

BUDAPESTI MŰSZAKI SZAKKÉPZÉSI CENTRUM BOLYAI JÁNOS MŰSZAKI TECHNIKUM ÉS KOLLÉGIUM

Szakképesítés azonosító száma, megnevezése
5 0613 12 03 Szoftverfejlesztő és -tesztelő

Vizsgatevékenység:
Projektfeladat

Asztali és webes szoftverfejlesztés, adatbázis-készítés gyakorló feladatsor

A vizsgafeladat időtartama: 240 perc

Elérhető pontszám: 65 pont

Értékelés: a projektfeladat eredménye a vizsgaretek értékelésére kapott pontszámokkal
összevontan kerül osztályozásra.

2024

Készítették:
Dorogházi Donát
Szilasi István

1. Grafikus és konzolos részt egyaránt tartalmazó asztali alkalmazás fejlesztése

A feladat megoldása során vegye figyelembe a következőket:

- A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 3. feladat:)!
- Az egyes feladatokban a kiírásokat a minta szerint készítse el!
- Az ékezetmentes azonosítók és kiírások is elfogadottak.
- Az azonosítókat kis- és nagybetűkkel is kezdheti.
- A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.
- A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!
- A feladatokat C#-ban oldja meg!
- A feladatokhoz tartozó forrásokat az asztalon a **forrasok** mappában találja!

a. Konzolos rész: Kalapácsvetés

A kalapácsvetés az atlétika egyik versenyszáma. Minden idők legjobb férfi kalapácsvetőinek rangsorát a kalapacsvetes.txt szöveges állomány tartalmazza, a 2021. júniusi állapotnak megfelelően. A fájl UTF-8 kódolású és minden sorában egy sportoló adatai szerepelnek az alábbi sorrendben:

- helyezés,
- eredmény (méterben, két tizedesjegy pontossággal)
- sportoló neve,
- a sportoló országkódja,
- helyszín,
- dátum.

Az állomány első sora a mezőneveket tartalmazza, az adatokat pontosvessző választja el a mintának megfelelően:

Helyezés	Eredmény	Sportoló	Országkód	Helyszín	Dátum
1	86,74	Yuriy Sedykh	URS	Stuttgart	1986.08.30
2	86,04	Szergej Litvinov	URS	Drezda	1986.07.03
3	84,90	Vadim Devyatovskiy	BLR	Minszk	2005.07.21
4	84,86	Koji Murofushi	JPN	Prága	2003.06.29

Feladatok

1. Készítsen konzolos alkalmazást és mentse el a projektet vezeteknev_keresztnev/konzol mappába Kalapacsvetes néven!
2. Hozzon létre saját osztályt Sportolo azonosítóval, melynek adattagjait felhasználva egy-egy sportoló adatait tudja majd tárolni! Készítse el a Sportolo osztály konstruktorát, mely hívásával az osztály adattagjait tudja inicializálni!

3. Olvassa be a kalapacsvetes.txt állomány sorait és tárolja az adatokat egy Sportolo osztályon alapuló összetett adatszerkezetben! Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza!
4. Határozza meg és írja ki, hány dobás eredménye található a forrásfájlban!
5. Határozza meg és jelenítse meg a forrásállományban szereplő magyar (HUN) sportolók dobásainak átlageredményét! Az eredményt két tizedesre kerekítve írja ki!
6. Kérjen be egy évszámot és írja ki, hogy abban az évben mennyi dobás került be a legjobbak közé, illetve írja ki, hogy mely sportolók érték el ezeket. Ellenkező esetben írja ki, hogy az adott évben nem került be egy dobás eredménye sem a legjobbak közé.
7. Készítsen statisztikát, hogy melyik országból hány kalapácsvetés eredménye szerepel a legjobb dobások között. Az eredményt a mintának megfelelően írassa ki a képernyő-re!
8. Hozzon létre magyarok.txt néven egy UTF-8 kódolású fájlt, amelyben csak a magyar (HUN)sportolók eredményei szerepelnek.

Képernyőkép:

```
4. feladat: 25 dobás eredménye található.
5. feladat: A magyar sportolók átlagosan 83,39 métert dobtak.
6. feladat: Adjon meg egy évszámot:
1998
    3 darab dobás került be ebben az évben.
    Tibor Gécsek
    Balázs Kiss
    Szergej Kirmaszov
7. feladat: Statisztika
    URS - 5 dobás
    BLR - 3 dobás
    JPN - 1 dobás
    HUN - 4 dobás
    POL - 2 dobás
    RUS - 2 dobás
    RPG - 1 dobás
    FIN - 1 dobás
    DEU - 2 dobás
    USA - 1 dobás
    NDK - 1 dobás
    UKR - 1 dobás
    SVN - 1 dobás
```

Képernyőkép, ha a megadott évben nem került be egy dobás sem:

4. feladat: 25 dobás eredménye található.
5. feladat: A magyar sportolók átlagosan 83,39 métert dobtak.
6. feladat: Adjon meg egy évszámot:
2010
Egy dobás sem került be ebben az évben.
7. feladat: Statisztika

URS	-	5	dobás
BLR	-	3	dobás
JPN	-	1	dobás
HUN	-	4	dobás
POL	-	2	dobás
RUS	-	2	dobás
RPG	-	1	dobás
FIN	-	1	dobás
DEU	-	2	dobás
USA	-	1	dobás
NDK	-	1	dobás
UKR	-	1	dobás
SVN	-	1	dobás

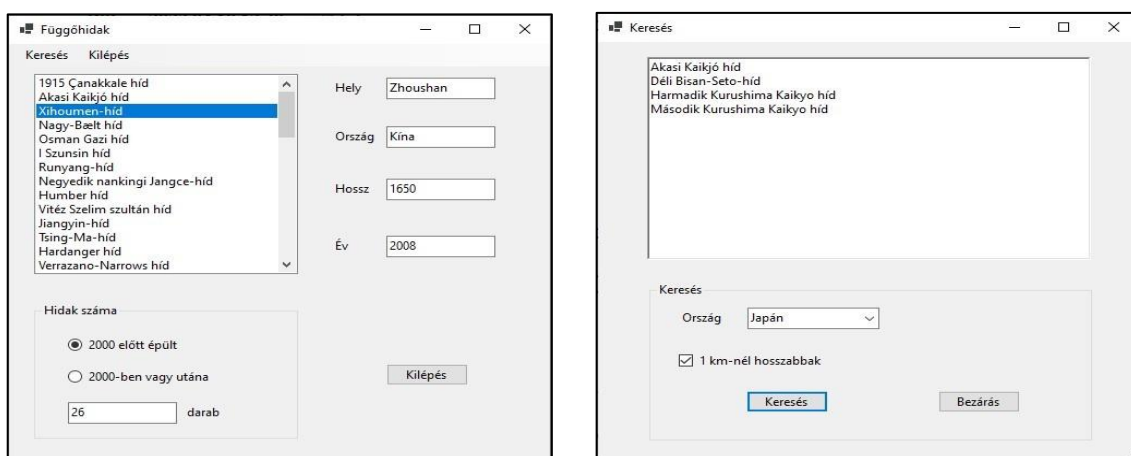
b. Grafikus rész: Függőhidak

Készítsen grafikus, két formából álló, menüvezérelt alkalmazást a mintának megfelelően a függőhidak adatainak megjelenítésére és kezelésére. A legnagyobb támaszközű függőhidak adatbázisát a függohidak.sql nevű, UTF-8 kódolású fájl tartalmazza.

A forrásfájl sorai egy-egy híd adatait tartalmazzák a következő sorrendben:

- helyezés, hosszúság szerint csökkenő sorrendben,
- a híd neve,
- földrajzi hely, ahol a híd található,
- ország,
- a híd hossza (támaszközök közti távolsága) méterben,
- átadás éve.

Az állomány első sora a mezőneveket tartalmazza, az adatokat tabulátor választja el egymástól.



Feladatok:

1. Hozzon létre saját osztályt Függohid azonosítóval, melynek adattagjait felhasználva egy-egy híd adatait tudja majd tárolni és készítse el a Függohid osztály konstruktorát, mely hívásával az osztály adattagjait tudja inicializálni! A projektet vezeteknev_keresztnev/grafikus mappába, **Függohidak** néven mentse el! (2 pont)
2. Az adatokat olvassa be adatbázisból. Az adatbázist hozza létre a forrasok/grafikus/fuggohidak.sql fájl segítségével. Az adatokat egy Függohid osztályon alapuló összetett adatszerkezetben kezelje! (2 pont)
3. A függőhidak nevei egy ListBox-ban jelenjenek meg, a függőhidak további adatai szövegdobozokban jelenjenek meg, amikor kiválasztunk egy hidat a listából. (1 pont)
4. Rádiógombok segítségével jelenítse meg a 2000 előtt, illetve 2000-ben és utána épült hidak számát! (1 pont)
5. Hozza létre a főmenüben a Keresés és a Kilépés menüpontokat, a Kilépés menüpont és a Kilépés gomb hatására záródjon be az alkalmazás! (1 pont)
6. A Keresés menüpont hatására jelenjen meg a keresés form, a főformot rejtse el, csak a bezárás gombra kattintva jelenjen meg újra! (1 pont)
7. A keresés form megjelenésekor a legördülő lista (ComboBox) elemeit dinamikusan töltse fel a forrásfájlban szereplő országok neveivel! (1 pont)
8. A Keresés gomb hatására a megadott feltételeknek megfelelő függőhidak nevei jelenjenek meg a szövegdobozban. (1 pont)

2. *Reszponzív viselkedésű weboldal*

Készítse el a leírásnak megfelelő rezponzív viselkedésű oldalakat, a megjelenített minták csak segítséget nyújtanak nem feltétlen tükrözik a valódi megjelenést. Ezért leírásban levő utasításokat kell követni ezt értékeljük! A feladat megoldásnak fájljait vezeteknev_keresztnev/web mappába mentse.

A feladatban elkészített elemeket felhasználhatja 4. vizsgarészben. A megoldásnál használhat Bootstrap-et. A CSS beállításokat külön fájlba mentse!

Főoldal:

- Készítse el a fő oldalt index.html néven és a hozzá tartozó stílus lapot indexstyle.css néven. Forrásokban mellékelt kisallat.jpg képet állítsa be háttérként. Igazítsa a képet középre függőlegesen és vízszintesen, a kép ne ismétlődjön.
- Készítsen két nyomógombot (más html elemet is használhat), amely megjeleníti a Bejelentkezés és a regisztrációs oldalakat. Ha a nyomógomb fölé viszem az egérmutatót változzon meg a háttérszíne a gomboknak és a szöveg váljon félkövérré. A gombok mérete ne változzon meg a szélek legyenek lekerekítve. A gombok kék színűek legyenek a szövegszín pedig fehér.
- A regisztráció gomb balra igazítva, a Bejelentkezés gomb jobbra igazítva jelenjen meg egy sorban az oldal alján az oldalak szélétől 50 pixel távolságra . Amennyiben az oldal szélessége lecsökken 400 pixel alá, akkor a gombok egymás alatt jelenjenek meg vízszintesen és függőlegesen is középen. A nyomógombok távolsága egymástól 30 pixel legyen. A háttér szélessége is igazodjon az oldal méretéhez a kép tartsa meg a méretarányát. A regisztráció gombra kattintva a signup.html oldal jelenjen meg. A bejelentkezés gombra kattintva pedig a login.html oldal.



Bejelentkezés oldal:

- Készítse el a bejelentkezés oldalt login.html néven és a hozzá tartozó stílus lapot loginstyle.css néven. Alakítsa ki mintán látható szerkezetet az oldal háttérszíne sötétebb kék legyen ne egyezzen meg a gombok színével.
- A Bejelentkezés felírat H1 es fejléc legyen a szöveg színe fehér és középen jelenjen meg. A nyomógomb alul jobboldalt helyezkedjen el. Rendelkezzen a nyomógomb azokkal az általános beállításokkal, ami a főoldalon található gombokra érvényes.
- Helyezze el a beviteli mezőket a hozzátartozó feliratokat a minta szerint. A beviteli mezőket tartalmazó doboz háttérszíne térjen el az oldal háttérszínétől legyen világosabb kék. a mérete maximum 350px, minimum pedig 250px legyen. A szövegszín pedig fekete. A beviteli mezők háttérszíne egyezzen meg a doboz háttér színével. A beviteli mezők szélessége legyen kisebb a doboz méretétől és középen jelenjen meg.

Bejelentkezés

E-mail cím

E-mail

Jelszó

Jelszó

Bejelentkezés

Regisztráció oldal:

- Készítse el a regisztrációs oldalt signup.html néven és a hozzá tartozó stílus lapot signupstyle.css néven. Alakítsa ki mintán látható szerkezetet az oldal háttérszíne sötétebb kék legyen ne egyezzen meg a gombok színével.
- A regisztráció felírat H1 es fejléc legyen a szöveg színe fehér és középen jelenjen meg. A nyomógomb alul jobboldalt helyezkedjen el. Rendelkezzen a nyomógomb azokkal az általános beállításokkal, ami a főoldalon található gombokra érvényes.
- Helyezze el a beviteli mezőket a hozzátartozó feliratokat a minta szerint.
- A beviteli mezőket tartalmazó doboz háttérszíne térjen el az oldal háttérszínétől legyen világosabb kék. a mérete maximum 350px, minimum pedig 250px legyen. A szövegszín pedig fekete. A beviteli mezők háttérszíne egyezzen meg a doboz háttér színével. A beviteli mezők szélessége legyen kisebb a doboz méretétől és középen jelenjen meg.

The image shows a registration form titled "Regisztráció" (Registration) in white text on a dark blue background. The form is contained within a light blue rectangular box. It features the following elements:

- Név** (Name): A text input field with a light blue placeholder "Név".
- Cím** (Address): A text input field with a light blue placeholder "Cím".
- E-mail cím** (Email address): A text input field with a light blue placeholder "E-mail".
- Jelszó** (Password): A text input field with a light blue placeholder "Jelszó".
- Jelszó újra** (Repeat password): A text input field with a light blue placeholder "Jelszó újra".
- Regisztráció** (Registration): A dark blue button with white text, located at the bottom right of the form.

Termékek listája oldal:

- Készítse el a termékek listája oldalt a mellékelt minta segítségével products.html néven és a hozzá tartozó stílus lapot productsstyle.css néven. A termékek adatait pet.js fájl tartalmazza. Ezeket a termékeket jelenítse meg.
- Helyezze el a mintában szereplő nyomógombokat, A rendelések gombra kattintva nem kell történnie semminek. A kijelentkezés gombra kattintva navigáljon vissza a fő oldalra (index.html). A nyomógombok a formai megjelenése egységesen kövesse a főoldalon is meghatározott formátumot.
- A „Kisállatok listája” felirat vízszintesen középen jelenjen meg. A növekvő csökkenő feliratokat a mintának megfelelően helyezze el, az aktuálisan kiválasztott rendezés az fékvővén jelenjen meg. (a termékek rendezését nem kell megvalósítani).
- A termékeket a mintához hasonlóan jelenítse meg. A termékeket megjelenítő dobozok mindig ugyanolyan méretűek legyenek és a képek is a termékekről. A méreteket Ön határozhatja meg. A szövegek a mintának megfelelően jelenjenek meg, a kosár kép pedig középen. A kép felirata amennyiben fölé visszük az egérmutatót „megrendel” legyen.

Rendeléseim

Kijelentkezés

Kisállatok listája

növekvő/csökkenő



Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db



3. Backend vizsgafeladat rész

A feladatot vezetéknev_keresztnev/backend mappába mentse. Készítsen SQL mentést is webshop.sql néven ebbe a mappába.

Készítsen adatbázist a következők szerint:

Az adatbázist MySQL használatával készítse, Docker lemezképet vagy XAMP-ot használjon az adatbázis tárolásához.

Adatbázis neve: kisállat

Hozza létre az alábbi táblákat:

Tábla neve: **user**

- **uid** a felhasználó azonosítója, egész (auto increment), ez az elsődleges kulcs.
- **name** felhasználó neve, max 50 karakter, kötelező.
- **email** a felhasználó email címe max 80 karakter, kötelező, egyedi legyen.
- **password** a felhasználó jelszava a választott titkosítási módnak megfelelő típus
- **address** A felhasználó címe, szöveges, maximum 80 karakter, kötelező.

Tábla neve: **pet**

- **id** A kisállat azonosítója, egész (auto increment), ez az elsődleges kulcs.
- **name** A kisállat neve, szöveges, max 30 karakter, kötelező, legyen egyedi.
- **description** A kisállat részletes leírása, szöveges, max. 100 karakter, kötelező.
- **pries** A kisállat ára szám numerikum, kötelező.
- **stock** A kisállatból rendelkezésre álló mennyiség, egész, kötelező.
- **picture** Kép a kisállatról, szöveges, max 100 karakter nem kötelező

Tábla neve: **orders**

- **orderid** A megrendelés azonosítója, egész (auto increment), ez az elsődleges kulcs.
- **userid** A vásárló azonosítója a user táblából. (idegen kulcs)
- **date** A megrendelés dátuma, dátum típusú, kötelező. Amennyiben az érték nincs kitöltve az aktuális dátum értékét vegye fel.

- Hozza létre a táblák közötti kapcsolatokat! Készítsen Triggert Insert_User, ami megvalósítja, hogy a beírt jelszó titkosítva tagolódjon, amikor egy új felhasználó kerül rögzítésre.
- Készítsen függvényt, ami visszaadja, hogy a felhasználó be van-e jelentkezve vagy nincs. paraméterként a az email-t és a jelszót adja meg. (Ne feledje a jelszó titkosítva van tárolva!)
- Módosítsa úgy az adatbázist (adjon hozzá táblát), hogy egyértelmű legyen, hogy egy adott megrendeléshez milyen tételek tartoznak 3NF-ben legyen.

Töltse fel a **user** táblát a következő adatokkal.

name	address	email	password
Mekk Elek	1112 Budapest Varjú u. 12	mekk.elek@bolyai.hu	Titok123
Gipsz Jakab	1024 Budapest Gipsz u. 23	gipsz.jakab@bolyai.hu	Titok123

Töltse fel a **pet** táblát a következő adatokkal.

Name	Description	Pries	Stock	Picture
Hörcsög	Ez meg rágcál	10500	5	./pictures/horcsog.jpg
Tengeri malac	Tengeri röfi	6300	8	./pictures/tmalac.jpg
Madárpók	Veszélyes madárpók	80000	3	./pictures/madarpok.jpg
Aranyhal	Szép aranyhal	3500	1	./pictures/aranyhal.jpg

Készítsen backend szerver alkalmazást RESTApi használatával!

1. Valósítsa meg a bejelentkezést alábbiak szerint, minden válasz és törzsadat JSON formátumban kerüljön megadásra. JWTToken használjon! A generált token tartalmazza a bejelentkezett felhasználó id-ét.

Metódus	URL	Body	Válasz
GET	api/login	JSON	JSON/szöveges

Body: [
 {
 „email” : „mekk.elek@bolyai.hu”,
 „password: „Titok123”,
 },
]

Válasz: [
 {
 „token” : „A generált token”,
 },
]

Ha nem megfelelő a felhasználónév vagy jelszó, akkor 401 NOT AUTHORIZED státuszkóddal és az „Hibás felhasználónév vagy jelszó!” üzenettel térjen vissza.

2. Valósítsa meg az **user** tábla karbantartását az alábbiak szerint, minden válasz és törzsadat JSON formátumban kerüljön megadásra!

Kérje le a bejelentkezett felhasználó adatát az alábbiak szerint! A token a kérés fejlécében (header) küldje el.

Metódus	URL	Body	Válasz
GET	api/user	JSON	JSON/szöveges

Válasz:
 {
 „id” : 1,
 „name” : „Mekk Elek”,
 „address: „1112 Budapest Varjú u. 12.”,
 „email”: „mekk.elek@bolyai.hu”
 }

Ha nem érvényes vagy nincs token, akkor 401 NOT AUTHORIZED státuszkóddal és az „Azonosítás nem sikerült!” hibaüzenettel térjen vissza.

3. Valósítsa meg egy új felhasználó adatainak a rögzítését, az alábbiak szerint!

Metódus	URL	Body	Válasz
POST	api/signup	JSON	JSON/szöveges

Body: [
 {
 „name” : „Bokor Endre”,
 „address: „1212 Budapest Morgó u. 24.”,
 „email”: „bokor.endre@bolyai.hu”
 },
]

Válasz 201 CREATED státuszkóddal:

```
{
  „id” : 3
}
```

4. Valósítsa meg egy adott felhasználó adatainak módosítását, az alábbiak szerint! A módosítást csak a bejelentkezett felhasználó a saját adatain végezheti el. A token-t a kérés fejlécében (header) küldje el.

Metódus	URL	Body	Válasz
PUT	api/user	JSON	JSON/szöveges

Lehetséges hasznos üzenet:

Body:

```
{
  „name” : „Mekk Elek”,
  „address” : „1112 Budapest Varjú u. 12.”,
  „email” : „mekk.elek@bolyai-szakkozep.hu”
},
vagy
{
  „email” : „mekk.elek@bolyai-szakkozep.hu”
},
```

Válasz: 201 Státuszkód „Sikeres módosítás” üzenet.

Ha nem található a módosítandó elem, akkor 404 státuszkód NOT FOUND „Az elem nem létezik” hibaüzenettel térjen vissza. Amennyiben a törzs üres vagy nem stimmel a mezőnév, akkor a 400 BAD REQUEST státuszkóddal és „Az adatok nem megfelelőek!” hibaüzenettel térjen vissza. Amennyiben nem érvényes vagy nincs token, akkor 401 NOT AUTHORIZED státuszkóddal és az „Azonosítás nem sikerült!” hibaüzenettel térjen vissza.

5. Valósítsa meg vevő megrendelésének rögzítését az alábbiak szerint! Ez a végpont csak a felhasználó számára generált token-nel hívható meg. A token-t a kérés fejlécében (header) adja meg!

Metódus	URL	Body	Válasz
POST	api/order	JSON	JSON/szöveges

Body: [

```
{
  „userid” : 1,
  „date” : „2024.05.12”,
  „pet” : [{ „id” : 2, „quantity” : 3 }, { „id” : 3, „quantity” : 2 } ],
},
]
```

Válasz 201 CREATED státuszkóddal:

```
{
  „id” : 1
}
```

Ha nem érvényes vagy nincs token, akkor 401 NOT AUTHORIZED státuszkóddal és az „Azonosítás nem sikerült!” hibaüzenettel térjen vissza. Egyéb hiba esetén a 400 BAD REQUEST státuszkóddal és „Hiba az adatok rögzítésekor!” hibaüzenettel térjen vissza.

6. Valósítsa meg a kisállatok listájának lekérését! Ez a végpont csak a felhasználó számára generált token-nel hajtható végre. A token-t a kérés fejlécében (header) adja meg

Metódus	URL	Body	Válasz
GET	api/pet vagy api/pet/:id	JSON	JSON/szöveges

Válasz: [

```
{  
  „id” : 1,  
  „name” : „kiskutya”,  
  „description”: „Ez egy kiskutya”  
  „preis”: „50000”  
  „stock”: 1  
  „picture”: „./pictures/kutya.jpg”  
},  
]
```

Ha nem érvényes vagy nincs token, akkor 401 NOT AUTHORIZED státuszkóddal és az „Azonosítás nem sikerült!” hibaüzenettel térjen vissza. Ha nem található a termék, akkor 404 státuszkód NOT FOUND „Az elem nem létezik” hibaüzenettel.

7. Tesztelje a végpontok helyességét POSTMAN program segítségével, és mentse le az eredményt a feladatrészben megadott mappába **testpostman** néven!

Ügyeljen az adatok mentésére! A feladat megoldásához NodeJs-t vagy Laravelt használhat!!!!
Natív PHP vagy JS nem használható!!

Amennyiben nem megfelelő az adat vagy nem létezik a termék, akkor a végpontok a 404-es hibával térjenek vissza, ha a feladat nem ad utasítást erre vonatkozóan.

4. Frontend vizsgafeladat rész

A feladatot vezeteknev_keresztnev/frontend mappába mentse. A megvalósításhoz Vue, React, Angular framework-öt használhat natív JS nem megengedett. Használja a Bootstrap-et.

A Frontend alkalmazás is reszponzív legyen! Használja a vizsgafeladatsor 2. és 3. részének megoldásait ennek a vizsgarésznek a megoldásában. Amennyiben nem tudta elkészíteni a 3. vizsgarészben levő feladatokat, akkor használja az ott leírt RESTapi végpont információkat a megoldáshoz mintha egy ilyen végpontokkal rendelkező szerverhez kapcsolódna.

1. Főoldal, kezdőlap:

- Az alkalmazáshoz készítsen egy kezdőlapot, aminek a kinézete megegyezik 2. vizsgarész megoldásával. (a minta itt alább is látható)
- A bejelentkezés gombra kattintva, jelenjen meg a login oldal!
- A regisztráció feliratú gombra kattintva majd jelenjen meg a regisztrációs oldal, ahol majd regisztrálni lehet.
- Sikeres bejelentkezés esetén automatikusan lépjen át a kisállat oldalra. A kisállat oldalt csak bejelentkezett felhasználók láthatják. Megrendelést is csak bejelentkezett felhasználók adhatnak le.

Főoldal minta:



2. Bejelentkezés oldal:

Készítsen bejelentkezés oldalt, amely megvalósítja a bejelentkezést, aminek a kinézete megegyezik 2. vizsgarész megoldásával. (a minta itt alább is látható).

- Sikeres bejelentkezés esetén a termékek listája oldal jelenik meg. Amennyiben nem sikerül bejeltnkezni, akkor a „Sikertelen bejelenkezés” hibaüzenet jelenik meg felugró ablakban (Ne alert-et használj!).
- Sikeres bejelntkezés esetén tárolja le a felhasználóval kapcsolatos információkat a kliensoldalon az Ön által választott technológia segítségével. Így tudja használni a feladat további megoldásában.
- A 3 vizsgarész-ben definiált /api/login RESTApi végpontot használja!



A képernyőn a "Bejelentkezés" című felület látható. A felület kék háttérrel rendelkezik. A cím alatt egy vízszintes vonal van. Alatta egy világkék téglalap tartalmazza a bejelentkezési mezőket. Ezen belül az "E-mail cím" című mező mellett egy "E-mail" szövegű input mező található. Alatta a "Jelszó" című mező mellett egy "Jelszó" szövegű input mező látható. A mezők alatt még egy üres input mező van. A form alján egy kék gomb található, amelyen a "Bejelentkezés" szöveg olvasható.

3. Kisállatok listája minta oldal:

- Készítsen oldalt a kisállatok adatainak listázásához, aminek a kinézete megegyezik 2. vizsgarész megoldásával. (a minta itt alább is látható)
- A kisállatokat önálló komponensként hozza létre. a minta szerinti formában jelenjen meg egy sorban max. négy termék jelenhet meg a teljes méretű ablakban asztali alkalmazás esetén. Reszponzív viselkedés esetén kevesebb.
- Az adatokat 3. vizsgarészben készített backend alkalmazástól kérje le a „**api/products**” végpont segítségével.
- Készítsen egy nyomógombot az oldal tetején jobb oldalon, aminek a segítségével kijelentkezik és visszanyavigál a főoldalra.
- Valósítsa meg a kisállatok név szerinti rendezését növekvően és csökkenően a nyomógombokra kattintva a minta szerint. A rendezésnek megfelelő felirat legyen aktív.
- Amennyiben nincs kisállat rögzítve az adatbázisban a „A nincs kisállat az adatbázisban” szöveg jelenjen meg.
- A rendeléseim gombra kattintva megrendelések oldalt jelenítse meg. ahol láthatóvá válnak a megrendelések. Ha nincs megrendelése még a felhasználónak, akkor csak figyelmeztessen egy felugró ablak segítségével.
- Amennyiben a terméknél található kosárra kattint adj hozzá a terméket a megrendelésekhez, átnavigálva a megrendelések oldalra ahol módosíthatja a darabszámot.
- A felhasználó nem véglegesített megrendeléseit kliens oldalon legyenek tárolva, addig, amíg a felhasználó ki nem jelentkezik.

Rendeléseim

Kijelentkezés

Kisállatok listája

növekvő/csökkenő



Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db





Név: Kiskutya
Leírás:
Ez egy kutyuska
Ár: 35000Ft
raktáron: 10db



4. Megrendelés:

A megrendelés kliens oldalon van tárolva, az Ön által választott technológia segítségével. Kijelentkezéskor a kliensoldali rendelési adatok törlődnek. A megrendelés lehet egy önálló oldal, vagy egy felugró ablak is.

- Készítsen egy új oldalt a megrendelés adatainak rögzítéséhez a mellékelt minta segítségével. Az adatok mentéséhez a 3 vizsgarész-ben definiált /api/order RESTapi végpontot használja!
- A megrendelés rögzítésekor **nem kell** figyelembe vennie a raktáron levő mennyiséget.
- Valósítsa meg, hogy legyen lehetőség a kiválasztott termék mennyiségének változtatására és a termék törlésére a listából!
- A megrendelés gombra kattintva mentse le a megrendelést a backend alkalmazás megfelelő végpontját alkalmazva, ha hiba történt a rögzítéskor adjon tájékoztatást erről.
- Amennyiben a „vásárlás folytatása” feliratú gombra kattint, akkor térjen vissza a termékek listája oldalhoz.

Megrendelés minta:

Megrendelés

Kiválasztott kisállatok:

Kiskutya

Mennyiség

1

Ár: 25000

[Törlés](#)

Kiscica

Mennyiség

2

Ár: 19000

[Törlés](#)

Összesen:

63 000 Ft

Vásárlás folytatása

Megrendelés

5. Regisztráció:

Készítsen regisztrációs oldalt egy új felhasználó rögzítéséhez a regisztrációt bárki elvégezheti. sikeres regisztráció befejezésekor navigáljon a login oldalra.

- A korábban 3. feladatrésznél elkészített oldalt felhasználhatja a regisztráció elkészítéséhez.
- Ellenőrizze, hogy az adatok helyesen lettek kitöltve a ha nem akkor figyelmeztesse a felhasználót felugró ablak segítségével (ne alert-et használjon).
- A regisztráció gombra kattintva mentse el az adatokat az adatbázisban a backend alkalmazás megfelelő végpontját használva (/api/signup). Hiba esetén felugró ablakban jelenítse meg a hibát.

Regisztráció minta:

Regisztráció

Név

Név

Cím

Cím

E-mail cím

E-mail

Jelszó

Jelszó

Jelszó újra

Jelszó újra

Regisztráció