl”:

A kiválasztott adatot törli az adatbázisból.

„Mégse”:

Bezárja ezt a formot és a főformra irányít.

Az „Új adat felvétele” a megadott táblákat új adatokkal tölti fel. Bármelyik menüpontra kattintva megnyit egy új formot amin 2 gomb található:

„Feltölt”: Az új adatokat feltölti az adatbázisba.

„Mégse”: Bezárja a formot és visszairányít a főformra.

3 Irodalomjegyzék

1.5 Internetes források

A vizsgaremek elkészítése során felhasznált források:

<https://stackoverflow.com/>

<https://www.php.net/>

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

<https://codepen.io/>

<https://www.geeksforgeeks.org/>

<https://playground.com/>

<https://uxplanet.org/>

<https://fontawesome.com/>

<https://getbootstrap.com/>

<https://learn.microsoft.com/>

www.javascripttutorial.net

<https://phppot.com/>

2. Mellékletek

2.1 Forrásállományok

A teljes vizsgaremek publikus github linkje:

<https://github.com/fedorbenjamin02/Vizsgaremek2024>

2.2 Adatbázis

Az adatbázis tartozó dump sql fájl linkje:

<https://github.com/fedorbenjamin02/Vizsgaremek2024/blob/main/Mellékletek/Adatbázis/rissfutar.sql>

Az üres, adatokat nem tartalmazó sql fájl:

https://github.com/fedorbenjamin02/Vizsgaremek2024/blob/main/Mellékletek/Adatbázis/rissfutar_ures.sql

2.3 Xampp

Az adatbázissal való kapcsolathoz szükséges a xampp alkalmazás, mely az alábbi linkről tölthető le:

<https://www.apachefriends.org/download.html>

2.4 Dokumentáció

A dokumentációt az alábbi linkekről töltheti le:

Dokumentáció PDF formátumban:

Dokumentáció WORD-ben:

<https://github.com/fedorbenjamin02/Vizsgaremek2024/blob/main/Mellékletek/Dokumentáció/Dokumentacio.docx>

2.5 Teszteléshez szükséges adatok

A teszteléshez szükséges belépési adatok egy txt fájlban:

https://github.com/fedorbenjamin02/Vizsgaremek2024/blob/main/Mellékletek/Tesztelés/belopesi_adatok.txt

