

ZÁRÓDOLGOZAT



Kecskeméti SZC
Kandó Kálmán
Technikum

CHRONUS – WEBOLDAL

Németh Áron

Tartalomjegyzék

Tartalom

Előszó - Projekt választás	4
Projektet segítő programok/nyelvek	5
HTML	5
XAMPP	5
PhotoshopCS6.....	5
PHP	5
JavaScript	6
Visual Studio Code	6
Chronus, logó:	7
Munkálatok közben:	7
Végeredmény:.....	7
Adatbázis felépítése:	8
Users tábla:	8
Admins tábla:	8
Wishlist tábla:	9
Cart tábla:.....	9
Products tábla:	9
Messages tábla:	10
Orders tábla:	10
Weboldal felépítése	11
Kezdő oldal (Főoldal, Header rész):	11
Regisztráció/bejelentkezés:	12
Regisztráció:	13
Bejelentkezés:	14
Jelszó változtatás:	17
Kezdő oldal (Rólunk):	18
Kezdő oldal (Rendelések, Termékek):.....	20
Márka választás, termékek megjelenése a főoldalon:	21
Áttekintés, kedvenc, kosár:	22
Termék keresése:	26

Rendelés:.....	27
Kezdő oldal (Kapcsolat):.....	29
Kezdő oldal (Footer rész):	30
Admin panel felépítése:	31
Admin panel (Irányítópult):.....	32
Admin panel (Termékek):	33
Admin panel (Rendelések):	35
Admin panel (Admin szerkesztés, törlés, létrehozás):.....	36
Admin regisztráció:.....	36
Szerkesztés/törlés:	37
Admin panel (Felhasználók):	38
Admin panel (Üzenetek):	39
Jövőben fejleszthető részek:	40
Felhasznált források:	41

Előszó - Projekt választás

A projektem kiválasztásánál több szempontot is szemügyre vettem. Elsőként a legfontosabbat, hogy melyik területen is tudnék igazán kiteljesedni, felhasználni az évek alatt tanultakat. Így a választásomra weboldal készítésre esett, hisz talán ez érdekelt a legjobban az évek alatt is. Miután ezt letudtam, sokáig törtem az agyam, hogy milyen weboldal kell létrehoznom. Beadandó dolgozatom egy órákkal foglalkozó weboldal. Eme döntést egy régi élmény miatt hoztam, érettségit elvégezve megleptem magam egy Casio órával és máig emlékszem, hogy alig tudtam eldönteni, hogy melyik oldalról rendeljem meg.

Weboldalnál elsődleges szempontként az minimalista stílust tartottam, könnyen átlátható legyen a felhasználónak, hisz elengedhetetlen, hogy a termék(eket) problémamentesen és gördülékenyen tudja kiválasztani, kosárba helyezni, majd megrendelni. Úgy gondolom mindig a vásárló szempontjai az első, de ez mellett fontos, hogy a fejlesztőnek is kedvezzünk. Hibákat, későbbi fejlesztéseket könnyedén, nagyobb problémák nélkül láthassa, bevezethesse, ezért egy egyszerű, gördülékeny stílusban került megírásra az oldal.

Tanulmányaim alatt mindig is a webfejlesztés volt az a témakör, amit jobban kedveltem, élveztem és élvezem is, hogy a kreativitás nem szab határt, bármilyen témában létrehozható egy-egy oldal. Programozási nyelve(i) szerintem könnyebben tanulható is, legalábbis számomra.

Weboldalt ihlette: chrono24.hu , timestore.hu.

Projektet segítő programok/nyelvek

HTML

A HTML (angolul: HyperText Markup Language, „hiperszoveges jelölőnyelv”) egy leíró nyelv, melyet weboldalak készítéséhez fejlesztettek ki, és mára már internetes szabvánnyá vált a W3C (World Wide Web Consortium) támogatásával.

XAMPP

A XAMPP – (kiejtése: /'zæmp/ vagy /'ɛks. æmp/) – egy szabad és nyílt forrású platformfüggetlen webszerver-szoftvercsomag, amelynek legfőbb alkotóelemei az Apache webszerver, a MariaDB (korábban a MySQL) adatbázis-kezelő, valamint a PHP és a Perl programozási nyelvek értelmezői (végrehajtó rendszerei).

PhotoshopCS6

A Photoshop ma a nyomdai előkészítés és képfeldolgozás legelterjedtebb programja, tudása miatt is a professzionális fotósok, web designerek, filmes utómunkával foglalkozó szakemberek megbecsült eszköze.

PHP

A PHP egy szerveroldali szkriptnyelv, mely segítségével dinamikus weblapokat készíthetünk. A PHP nyelven írt kódokat a webszerver PHP feldolgozó modulja értelmezi. A PHP egy olyan programozási nyelv, mely segítségével képesek lehetünk elkészíteni egy adatbázis alapú weboldalt is. Ilyenkor kerül képbe a szerver oldal, de erről majd egy kicsit később.

A PHP szó eredetileg a “Personal Home Page Tools” kifejezést takarta, amely arra utalt, hogy segítségével személyre szabhatjuk a weboldalakat, azonban ez a név később módosult a “Hypertext Preprocessor”-rá, amely már inkább arra utal, hogy ez a nyelv egy hypertext előfeldolgozó.

JavaScript

A JavaScript (röviden JS) egy programozási nyelv, amelyet kifejezetten az internetre fejlesztettek ki. A legtöbb böngésző szoftver, és a modern okostelefonok is mind támogatják a JavaScriptet.

A JavaScriptet első sorban arra használják, hogy gazdagabb, felhasználóbarát élményeket teremtsenek vele az internetet böngészők számára, például dinamikusan frissülő weboldalakot, intuitív felhasználói felületeket, menüket, párbeszédpaneleket, 2D-s és 3D-s grafikákat, interaktív térképeket, videólejátszókat, és számos egyéb elemet, illetve funkciót. A JavaScript ilyesfajta, webböngészőkben történő alkalmazását kliensoldalú JavaScriptnek szokás nevezni.

Visual Studio Code

A Visual Studio Code (rövidítve: VSCode vagy VS Code) ingyenes, nyílt forráskódú kódszerkesztő, melyet a Microsoft fejleszt Windows, Linux és OS X operációs rendszerekhez. Támogatja a hibakeresőket, valamint beépített Git támogatással rendelkezik, továbbá képes az intelligens kódkiegészítésre az IntelliSense segítségével. A VSCode-ban a felhasználók megváltoztathatják a kinézetet (témát), megváltoztathatják a szerkesztő gyorsbillentyű-kiosztását, az alapértelmezett beállításokat és még sok egyebet. Támogatja a kiegészítőket, melyek segítségével további funkciók, teszteszközök lehetőségei érhetőek el.



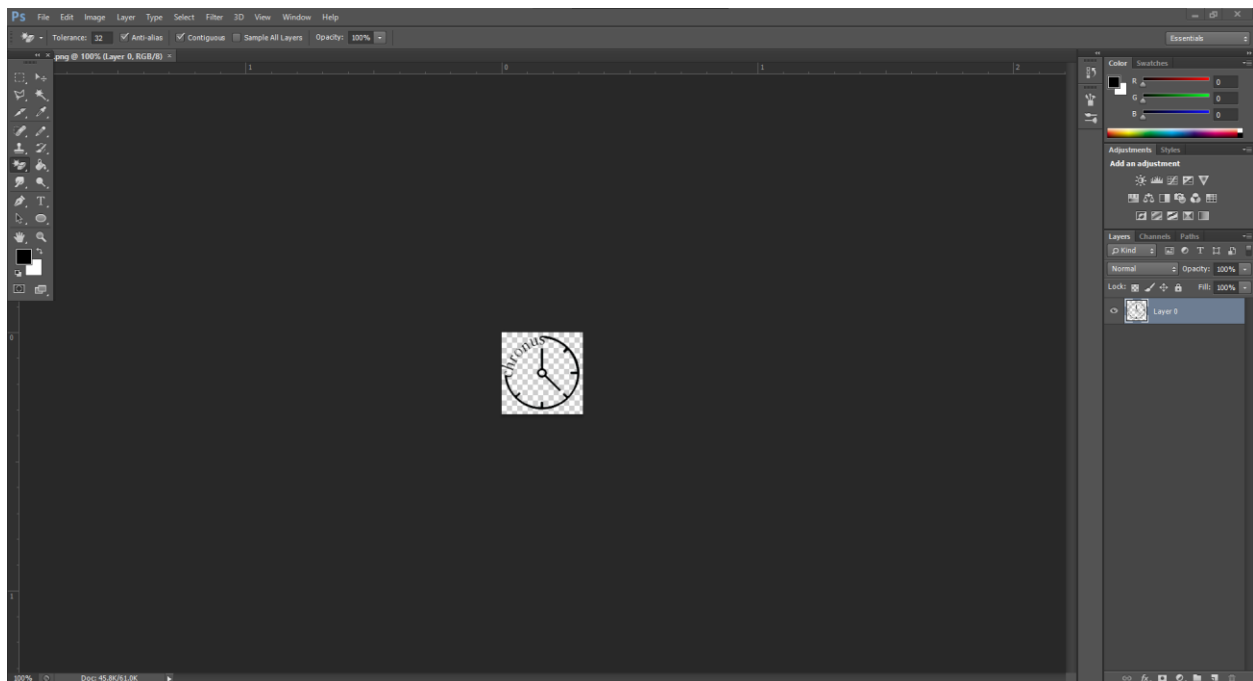
Visual Studio Code

Chronus, logó:

Weboldal kezdetleges változatánál fontolgattam, hogy milyen nevet is viseljen az egész, így elkezdtem különböző Istenekkel kapcsolatos nevek után keresgélni, míg megtaláltam a Kronosz névre hallgató istent, aki az idővel van összefüggésben. A választás rá esett, már csak egy egyszerű logót kellett terveznem az oldalhoz. Ehhez segítségül vettem a fentebb említett programok közül a Photoshop CS6 névre hallgató szoftvert.

Internetről letöltöttem egy háttér nélküli egyszerűen megrajzolt órát, majd pár egyszerű segédeszközzel kivágtam a bal felső sarkát és beillesztettem a oldalnak a nevét, Chronus-t.

Munkálatok közben:

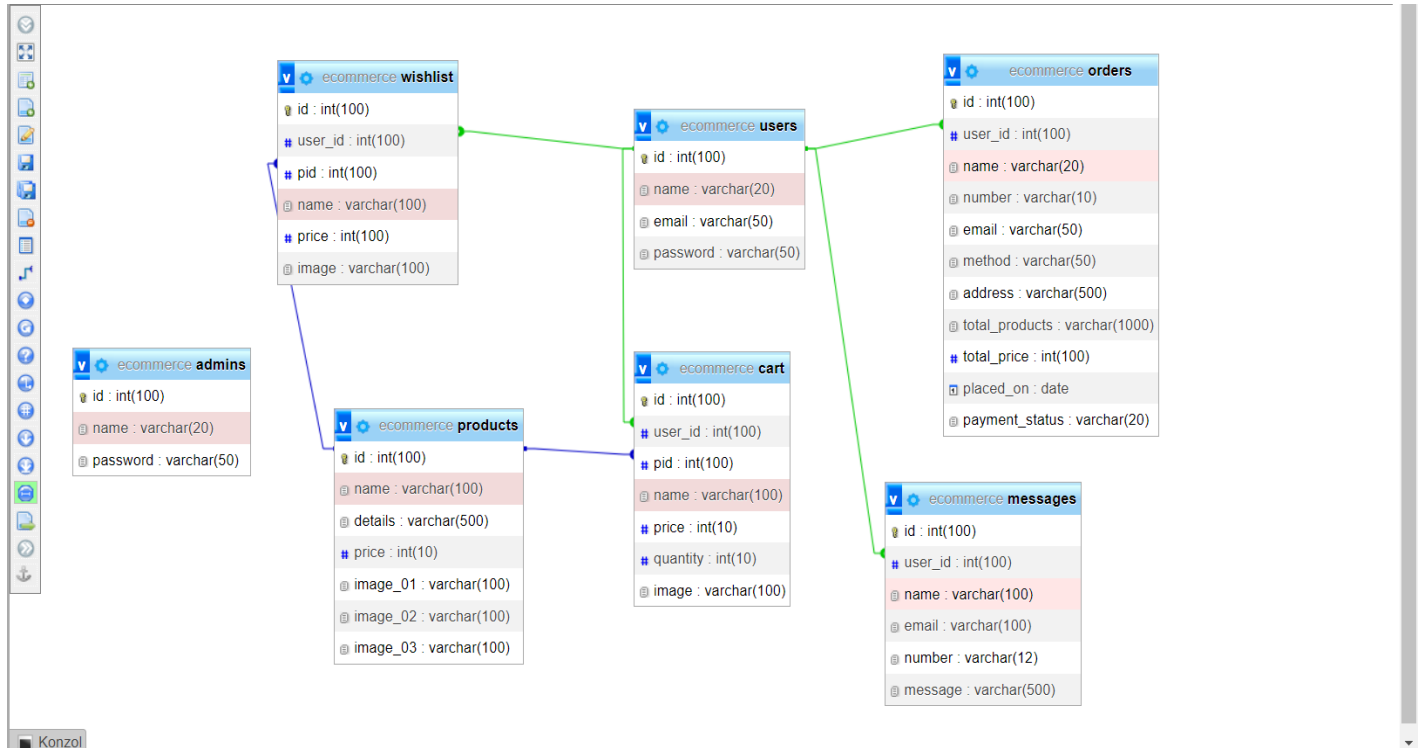


Végeredmény:



Adatbázis felépítése:

Egy adatbázist hoztam létre ecommerce néven, 7 táblával:



Users tábla:

A felhasználók eltárolására szolgál.

- **ID:** Felhasználók fiókjához tartozó azonosító (szám).
- **NAME:** Felhasználó által megadott név, amit használ a weboldalon.
- **E-MAIL:** Felhasználó által megadott e-mail, amit használ a weboldalon.
- **PASSWORD:** Felhasználó által megadott jelszó, amit használ a weboldalon.

Admins tábla:

Az adminok eltárolására szolgál.

- **ID:** Admin(ok) fiókjához tartozó azonosító (szám).
- **NAME:** Admin által létrehozott, fiókhoz tartozó felhasználói név.
- **PASSWORD:** Admin által létrehozott, fiókhoz tartozó jelszó.

Wishlist tábla:

A kedvencek/kívánságlista eltárolására szolgál.

- **ID:** Azonosító a létrehozott kedvencek menüponthoz.
- **USER_ID:** Azonosító, hogy a létrehozott kedvencek menüpontot tudjuk egy adott felhasználóhoz kötni.
- **PID:** Termékhez rendelt azonosító.
- **NAME:** Termékhez rendelt név.
- **PRICE:** Termékhez rendelt ár.
- **IMAGE:** Termék(ek)ről feltöltött képek, mely megjelenik a kedvencek menüpontban.

Cart tábla:

A kosár eltárolására szolgál.

- **ID:** Azonosító a kosár tartalmához.
- **USER_ID:** Azonosító, hogy a kosár tartalmát tudjuk egy adott felhasználóhoz kötni.
- **PID:** Termékhez rendelt azonosító.
- **NAME:** Termékhez rendelt név.
- **PRICE:** Termékhez rendelt ár.
- **QUANTITY:** Kosárba helyezett termék(ek) mennyisége.
- **IMAGE:** Termék(ek)ről feltöltött képek, mely megjelenik a kedvencek menüpontban.

Products tábla:

A termékek eltárolására szolgál.

- **ID:** Az adott termék azonosítója:
- **NAME:** Termék az admin által megadott neve:
- **DETAILS:** Termékről szóló leírás... (méret, anyag, kinézet, stb...).
- **PRICE:** Termék ára.
- **IMAGE_01, _02, _03:** Három darab képet vagyunk képesek feltölteni a termékről, érdemes több nézetből.

Messages tábla:

Az üzenetek eltárolására szolgál.

- **ID:** Üzenethez tartozó azonosító.
- **USER_ID:** Regisztrált felhasználóhoz társított azonosító.
- **NAME:** Üzenetet küldő felhasználó neve.
- **EMAIL:** Felhasználó által megadott e-mail cím.
- **NUMBER:** Felhasználó által megadott telefonszám.
- **MESSAGE:** Felhasználó által megfogalmazott üzenet.

Orders tábla:

Rendelések eltárolására szolgál.

- **ID:** Rendeléshez kapcsolt azonosító.
- **USER_ID:** Felhasználó azonosítója rendelésnél.
- **NAME:** Felhasználó által megadott név.
- **NUMBER:** Felhasználó által megadott telefonszám.
- **EMAIL:** Felhasználó által megadott e-mail cím.
- **METHOD:** Felhasználó által kiválasztott fizetési mód: (UTÁNVÉT, PAYPAL, HITELKÁRTYA)
- **ADDRESS:** Felhasználó által megadott szállítási cím (házszám, utca név, város, irányítószám)
- **TOTAL_PRODUCTS:** Felhasználó által rendelt termék(ek) mennyisége.
- **TOTAL_PRICE:** Rendelés teljes ára.
- **PLACED_ON:** Dátum, amikor a rendelés létre lett hozva.

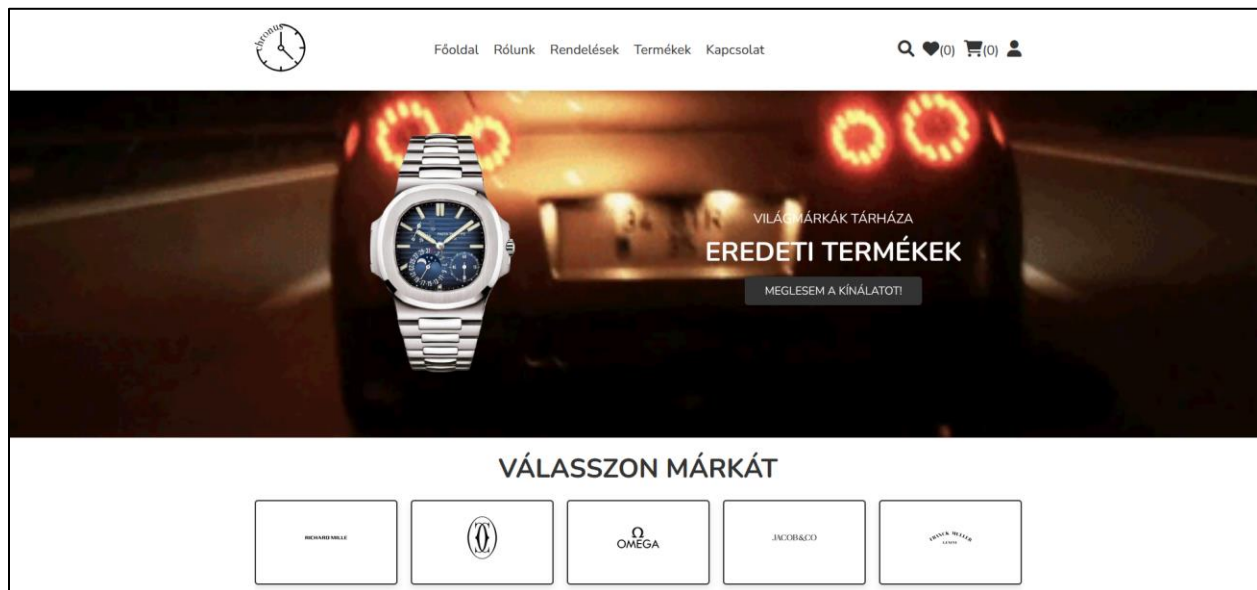
Weboldal felépítése

Kezdő oldal (Főoldal, Header rész):

A kezdő oldalra érkezve a felhasználónak van lehetősége már nézelődni, menüpontok között választani, termékeket keresgélni, de a főbb dolgokat nem tudja elkezdni, például terméket rendelni.

Ehhez előbb, regisztrálnia kell és / vagy bejelentkeznie. Amint ezek megtörténnek, utána a felhasználónak minden lehetősége megvan, amivel egy átlag felhasználó szert tehet.

Ha megnyitjuk az oldalt, a képen látható felület fog látszódni.



Bejelentkezést követően a felhasználót ugyanaz az oldal várja, mint regisztráció/bejelentkezés előtt, csak már tud bizonyos termékeket a saját kosarába/kedvencekhez helyezni.

A feltüntetett menüpontok között lehet mászkálni, de ha a felhasználó visszatérne a kezdő/főoldalra azt egyszerűen az első menüpontra (Főoldal) vagy a menüpont melletti logóra kattintva másodpercek töredéke alatt megteheti.

```

<div class="swiper-slide slide">
  <div class="image">
    
  </div>
  <div class="content">
    <span>KEDVEZŐ ÁRAK, </span>
    <h3>LEGÚJABB ÓRÁK</h3>
    <a href="shop.php" class="btn">SZÉTNÉZEK KÖZÖTTÜK!</a>
  </div>
</div>

<div class="swiper-slide slide">
  <div class="image">
    
  </div>
  <div class="content">
    <span>VILÁGMÁRKÁK TÁRHÁZA</span>
    <h3>EREDETI TERMÉKEK</h3>
    <a href="shop.php" class="btn">MEGLESEM A KÍNÁLATOT!</a>
  </div>
</div>

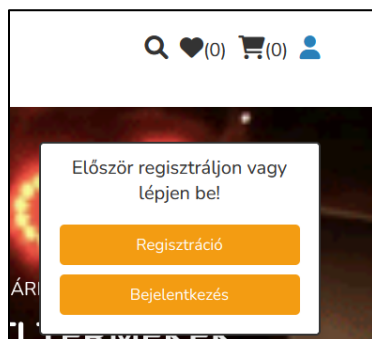
<div class="swiper-slide slide">
  <div class="image">
    
  </div>
  <div class="content">
    <span>GYORS KISZÁLLÍTÁS,</span>
    <h3>MEGBÍZHATÓSÁG</h3>
    <a href="shop.php" class="btn">LECSAPOK MOST!</a>
  </div>
</div>
</div>

```

A főoldalon egy mozgó kép előtt három óra, illetve szöveg és a gomb váltakozva mennek maguktól oda-vissza. Ha bármely gombot megnyomjuk a három közül, akkor már át is vagyunk irányítva a termékekhez. A képek diven belüli swiper-es szerkezettel működnek.

Regisztráció/bejelentkezés:

Ahhoz, hogy tudjunk regisztrálni vagy bejelentkezni, nincs más dolgunk, mint az oldal jobb oldalán található figurára kattintunk és választunk a két lehetőség közül:



Regisztráció:

user_register.php

Regisztrálásnál pár egyszerű adatot kell megadnunk, ezek a következők: Felhasználónév, E-mail, Jelszó és a Jelszó megerősítése.

Regisztrációs felület ábra:



isset segítségével megvizsgálja, a kívánt adatainkat regisztrációnál, hogy megfelelően formátumban adjuk-e meg, illetve, hogy a megfelelő mezőkbe megfelelő adatokat írunk-e, gondolok itt olyanra, hogy például, hogy az e-mail címből ne felejtsük ki a '@' karaktert, vagy, hogy a jelszavak megegyeznek-e. Visszajelzéseket is küld a rendszer, ilyenkor az oldal legtetején kapunk bizonyos üzeneteket. A kódsor ellenőrzi, hogy van-e már a rendszerben elmentve azonos e-mail cím, ha igen, akkor szimplán kapunk egy hibaüzenetet, miszerint az **'E-mail cím már használatban van!'**, másrészt, ha a jelszavainkat véletlenül elírtuk volna, akkor szintén kapunk egy üzenetet, hogy javítsuk az adott részt. Miután

```
if(isset($_POST['submit'])){\n\n    $name = $_POST['name'];\n    $name = filter_var($name, FILTER_SANITIZE_STRING);\n    $email = $_POST['email'];\n    $email = filter_var($email, FILTER_SANITIZE_STRING);\n    $pass = sha1($_POST['pass']);\n    $pass = filter_var($pass, FILTER_SANITIZE_STRING);\n    $cpass = sha1($_POST['cpass']);\n    $cpass = filter_var($cpass, FILTER_SANITIZE_STRING);\n\n    $select_user = $conn->prepare("SELECT * FROM `users` WHERE email = ?");\n    $select_user->execute([$email]);\n    $row = $select_user->fetch(PDO::FETCH_ASSOC);\n\n    if($select_user->rowCount() > 0){\n        $message[] = 'E-mail cím már használatban van!';\n    }else{\n        if($pass != $cpass){\n            $message[] = 'A jelszavak nem egyeznek meg!';\n        }else{\n            $insert_user = $conn->prepare("INSERT INTO `users` (name, email, password) VALUES(?,?,?)");\n            $insert_user->execute([$name, $email, $cpass]);\n            $message[] = 'Sikeres regisztráció, jelentkezzen be!';\n        }\n    }\n}
```

sikeresen megadtunk minden adatot, a rendszer ismét dob egy üzenetet, hogy **‘Sikeres regisztráció, jelentkezzen be!’** .

Section-ben kialakított résznél, azon belül pedig a **form** és az **input** kódok segítségével létre lett hozva a fentebb beillesztett regisztrációs panel. A felhasználó lehetőségei az adatoknál korlátozott karakterszámúgyileg, hisz senki nem akarná, hogy végtelen hosszú felhasználónevek között akarjuk keresgélni. Felhasználónévénél és a jelszónál a maximálisan megengedett karakterszám 20, míg az E-mail címnél 50.

A regisztrációs adatok alatt még van egy elhelyezett gomb, ami a href="user_login.php"-ra dob át, ha rákattintunk, ezen opció segítségével ha mégis lenne fiókunk, könnyedén áthelyez minket a bejelentkezés menüre, így csak meg kell adnunk ott az adatainkat.

```
<section class="form-container">
  <form action="" method="post">
    <h3>Regisztráció</h3>
    <input type="text" name="name" required placeholder="Felhasználónév" maxlength="20" class="box">
    <input type="email" name="email" required placeholder="E-mail" maxlength="50" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="password" name="pass" required placeholder="Jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="password" name="cpass" required placeholder="Jelszó megerősítése" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="submit" value="Regisztráció" class="btn" name="submit">
    <p>Van már fiókja?</p>
    <a href="user_login.php" class="option-btn">Bejelentkezés</a>
  </form>
</section>
```

Bejelentkezés:

user_login.php

A regisztrációt követően az oldal tetején kapunk egy üzenetet, amely a felhasználó számára jelzi, hogy a regisztráció sikeres/sikertelen volt-e. Miután a regisztrációval megvoltunk, ideje bejelentkezni a fiókunkba, itt már csak két adatot kell megadnunk, amit előzőleg regisztrációnál használtunk, ezek a következők: E-mail és a Jelszó.

Bejelentkezési felület ábra:

BEJELENTKEZÉS

Nincs még fiókja?

```

if(isset($_POST['submit'])){
    $email = $_POST['email'];
    $email = filter_var($email, FILTER_SANITIZE_STRING);
    $pass = sha1($_POST['pass']);
    $pass = filter_var($pass, FILTER_SANITIZE_STRING);

    $select_user = $conn->prepare("SELECT * FROM `users` WHERE email = ? AND password = ?");
    $select_user->execute([$email, $pass]);
    $row = $select_user->fetch(PDO::FETCH_ASSOC);

    if($select_user->rowCount() > 0){
        $_SESSION['user_id'] = $row['id'];
        header('location:home.php');
    }else{
        $message[] = 'Helytelen e-mail cím vagy jelszó!';
    }
}

```

Regisztrációs panelhez hasonlóan, itt is egy if-es ismét szerkezet íródott meg, amiben ellenőrizzük, hogy a felhasználó által megadott regisztrációs adatok helyesen lettek-e megadva. Ha a

felhasználó helyesen adta meg az adatokat, átirányítódik a home.php-ra, ami jelen esetben nálam a Főoldalt jelenti, ha pedig az adatok nem egyeznek, akkor egy else{-én belül \$message[] segítségével kap az illető egy hibaüzenetet: **'Helytelen e-mail cím vagy jelszó!'**.

Itt is section-ön belül, form és input segítségével létrejön a kialakított bejelentkezési menü az adatok sávi. A karakterszámok itt sem különböznek, E-mail cím maximum 50 karakter lehet, a Jelszó pedig 20. Beírható adatok alatt, ennél a panelnél is található egy kis üzenet, az alatt pedig egy gomb, ami az előzővel ellentétben, a 'user_register.php' oldalra visz át minket, ha megnyomjuk. Ott pedig a regisztrációs folyamatokat kezdetjük el, ha nincs még létező fiókunk.

```

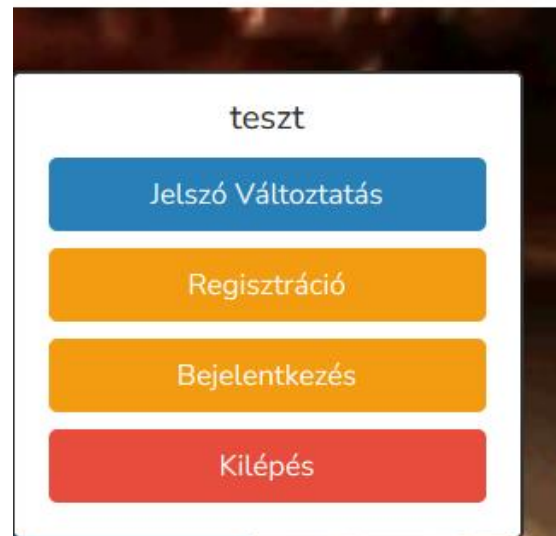
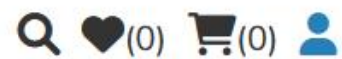
<section class="form-container">

  <form action="" method="post">
    <h3>Bejelentkezés</h3>
    <input type="email" name="email" required placeholder="E-mail" maxlength="50" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="password" name="pass" required placeholder="Jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="submit" value="Bejelentkezés" class="btn" name="submit">
    <p>Nincs még fiókja?</p>
    <a href="user_register.php" class="option-btn">Regisztráció</a>
  </form>
</section>

```

Sikeres bejelentkezés után újból a figurára kattintva már több lehetőség tárul elénk;

Ha mindent jól csináltunk, akkor a regisztrációt, majd a bejelentkezést követően a kis figurára kattintva már az általunk megadott felhasználónév virít. Ahogy látjuk, több lehetőség tárul már elénk bejelentkezés után. Tudjuk a fiókunk jelszavát változtatni a **'Jelszó Váloztatás'** gombra kattitva, ha pedig egy másik fiókkal szeretnék bejelentkezni vagy csak szimplán nem szeretnék tovább bejelentkezve maradni, arra szolgál egy **'Kilépés'** gomb.



Regisztrációs adatok elmentésre kerülnek az adatbázisba, **users** táblába:

+ Options

		id	name	email	password		
<input type="checkbox"/>	Edit	Copy	Delete	5	teszt	teszt@gmail.com	34228a532093278fcdc65c3a1338482e8bdc44f6

Jelszó változtatás:

JELSZÓ VÁLTOZTATÁS

teszt

teszt@gmail.com

Régi jelszó

Új jelszó

Új jelszó megerősítése

Frissítés

```
<section class="form-container">
  <form action="" method="post">
    <div>JELSZÓ VÁLTOZTATÁS</div>
    <input type="hidden" name="prev_pass" value="{<%= $fetch_profile['password']; %>}">
    <input type="text" name="name" required placeholder="Felhasználónév" maxlength="20" class="box" value="{<%= $fetch_profile['name']; %>}">
    <input type="email" name="email" required placeholder="E-mail" maxlength="50" class="box" oninput="this.value = this.value.replace(/\s/g, '')" value="{<%= $fetch_profile['email']; %>}">
    <input type="password" name="old_pass" placeholder="Régi jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="password" name="new_pass" placeholder="Új jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="password" name="cpass" placeholder="Új jelszó megerősítése" maxlength="20" class="box" oninput="this.value = this.value.replace(/\s/g, '')">
    <input type="submit" value="Frissítés" class="btn" name="submit">
  </form>
</section>
```

Természetesen a paneleknél két adat már alaphól ott lesz, ezek nem mások, mint a felhasználónevünk és az e-mail

címünk, könnyítve a felhasználói élményt. A parancsok több hibát is képesek kiszűrni. Ha nem adjuk meg a régi jelszavunkat és úgy próbálunk meg jelszót változtatni, akkor az oldal figyelmeztet, hogy **‘Add meg a régi jelszavad!’**, ez akkor lesz, ha az \$old_pass egyenlő less \$empty_pass-al. Második lehetőség a hibákra, ha az új és a régi jelszó azonos, ilyenkor elseif szerkezetben nézi a

program, hogyha \$old_pass nem egyenlő \$prev_pass-al, akkor egy **‘Előző jelszavaddal megegyezik’** üzenet fogad minket. Harmadik hibaként az elírt jelszavakra vonatkoznak, ha a \$new_pass nem egyenlő a \$cpass-al, akkor **‘Jelszavak nem egyeznek meg!’** hibaüzenetet kapunk. Az if-else szerkezetben két pont is van még, ha minden adatot helyesen adtunk meg, akkor már csak dicsérő üzenetet kapunk, miszerint **‘Sikeresen frissítetted a jelszavad!’**. Ez a folyamat bármikor megismételhető, nincs korlátozva a jelszóváltoztatások mennyisége.

```
if(isset($_POST['submit'])) {
    $name = $_POST['name'];
    $name = filter_var($name, FILTER_SANITIZE_STRING);
    $email = $_POST['email'];
    $email = filter_var($email, FILTER_SANITIZE_STRING);

    $update_profile = $conn->prepare("UPDATE `users` SET name = ?, email = ? WHERE id = ?");
    $update_profile->execute([$name, $email, $user_id]);

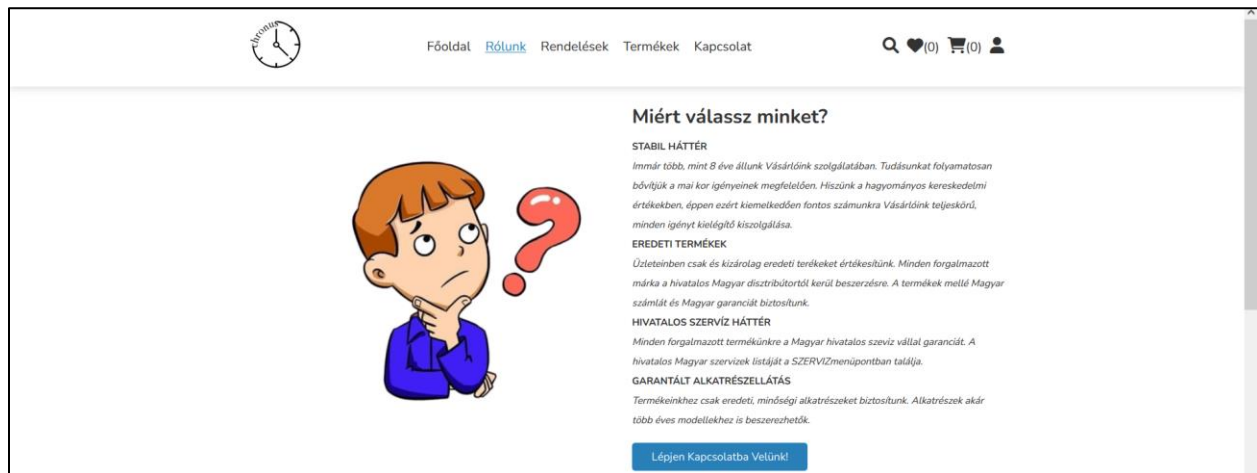
    $empty_pass = 'da39a3ee5e6b4b0d3255bfef95601890afd80709';
    $prev_pass = $_POST['prev_pass'];
    $old_pass = sha1($_POST['old_pass']);
    $old_pass = filter_var($old_pass, FILTER_SANITIZE_STRING);
    $new_pass = sha1($_POST['new_pass']);
    $new_pass = filter_var($new_pass, FILTER_SANITIZE_STRING);
    $cpass = sha1($_POST['cpass']);
    $cpass = filter_var($cpass, FILTER_SANITIZE_STRING);

    if($old_pass == $empty_pass){
        $message[] = 'Add meg a régi jelszavad!';
    }elseif($old_pass != $prev_pass){
        $message[] = 'Nem megfelelő a jelszó!';
    }elseif($new_pass != $cpass){
        $message[] = 'Jelszavak nem egyeznek meg!';
    }else{
        if($new_pass != $empty_pass){
            $update_admin_pass = $conn->prepare("UPDATE `users` SET password = ? WHERE id = ?");
            $update_admin_pass->execute([$cpass, $user_id]);
            $message[] = 'Sikeresen frissítetted a jelszavad!';
        }else{
            $message[] = 'Adj meg egy új jelszót!';
        }
    }
}
```

Kezdő oldal (Rólunk):

A Rólunk menüpontra kattintva a “cégről” van ízelítő, itt bemutatam, hogy miért érdemes az oldalról rendelni, mióta létezik a “cég”, esik szó még a szervízárról és a kínált garanciáról is. Ennél a menüpontnál IS van lehetősége a felhasználónak kapcsolatba lépni az üzemeltetővel a ‘**Lépjén Kapcsolatba Velünk**’ gombra kattintva - ami után egy másik menübe dobja át a felhasználót (Kapcsolat) - , illetve itt olvashat pár vásárlói visszajelzést is.

about.php bemutatása:



```
<?php include 'components/user_header.php'; ?>

<section class="about">

    <div class="row">

        <div class="image">
            
        </div>

        <div class="content">
            <h3>Miért válassz minket?</h3>
            <p><b>STABIL HÁTTER</b></p>
            <br><i>Imár több, mint 8 éve állunk Vásárlóink szolgálatában. Tudásunkat folyamatosan bővítjük a mai kor igényeinek megfelelően. Hiszünk a hagyományos kereskedelmi értékekben, éppen ezért kiemelkedően fontos számunkra Vásárlóink teljeskörű, minden igényt kielégítő kiszolgálása.</i>
            <p><b>EREDETI TERMÉKEK</b></p>
            <br><i>Üzleteinkben csak és kizárólag eredeti terékeket értékesítünk. Minden forgalmazott márka a hivatalos Magyar disztribútortól kerül beszerzésre. A termékek mellé Magyar számlát és Magyar garanciát biztosítunk.</i>
            <p><b>HIVATALOS SZERVÍZ HÁTTER</b></p>
            <br><i>Minden forgalmazott termékünk a Magyar hivatalos szervíz vállal garanciát. A hivatalos Magyar szervizek listáját a SZERVÍZmenüponthon találja.</i>
            <p><b>GARANTÁLT ALKATRÉSZELLÁTÁS</b></p>
            <br><i>Termékeinkhez csak eredeti, minőségi alkatrészeket biztosítunk. Alkatrészek akár több éves modellekhez is beszerezhetők.</i>
            <p><a href="contact.php" class="btn">Lépjén kapcsolatba velünk!</a>
        </div>
    </div>

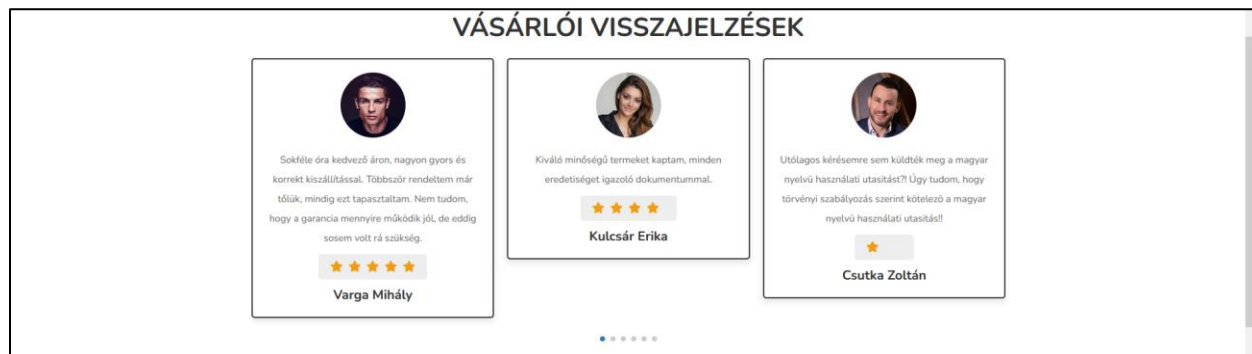
</section>
```

Legörgetve megtalálható pár általam feltöltött vásárlói vélemény/visszajelzés, amiről lentebb írok bővebben.

A vásárlói visszajelzések között a felhasználó az egér/touchpad segítségével tud válogatni, egyszerűen csak húznia kell balra vagy jobbra, attól függ, hogy melyik vélemény érdekli jobban a felhasználót. Ez a szerkezet egy swiper segítségével működik div-en belül. Ehhez külön script van, ami a weboldalon több pontra is érvényes, van amely magától is megy más időbeli különbségekkel, van ami pedig nem megy magától, a felhasználónak kell közöttük válogatni.

```
<section class="reviews">
  <h1 class="heading">VÁSÁRLÓI VISSZAJELZÉSEK</h1>
  <div class="swiper reviews-slider">
    <div class="swiper-wrapper">
      <div class="swiper-slide slide">
        
        <p>Sokféle óra kedvező áron, nagyon gyors és korrekt kiszállítással. Többször rendeltem már tőlük, mindig ezt tapasztaltam. Nem tudom, hogy a garancia mennyire működik jól, de eddig</p>
        <div class="stars">
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
        </div>
        <h3>Varga Mihály</h3>
      </div>
      <div class="swiper-slide slide">
        
        <p>Kiváló minőségű terméket kaptam, minden eredetiséget igazoló dokumentummal.</p>
        <div class="stars">
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star-empty"></i>
        </div>
        <h3>Kulcsár Erika</h3>
      </div>
      <div class="swiper-slide slide">
        
        <p>Utólagos kérdésemre sem küldték meg a magyar nyelvű használati utasítást!! Úgy tudom, hogy törvényi szabályozás szerint kötelező a magyar nyelvű használati utasítás!</p>
        <div class="stars">
          <i class="fas fa-star"></i>
          <i class="fas fa-star-empty"></i>
          <i class="fas fa-star-empty"></i>
          <i class="fas fa-star-empty"></i>
          <i class="fas fa-star-empty"></i>
        </div>
        <h3>Csutka Zoltán</h3>
      </div>
    </div>
  </div>
```

```
<script>
var swiper = new Swiper(".reviews-slider", {
  loop:true,
  spaceBetween: 20,
  pagination: {
    el: ".swiper-pagination",
    clickable:true,
  },
  breakpoints: {
    0: {
      slidesPerView:1,
    },
    768: {
      slidesPerView: 2,
    },
    991: {
      slidesPerView: 3,
    },
  },
});
```



```
<section class="reviews">
  <h1 class="heading">VÁSÁRLÓI VISSZAJELZÉSEK</h1>
  <div class="swiper reviews-slider">
    <div class="swiper-wrapper">
      <div class="swiper-slide slide">
        
        <p>Sokféle óra kedvező áron, nagyon gyors és korrekt kiszállítással. Többször rendeltem már tőlük, mindig ezt tapasztaltam. Nem tudom, hogy a garancia mennyire működik jól, de eddig sosem volt rá szükség.</p>
        <div class="stars">
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
        </div>
        <h3>Varga Mihály</h3>
      </div>
    </div>
  </div>
```

Kezdő oldal (Rendelések, Termékek):

Elsődlegesen, hogy egy felhasználó tudjon bármit is rendelni, az oldalon szerepelnie kell termékeknek.

Az én oldalamon phpMyAdmin segítségével könnyedén megtehetjük. (Ezt lehetett az admin felületről is, de úgy minden egyes alkalommal ha másik gépről kellene, illetve másik hálózatról elérni a weboldalt, akkor vinni kellett volna a képeket, leírásokat, stb..., így ezt a könnyedebb, időhatékonyabb módszer mellett maradt a weboldal.

Egy példa egy órához tartozó adatokról:

id	name	details	price	image_01	image_02	image_03
1	ROLEX DAYTONA	WATCH DETAILS Sku: X-RDYT07215 Box: Original Fac...	12500	rol2-removebg-preview.png	rol1-removebg-preview.png	rol3-removebg-preview.png

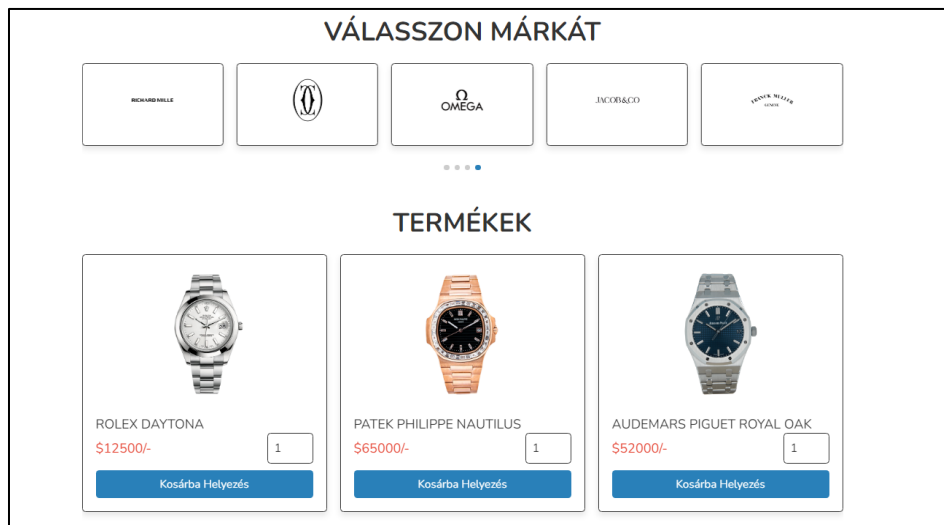
Pár adatot meg kell adnunk ahhoz, hogy normálisan látható legyen az oldalon a termék, ezek a következők: ID, a termék neve, leírás a termékről, a termék ára és három tetszőleges kép a termékről.

A képeknél ügyeltem arra, hogy ne legyen háttere ezeknek, amik elrontanák a képet, ehhez egy weboldalt használtam, mégpedig a <https://www.remove.bg/> oldalát.

Pár példa:



Márka választás, termékek megjelenése a főoldalon:



```
<section class="category">

  <h1 class="heading">VÁLASSZON MÁRKÁT</h1>

  <div class="swiper category-slider">

    <div class="swiper-wrapper">

      <a href="category.php?category=rolex" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=patek_philippe" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=audemars_piguet" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=richard_mille" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=cartier" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=omega" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=jacob_&_co" class="swiper-slide slide">
        
      </a>

      <a href="category.php?category=franck_muller" class="swiper-slide slide">
        
      </a>

    </div>

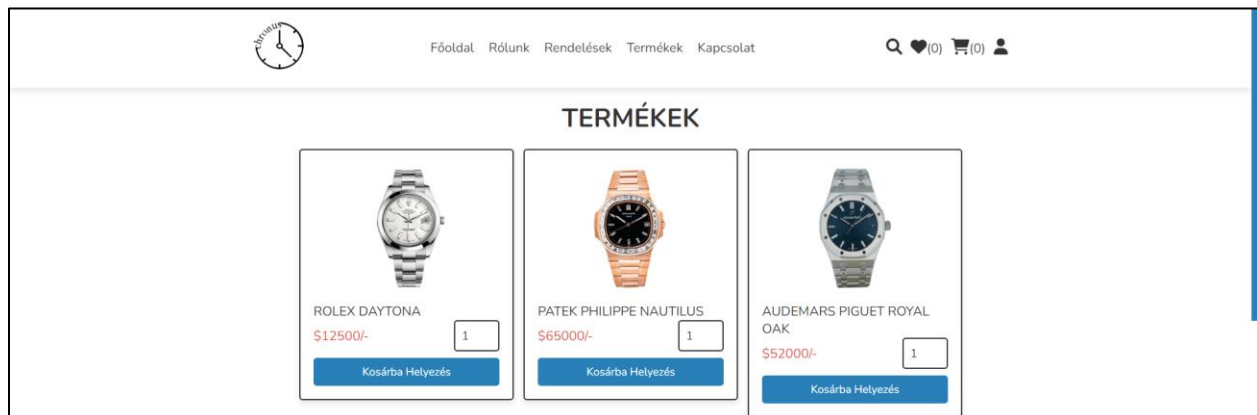
    <div class="swiper-pagination"></div>

  </div>

</section>
```

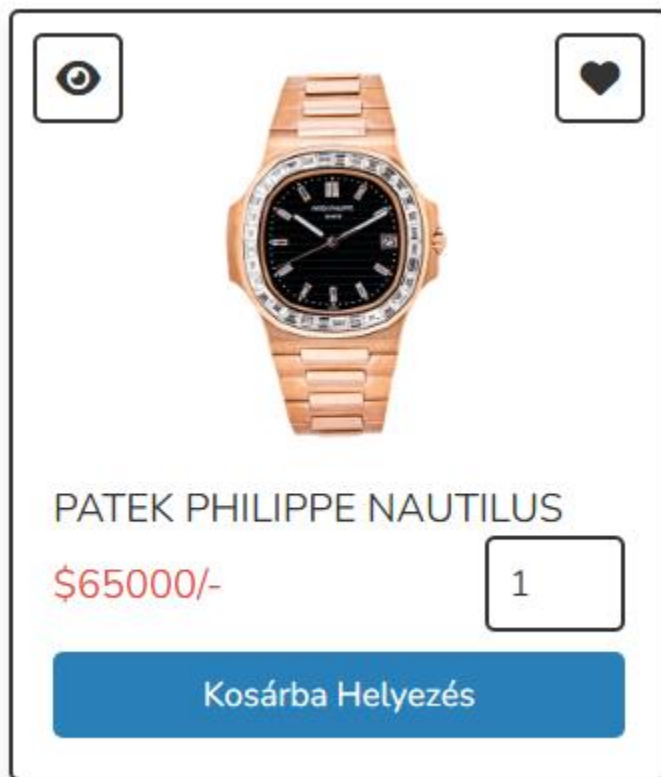
Noha a márka választás résznél nem a legjobb módszert választottam, de szerintem ez egy mezei felhasználónak nem szúr szemet, az oldalon eltöltött ideje alatt pedig semmi hátránya nem születhet ebből. Ahogy a képen is látszik, swiper-es div-en belül a kis rubrikák belsejében egy-egy logó látható, ami mind-mind egy márka. A felhasználó egere segítségével elérhet több márkát is, egy kattintás segítségével pedig kiszűrheti az adott márka termékét. Természetesen alapból az összes termék látható, ez egyfajta szűrőként használható, ha már van egy vásárlási célból kiválasztott márka a fejünkben.

Termékek pontra kattintva kaphatunk egy nagyobb ,tisztább képet az aktuális ajánlatokról.




Áttekintés, kedvenc, kosár:

Ha egy termékre rávisszük az egerünk kurzurát, két plusz funkció is láthatóvá válik:



1, Bal oldali ikon: Lehetőségünk van bővebb képet kapni az adott termékről (leírás, képek..)

ÁTTEKINTÉS






PATEK PHILIPPE NAUTILUS

\$65000/-

1

Model - Patek Philippe Nautilus *ORIGINAL PATEK MOVEMENT ONLY*
Reference Number - 5711 Movement – Automatic Bezel - Rose Gold Custom
Baguette Diamond Bezel Dial - Black Custom Diamond Dial Case Size –
40mm Crystal - Scratch-resistant Sapphire Crystal Bracelet - Rose Gold
Bracelet Water Resistant - 10 ATM Gender – Men's

Kosár Kedvenc



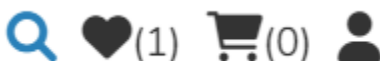
Bármelyik képre kattintunk, megjelenik, így minden szögből át tudjuk nézni a terméket. Itt már kosárba is tudjuk helyezni a terméket, kért mennyiséget tudjuk változtatni, illetve a kedvencekbe is helyezhetjük, ha későbbre szeretnénk elmenteni.

2, Jobb oldali ikon: Egy gyors gomb, ami a Kedvencekbe helyezi a terméket.

Megnyomást követően egy üzenet kapunk fent, ami mutatja, hogy sikeres volt a művelet, illetve a jobb oldali header sávban a szív mellett megváltozik az érték.

Képekkel ábrázolva:

Hozzáadva a kedvencekhez! ✖



3, Kosárba helyezés gomb: Nevéből adódóan a kiválasztott terméket a kosárba rakja.

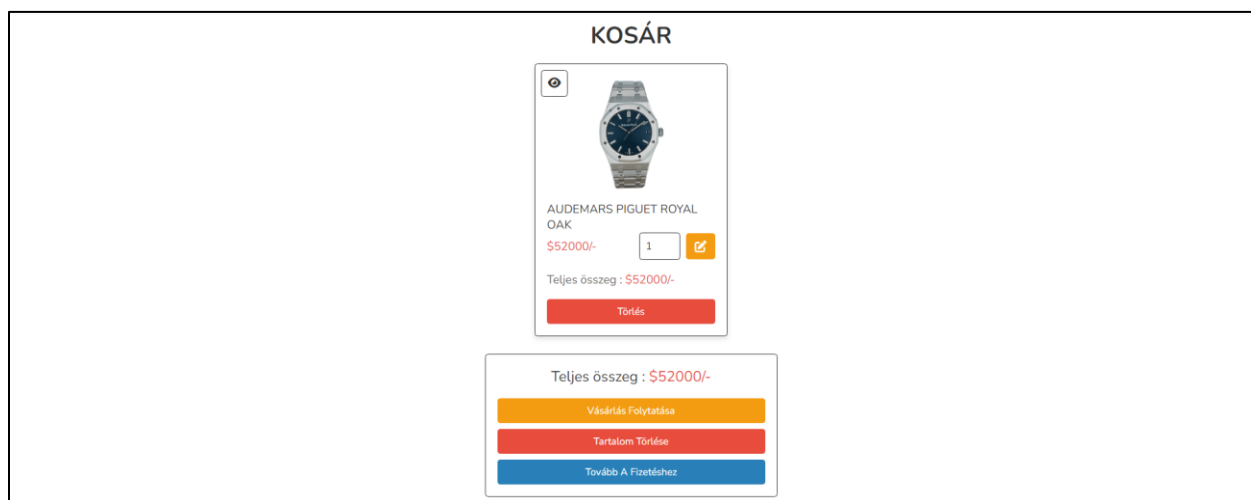
Megnyomást követően hasonlóan mint a kedvenceknél, itt is kapunk egy üzenet, illetve a kosár ikon melletti szám értéke megváltozik.


Képekkel ábrázolva:

Kosárhoz adva! ✖



Bevásárlókocsi ikonra kattintva ez az oldal fogad minket:



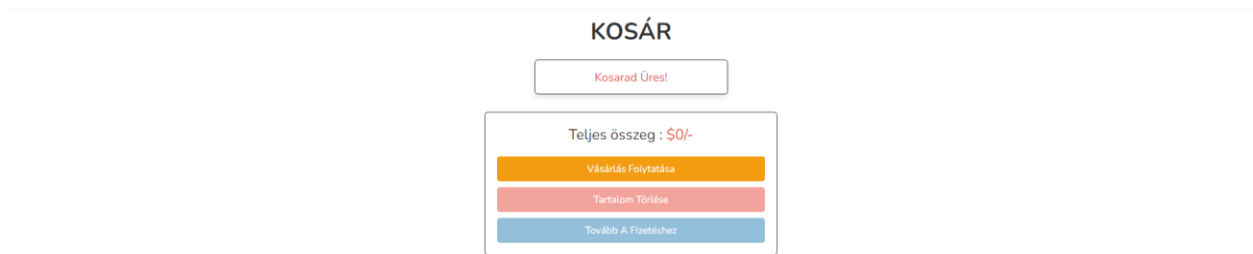
Egerünk kurzurát a termékre mozdítva szintén előjön az ikon ,amivel megint át tudjuk tekinteni a terméket. Ha az adott termékből többet szeretnék rendelni, anélkül, hogy többször a kosárba helyeznék a terméket, szimplán az ár melletti rublikában lévő számot a kívánt mennyiségre kell változtatni, majd a mellette lévő ikont  megnyomva a kosarunk tartalma frissül, ezt is egy üzenet követi.

Kosár tartalma frissült! ✖

Törles gombra nyomva a kosarunk tartalma kiürül, amit először megkérdez a rendszer, hogy biztosan akarjuk-e.



Miután megnyomtuk az OK felirattal ellátott gombot, a kosarunk tartalma kiürül.



További gombok, amik megtalálhatóak:

-1, Vásárlás folytatása

-2, Tartalom törlése

-3 Tovább a fizetéshez

Kettes és a harmadik pontban látható (`$grand_total > 1`) azt a segítséget nyújtja számunkra, hogyha nincsen a kosárban semmi, akkor a gombok csak látszódnak, de a felhasználó számára nem nyújtanak semmit.

```
<? = ($grand_total > 1)?'': 'disabled'; ?>
```

1, Vásárlás folytatás:

```
<a href="shop.php" class="option-btn">Vásárlás folytatása</a>
```

Gombra linkelve lett az a 'shop.php', mely megnyomásával visszakerülünk a termékekhez, így ha esetleg elfelejtettünk, szeretnénk más terméket is rendelni, könnyedén megtehetjük.

2, Tartalom törlése:

```
<a href="cart.php?delete_all" class="delete-btn <? = ($grand_total > 1)?'': 'disabled'; ?>" onclick="return confirm('Kosár tartalmának törlése?');">Tartalom törlése</a>
```

Tartalom törlése gombra kattintva kapunk egy felugró üzenetet, hogy biztos szeretnénk-e folytatni a folyamatot, ott, ha engedélyezzük, akkor az oldal tetején lévő kép fogad újra minket és a kosarunk tartalma üres lesz ismét.

3, Tovább a fizetéshez:

```
<a href="checkout.php" class="btn <? = ($grand_total > 1)?'': 'disabled'; ?>">Tovább a fizetéshez</a>
```

Tovább a fizetéshez gombot miután megnyomtuk, a gombra linkelt 'checkout.php' oldalára kerülünk, ahol egy másik panel kerül elénk.

Termék keresése:

Van lehetőségünk megnyitni egy keresési mezőt, amely segítségével rá tudunk keresni az általunk kimondottan keresett termékekre. Ehhez nincs más dolgunk, csak a header részben rányomni az erre szolgáló ikonra.



Megnyomását követően ez a sor fog megjelenni: search_page.php



A keresés a következő módon történik:

```
36 <section class="search-form">
37 <form action="" method="post">
38 <input type="text" name="search_box" placeholder="Keresés..." maxlength="100" class="box" required>
39 <button type="submit" class="fas fa-search" name="search_btn"></button>
40 </form>
41 </section>
42
43 <section class="products" style="padding-top: 0; min-height:100vh;">
44
45 <div class="box-container">
46
47 <?php
48 if(isset($_POST['search_box']) OR isset($_POST['search_btn']))){
49 $search_box = $_POST['search_box'];
50 $select_products = $conn->prepare("SELECT * FROM `products` WHERE name LIKE '%{$search_box}%'");
51 $select_products->execute();
52 if($select_products->rowCount() > 0){
53 while($fetch_product = $select_products->fetch(PDO::FETCH_ASSOC)){
54 }
55 }
56 <form action="" method="post" class="box">
57 <input type="hidden" name="pid" value="<?=$fetch_product['id']; ?>">
58 <input type="hidden" name="name" value="<?=$fetch_product['name']; ?>">
59 <input type="hidden" name="price" value="<?=$fetch_product['price']; ?>">
60 <input type="hidden" name="image" value="<?=$fetch_product['image_01']; ?>">
61 <button class="fas fa-heart" type="submit" name="add_to_wishlist"></button>
62 <a href="quick_view.php?pid=<?=$fetch_product['id']; ?>" class="fas fa-eye"></a>
63 
64 <div class="name"><?=$fetch_product['name']; ?></div>
65 <div class="flex">
66 <div class="price"><span><?=$fetch_product['price']; ?></span></div>
67 <input type="number" name="qty" class="qty" min="1" max="99" onkeypress="if(this.value.length == 2) return false;" value="1">
68 </div>
69 <input type="submit" value="Kosárba helyezés" class="btn" name="add_to_cart">
70 </form>
71 <?php
72 }
73 }else{
74 echo '<p class="empty">Nem található termék!</p>';
75 }
76 }
77 ?>
78
79 </div>
```

Bármelyin
karaktert
beüthetünk,
viszont csak a
betűket nézi az
oldal, ha egy
betűt írunk be,
akkor ha az adott
termék(ek)
tartalmazza ezt a
betűt, az összes
terméket ki fogja
dobni.

Érdekes hát
biztosra mennünk
és helyesen beírni
a keresett

termékünk nevét. Ha elírjuk a nevet, vagy nincs olyan feltöltött termék, akkor else-n belül echo segítségével kapunk egy üzenetet, miszerint '**Nem található termék!**'.

Ha jól adtuk meg a termék nevét, akkor minden feltöltött adatot, képet megkapunk a termékről.

Ábrázolom a két lehetőséget:


1, Elírt, hibás keresés:

‘Rolex’ márkájú órára kerestem rá, de elírtam, így ‘Rolx’ lett belőle, amit a rendszer nem talál már.

Nem Található Termék!

2, Jól beírt, sikeres keresés:

Itt is a ‘Rolex’ márkájú órára kerestem, már helyesen írtam be az adatokat, meg is lett az eredménye:



ROLEX DAYTONA
\$12500/-

1

Kosárba Helyezés

Információkat megkaptuk, helyezhetjük is a kosárba! 😊

Rendelés:

Miután megnyomtuk a ‘Tovább a fizetéshez’ gombot, akkor a ‘checkout.php’ tartalmát fogja betölteni:

Itt, minden weboldalon ahol termékek árulásával és kézbesítésével foglalkoznak, kiszállítási adatokat kell megadni.

RENDELÉSEID cím alatt összegzi az oldal, hogy mit is rendeltünk, mennyit, illetve mennyibe kerül az egész.

RENDELÉSEID

PATEK PHILIPPE NAUTILUS (\$65000/- x 1)

Teljes összeg: \$65000/-

ADD LE A RENDELÉSED

Név:

Telefonszám:

E-mail:

Fizetési mód:

Szállítási cím 01:

Szállítási cím 02:

Város:

Ország:

Irányítószám:

Rendelés Leadása

```

<h3>Add le a rendelésed</h3>

<div class="flex">
  <div class="inputBox">
    <span>Név:</span>
    <input type="text" name="name" placeholder="Név" class="box" maxlength="20" required>
  </div>
  <div class="inputBox">
    <span>Telefonszám:</span>
    <input type="number" name="number" placeholder="Telefonszám" class="box" min="0" max="999999999" onkeypress="if(this.value.length == 11) return false;" required>
  </div>
  <div class="inputBox">
    <span>E-mail:</span>
    <input type="email" name="email" placeholder="E-mail" class="box" maxlength="50" required>
  </div>
  <div class="inputBox">
    <span>Fizetési mód:</span>
    <select name="method" class="box" required>
      <option value="Utánvétes fizetés">Utánvétes fizetés</option>
      <option value="Hitelkártya">Hitelkártya</option>
      <option value="Paypal">Paypal</option>
    </select>
  </div>
  <div class="inputBox">
    <span>Szállítási cím 01:</span>
    <input type="text" name="flat" placeholder="Házszám / kapuszám" class="box" maxlength="50" required>
  </div>
  <div class="inputBox">
    <span>Szállítási cím 02:</span>
    <input type="text" name="street" placeholder="Utca név" class="box" maxlength="50" required>
  </div>
  <div class="inputBox">
    <span>Város:</span>
    <input type="text" name="city" placeholder="Város" class="box" maxlength="50" required>
  </div>
  <div class="inputBox">
    <span>Ország:</span>
    <input type="text" name="country" placeholder="Ország" class="box" maxlength="50" required>
  </div>
  <div class="inputBox">
    <span>Írányítószám:</span>
    <input type="number" min="0" name="pin_code" placeholder="Írányítószám" min="0" max="99999" onkeypress="if(this.value.length == 4) return false;" class="box" required>
  </div>
</div>

<input type="submit" name="order" class="btn" <?=( $grand_total > 1) ? '':'disabled'; ?>" value="Rendelés leadása">

```

ADD LE A RENDELÉSED
CÍM alatt pedig az adatainkat kell megadni, segítségünkre szolgál, hogy kétszer is láthatóak, hogy melyik adat hova kell. Fizetési módnál három

lehetőség van: Utánvétes, PayPal, illetve Hitelkártya, ha a felhasználó rákattint, akkor egy legördülő listából könnyedén ki tudja választani, melyiket is szeretné. Vannak korlátozva bizonyos adatok hosszúságai, ezek például a telefonszám és az irányítószám.

Miután kitöltöttük helyesen az adatainkat, 'Rendelés leadása' gomb megnyomása után felül kapunk egy visszaigazoló üzenetet, hogy sikeresen leadtuk, majd a rendelés adatok, illetve a kosár tartalma nullázódik, vagyis törlődnek.

A leadott rendelést admin felüleleten a tulajdonos láthatja, ezt kicsit később még be.

```

<h3>Rendeléseid</h3>

<div class="display-orders">
  <?php
    $grand_total = 0;
    $cart_items[] = '';
    $select_cart = $conn->prepare("SELECT * FROM `cart` WHERE user_id = ?");
    $select_cart->execute([$user_id]);
    if($select_cart->rowCount() > 0){
      while($fetch_cart = $select_cart->fetch(PDO::FETCH_ASSOC)){
        $cart_items[] = $fetch_cart['name'].' ( '.$fetch_cart['price'].' x '. $fetch_cart['quantity'].' ) - ';
        $total_products = implode($cart_items);
        $grand_total += ($fetch_cart['price'] * $fetch_cart['quantity']);
      }
    }
    <p> <?=( $fetch_cart['name']; ?> <span><?=( $fetch_cart['price'].' x '. $fetch_cart['quantity']; ?>)</span> </p>
  <?php
    }
  }else{
    echo 'A kosarad üres!';
  }
  <?>
  <input type="hidden" name="total_products" value="<?=( $total_products; ?>)">
  <input type="hidden" name="total_price" value="<?=( $grand_total; ?>" value="">
  <div class="grand-total">Teljes összeg: <span><?=( $grand_total; ?>)</span></div>
</div>

<h3>Add le a rendelésed</h3>

```

Kezdő oldal (Kapcsolat):

Az oldal felhasználóinak van lehetőségük felkeresni az oldal tulajdonosát/it. Könnyedén tudnak a 'Kapcsolat' menüpontra lépés után egy pár adat megadása után privát üzenetet küldeni, ami csak a felhasználó és az admin számára látható, olvasható.

Lépjén Kapcsolatba Velünk!

Üzenet

Üzenet Elküldése

```
<section class="contact">
  <form action="" method="post">
    <h3>Lépjén kapcsolatba velünk!</h3>
    <input type="text" name="name" placeholder="Név" required maxlength="30" class="box">
    <input type="email" name="email" placeholder="E-mail" required maxlength="50" class="box">
    <input type="number" name="number" placeholder="Telefonszám" required onkeypress="if(this.value.length == 11) return false;" class="box">
    <textarea name="msg" class="box" placeholder="Üzenet" cols="30" rows="10"></textarea>
    <input type="submit" value="Üzenet elküldése" name="send" class="btn">
  </form>
</section>
```

Isset segítségével
küldjük az adatokat
ismét, kell a
felhasználónév, e-
mail, telefonszám és
maga a
megfogalmazott
üzenetünk.

A parancssor átnézi,
hogy küldünk-e már
ilyen üzenetet
hasonló adatokkal a

\$select_message résznél, ha nem küldtünk, az \$insert_message feltölti a 'messages' adatbázis táblába az adatokat, illetve el is küldi az adatokat az admin/tulaj részére.

```
if(isset($_POST['send'])){
    $name = $_POST['name'];
    $name = filter_var($name, FILTER_SANITIZE_STRING);
    $email = $_POST['email'];
    $email = filter_var($email, FILTER_SANITIZE_STRING);
    $number = $_POST['number'];
    $number = filter_var($number, FILTER_SANITIZE_STRING);
    $msg = $_POST['msg'];
    $msg = filter_var($msg, FILTER_SANITIZE_STRING);

    $select_message = $conn->prepare("SELECT * FROM `messages` WHERE name = ? AND email = ? AND number = ? AND message = ?");
    $select_message->execute([$name, $email, $number, $msg]);

    if($select_message->rowCount() > 0){
        $message[] = 'Elküldte már az üzenetet!';
    }else{
        $insert_message = $conn->prepare("INSERT INTO `messages` (user_id, name, email, number, message) VALUES(?,?,?,?,?)");
        $insert_message->execute([$user_id, $name, $email, $number, $msg]);

        $message[] = 'Sikeres üzenetküldés!';
    }
}
```

Kezdő oldal (Footer rész):

```
<footer class="footer">
<section class="grid">
<div class="box">
<a href="https://megbizhatobolt.hu/" target="_blank"></a>
</div>
<div class="box">
<h3>Főbb linkek</h3>
<a href="home.php"><i class="fas fa-angle-right"></i>Főoldal</a>
<a href="about.php"><i class="fas fa-angle-right"></i>Rólunk</a>
<a href="shop.php"><i class="fas fa-angle-right"></i>Termékek</a>
<a href="contact.php"><i class="fas fa-angle-right"></i>Elérhetőség</a>
</div>
<div class="box">
<h3>Kisegítő linkek</h3>
<a href="user_login.php"><i class="fas fa-angle-right"></i>Bejelentkezés</a>
<a href="user_register.php"><i class="fas fa-angle-right"></i>Regisztráció</a>
<a href="cart.php"><i class="fas fa-angle-right"></i>Kosár</a>
<a href="orders.php"><i class="fas fa-angle-right"></i>Rendelések</a>
</div>
<div class="box">
<h3>Elérhetőségeink</h3>
<a href="index.php"><i class="fas fa-phone"></i>+36 30 555 4444</a>
<a href="mailto:chronus@protonmail.com" target="_blank"><i class="fas fa-envelope"></i>chronus@watch.com</a>
<a href="https://www.google.com/maps/d/viewer?mid=iuPDbY7u5xIF-jnfrBBU9MG7wf-w&hl=en_US&ll=46.906250000000014%2C19.688" target="_blank"><i class="fas fa-map-marker"></i>Kecskemét, HU - 6000</a>
</div>
<div class="box">
<h3>Közösségi oldalaink</h3>
<a href="https://www.facebook.com" target="_blank"><i class="fab fa-facebook-f"></i>Facebook</a>
<a href="https://www.twitter.com" target="_blank"><i class="fab fa-twitter"></i>Twitter</a>
<a href="https://www.instagram.com" target="_blank"><i class="fab fa-instagram"></i>Instagram</a>
</div>
</section>
<div class="credit"><span>CHRONUS</span> &copy; 2015 - <?= date('Y')> Minden jog fenntartva </div>
</footer>
```

Az oldal legaljára görgetve pár linket, közösségi oldalakat, az oldalról pár információt és az elérhetőségeket helyeztem el. Oszlopra vannak szedve, kategorizálva. **Főbb linkeknél** a főbb menüpontokat helyeztem el, amelyre ha egyszerűen rákattintanak, oda dobja őket az oldal.

Kisegítő linkek pontnál inkább a termékekhez

kötődő dolgokat helyeztem el, ha valaki esetleg nem találná a regisztrációs vagy bejelentkezős felületet, akkor egy egyszerű kattintással, már az adott panelnél láthatják magukat.

Elérhetőségeink résznél egyértelműen az oldal elérhetőségeit írtam le, persze ezek fiktív információk, nem mind igazi, inkább csak dizájn elem.

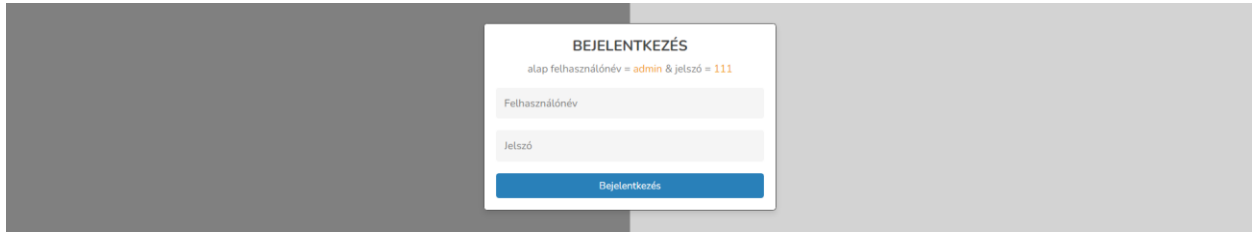
Közösségi oldalaink menünél, ami az utolsó, ott pedig pár elég neves social media felületet írtam be, ahol egy órák oldal reklámozhatja magát, itt a szöveg mellett 'fab-fa-(social media neve)' class-al az adott oldal ikonját is behelyeztem, kicsit dob a kinézetén.

```
<div class="box">
<a href="https://megbizhatobolt.hu/" target="_blank"></a>
</div>
```

Egy megbízható bolt logót is helyeztem el a pontok mellett, ez jó benyomást adhat a felhasználónak, hisz nem újdonság napjainkban, hogy átverésre is készíthetnek weboldalakat.

	Főbb Linkek	Kisegítő Linkek	Elérhetőségeink	Közösségi Oldalaink
	> Főoldal	> Bejelentkezés	+36 30 555 4444	f Facebook
	> Rólunk	> Regisztráció	chronus@watch.com	t Twitter
	> Termékek	> Kosár	Kecskemét, HU - 6000	@ Instagram
	> Elérhetőség	> Rendelések		
CHRONUS © 2015 - 2023 Minden jog fenntartva				

Admin panel felépítése:



```
<section class="form-container">  
  <form action="" method="post">  
    <h3>Bejelentkezés</h3>  
    <p>alap felhasználónév = <span>admin</span> & jelszó = <span>111</span></p>  
    <input type="text" name="name" required placeholder="Felhasználónév" maxlength="20" class="box" oninput="this.value = this.value.replace(/\\s/g, '')">  
    <input type="password" name="pass" required placeholder="Jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/\\s/g, '')">  
    <input type="submit" value="Bejelentkezés" class="btn" name="submit">  
  </form>  
</section>
```

Készült az oldal mellé egy admin “weboldal” is, melyhez a weboldal tulajdonosának és annak van hozzáférése, aki kap felhatalmazást. Természetesen több adminisztrációs fiókot is lehet létrehozni, de jobbnak láttam, ha egy olyan van, ami alap és látható minden adata.

A belépési adatok hozzá láthatóak a ‘BEJELENTKEZÉS’ címszó után.

Felhasználónév – admin, Jelszó – 111.

Természetesen itt is figyelve van az, hogy helyes adatokat adjunk meg, így két opciót ad meg az oldal, ha helyesen adjuk meg az adatokat, akkor admin/dashboard.php menüre dob át minket, ha pedig helytelen valamelyik adatunk, hibaüzenetet kapunk, úgyhogy jobb, ha megjegyezzük az adatainkat. 😊

```
if(isset($_POST['submit'])){  
    $name = $_POST['name'];  
    $name = filter_var($name, FILTER_SANITIZE_STRING);  
    $pass = sha1($_POST['pass']);  
    $pass = filter_var($pass, FILTER_SANITIZE_STRING);  
  
    $select_admin = $conn->prepare("SELECT * FROM `admins` WHERE name = ? AND password = ?");  
    $select_admin->execute([$name, $pass]);  
    $row = $select_admin->fetch(PDO::FETCH_ASSOC);  
  
    if($select_admin->rowCount() > 0){  
        $_SESSION['admin_id'] = $row['id'];  
        header('location:dashboard.php');  
    }else{  
        $message[] = 'Helytelen felhasználónév vagy jelszó!';  
    }  
}
```

Admin panel (Írányítópult):

Bejelentkezést követően a dashboard menüre, vagyis az irányítópultra kerülünk, ezen az oldalon látható számunkra a felhasználók által létrehozott profilok, a feltöltött termékek, a létrehozott admin profilok



Az elhelyezéshez box class-os diveket használtam, amelyeket css-ben szépen középre is helyeztem, egymás mellé, hogy az átláthatóság könnyű legyen. A boxokban lévő gombok mind-mind az adott felül látható menüpontokra mutatnak, amik linkelve vannak hozzájuk. Egy

```
<section class="dashboard">
  <h1 class="heading">Írányítópult</h1>
  <div class="box-container">
    <div class="box">
      <h3>Üdvözöllek!</h3>
      <p><?php echo $fetch_profile['name']; ?></p>
      <a href="update_profile.php" class="btn">Profil szerkesztése</a>
    </div>
    <div class="box">
      <?php
        $total_pendings = 0;
        $select_pendings = $conn->prepare("SELECT * FROM `orders` WHERE payment_status = ?");
        $select_pendings->execute(['pending']);
        if($select_pendings->rowCount() > 0){
          while($fetch_pendings = $select_pendings->fetch(PDO::FETCH_ASSOC)){
            $total_pendings += $fetch_pendings['total_price'];
          }
        }
      >
      <h3><span><?php echo $total_pendings; ?></span></h3>
      <p>Függőben lévő rendelések</p>
      <a href="placed_orders.php" class="btn">Megtekintése</a>
    </div>
    <div class="box">
      <?php
        $select_products = $conn->prepare("SELECT * FROM `products`");
        $select_products->execute();
        $number_of_products = $select_products->rowCount();
      >
      <h3><span><?php echo $number_of_products; ?></span></h3>
      <p>Feltöltött termékek</p>
      <a href="products.php" class="btn">Megtekintés</a>
    </div>
    <div class="box text-align center">
      <?php
        $select_users = $conn->prepare("SELECT * FROM `users`");
        $select_users->execute();
        $number_of_users = $select_users->rowCount();
      >
      <h3><span><?php echo $number_of_users; ?></span></h3>
      <p>Felhasználók</p>
      <a href="users_accounts.php" class="btn">Megtekintés</a>
    </div>
  </div>
```

kivételt képez, hisz az első gomb az admin szó alatt, a 'Profil Szerkesztése', az egy másik panelre juttatja el a tulajdonost, mégpedig arra, ahol lehetősége van az épp bejelentkezett admin fiók szerkesztésére.

<?php ?> Közötti részek segítségével az

admin oldal megkapja a főoldal adatait, amik változnak, amint frissül valami adat odaát, azt egyből láthatjuk itt is, például ha egy új felhasználó regisztrál, vagy egy új terméket töltünk fel a kínálatba.


Admin panel (Termékek):



Termékek menüpontra kattintva betöltődnek az adatbázisba felvitt termékek, az összes adattal, képpel együtt.


A különbség itt az, hogy az aktuálisan bejelentkezett adminnak lehetősége van a termékek adatait szerkeszteni, ellenkező esetben, ha mondjuk elavult már a termék, nincs a raktárban, akkor szimplán törölni is.

FELTÖLTÖTT TERMÉK(EK)




ROLEX DAYTONA
\$12500/-
WATCH DETAILS Sku: X-RDYT07215
Box: Original Factory Box Papers: Original
Factory Papers Brand: Rolex Model:
Daytona Condition: Pre-Owned Reference
Number: 116500LN CASE & BEZEL: Case
Material: Stainless Steel Case Size: 40MM
Bezel: Ceramic Bezel Function: Tachymeter
DIAL: Dial Material: Index Hour Markers
BRACELET: Bracelet Material: Stainless
Steel Closure: Fold Over Clasp Closure
Material: Stainless-Steel FEATURES:
Water Resistance: 100 meters / 330 feet
Movement:

Szerkesztés Törlés



PATEK PHILIPPE NAUTILUS
\$65000/-
Model - Patek Philippe Nautilus
ORIGINAL PATEK MOVEMENT ONLY
Reference Number - 5711 Movement -
Automatic Bezel - Rose Gold Custom
Baguette Diamond Bezel Dial - Black
Custom Diamond Dial Case Size - 40mm
Crystal - Scratch-resistant Sapphire
Crystal Bracelet - Rose Gold Bracelet
Water Resistant - 10 ATM Gender - Men's

Szerkesztés Törlés



AUDEMARS PIGUET ROYAL OAK
\$52000/-
WATCH DETAILS: SKU: 4789327
Reference Number:
26730ST.OO.1320ST.02 Box: Yes Papers:
Yes Year: 2022 Manufactured In:
Switzerland Style: Casual, Dress Gender:
Mens CASE & DIAL: Case Size: 41mm
Case Material: Stainless Steel Dial Color:
Blue Hours Markers: Index Caseback:
Exhibition Case Shape: Octagon Water
Resistance: 50m FUNCTION: Movement:
Automatic Complications: Tourbillon
STRAP/BRACELET: Strap/Bracelet
Material: Stainless Steel Band Color: Silver
Buckle T

Szerkesztés Törlés

```
<div class="flex-btn">
  <a href="update_product.php?update=?= $fetch_products['id']; ?>" class="option-btn">Szerkesztés</a>
  <a href="products.php?delete=?= $fetch_products['id']; ?>" class="delete-btn" onclick="return confirm('Törli ezt a terméket?');">Törlés</a>
</div>
```

Szerkesztés gombra nyomva, átkerülünk az 'update_product.php'-ra és az adott termék ID-je is belekerül, hogy az általunk kiválasztott termék adatait tudjuk szerkeszteni, ezért fontos minden adatot jól megadni, amikor feltöltünk egy terméket, hogy ne legyen két azonos számú termékünk.

Ahogy átkerültünk a másik oldalra, élénk tárul a panel, a weboldalon is tudjuk, de természetesen adatbázison belül is lehet szerkeszteni a termék adatait, de én úgy gondolom, ez egy ízlésesebb módja. Minden egyes adatát tudjuk változtatni a terméknek. Nevét, az árát, a leírását (internetről szedtem őket), mind a három képet, külön-külön, de itt érdemes figyelni a nagyságra, mert egy bizonyos méret után nem engedi feltölteni a képet a program, ezért jobb, ha méretre van szabva, arányosan a weboldalhoz.

```
if(isset($_POST['update'])) {
    $pid = $_POST['pid'];
    $name = $_POST['name'];
    $name = filter_var($name, FILTER_SANITIZE_STRING);
    $price = $_POST['price'];
    $price = filter_var($price, FILTER_SANITIZE_STRING);
    $details = $_POST['details'];
    $details = filter_var($details, FILTER_SANITIZE_STRING);

    $update_product = $conn->prepare("UPDATE `products` SET name = ?, price = ?, details = ? WHERE id = ?");
    $update_product->execute([$name, $price, $details, $pid]);

    $message[] = 'Termék sikeresen frissítve!';

    $old_image_01 = $_POST['old_image_01'];
    $image_01 = $_FILES['image_01']['name'];
    $image_01 = filter_var($image_01, FILTER_SANITIZE_STRING);
    $image_size_01 = $_FILES['image_01']['size'];
    $image_tmp_name_01 = $_FILES['image_01']['tmp_name'];
    $image_folder_01 = '../uploaded_img/' . $image_01;
}
```

Bármelyik adatát frissítjük a terméknek és megnyomjuk a 'Frissítés' gombot, akkor az oldal kiírja nekünk, hogy a termék sikeresen frissült.

A képeknél kicsit másabb az egész, ott, ahogy említettem előbb, nem szabad átlépnie egy bizonyos határt, különben hibaüzenetet kapunk. Ha jó a méretek, az else szerkezetben unlinkeljük a régi képet, majd az új kép átveszi a helyét és kapunk egy visszaigazoló üzenetet, ismét fent.

```
if(!empty($image_01)) {
    if($image_size_01 > 2000000) {
        $message[] = 'A kép mérete túl nagy!';
    } else {
        $update_image_01 = $conn->prepare("UPDATE `products` SET image_01 = ? WHERE id = ?");
        $update_image_01->execute([$image_01, $pid]);
        move_uploaded_file($image_tmp_name_01, $image_folder_01);
        unlink('../uploaded_img/' . $old_image_01);
        $message[] = 'Első kép sikeresen frissítve!';
    }
}
```

Admin panel (Rendelések):

TELJESÍTENDŐ RENDELÉS(EK)

Létrehozva : 2023-04-19

Név : teszt

Telefonszám : 0630300300

Szállítási cím : Teszt Utca,
Tesztváros, Tesztország - 6000

Össztermék : PATEK PHILIPPE
NAUTILUS (65000 x 1) -

Összár : \$65000/-

Fizetési mód : Utánvétes fizetés

Törölés

A placed_orders.php megnyitása, vagyis a 'Rendelések' menüpont megnyitása után egy olyan panel tárul elénk, ahol a felhasználók által létrehozott, elküldött rendeléseiket láthatjuk, az összes adattal.

```
<section class="orders">
<h1 class="heading">Teljesítendő rendelés(ek)</h1>
<div class="box-container">
<?php
$select_orders = $conn->prepare("SELECT * FROM `orders`");
$select_orders->execute();
if($select_orders->rowCount() > 0){
    while($fetch_orders = $select_orders->fetch(PDO::FETCH_ASSOC)){
    <div class="box">
        <p> Létrehozva : <span><?=$fetch_orders['placed_on']; ?></span> </p>
        <p> Név : <span><?=$fetch_orders['name']; ?></span> </p>
        <p> Telefonszám : <span><?=$fetch_orders['number']; ?></span> </p>
        <p> Szállítási cím : <span><?=$fetch_orders['address']; ?></span> </p>
        <p> Össztermék : <span><?=$fetch_orders['total_products']; ?></span> </p>
        <p> Összár : <span><?=$fetch_orders['total_price']; ?></span> </p>
        <p> Fizetési mód : <span><?=$fetch_orders['method']; ?></span> </p>
        <div class="flex-btn">
            <a href="placed_orders.php?delete=?=$fetch_orders['id']; ?>" class="delete-btn" onclick="return confirm('Törli ezt a rendelést?');">Törölés</a>
        </div>
    </div>
    <?php
    }
    }else{
        echo 'Nincs még leadott rendelés!';
    }
    <?>
    </div>
</section>
```

Itt is php segítségével kikérjük az 'orders.php'-ban létrehozott rendelések adatait, amiket meg is jelenítünk az admin oldalon.

<p> Tagekben, spanokon belül \$fetch_orders

segítségével egyesével megtölti adatokkal az oldalt. Az adminnak ezen kívül lehetősége van még törölni a rendelést, hisz ez sokszor előfordulhat, hogy az adott vásárló meggondolja magát, téves rendelést adott. Egyszerű dolgunk van, gombunkra az order id-je alapján linkelve van egy törlési funkció, amit az adott felhasználó fiókjához van kapcsolva.

Ha pedig nincs még leadva rendelés, akkor az else{ részben ügyeltem erre is, az oldal csak egy üzenetet fog kiírni, '**Nincs még leadott rendelés!**'

Admin panel (Admin szerkesztés, törlés, létrehozás):

Admin panelnél kattintsunk az 'Adminok' pontra, hogy elérjük ezt a felületet.

Több lehetőségünk van itt, tudunk másoknak létrehozni admin profilt, illetve tudjuk törölni azt vagy szerkeszteni, az itt létrehozott fiókok, adatmódosítások elmentődnek nyilván az adatbázisba is.



ADMIN FIÓK(OK)

Új fiók létrehozása

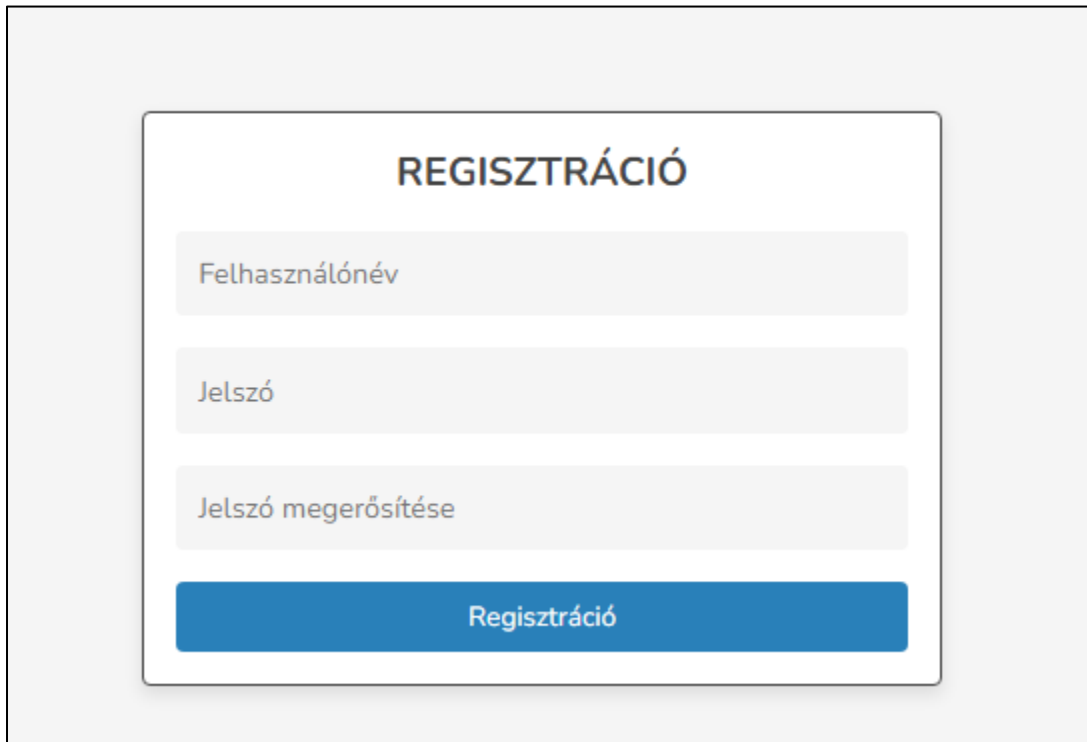
Admin Regisztráció

id : 1

név : admin

Törlés Szerkesztés

Admin regisztráció:



REGISZTRÁCIÓ

Felhasználónév

Jelszó

Jelszó megerősítése

Regisztráció

Admin fióknál kevesebb adat szükséges, csak egy felhasználónévre és egy jelszóra van szükségünk, ez nem e-mailhez kötött.

```
<section class="form-container">
<form action="" method="post">
<h3>Regisztráció</h3>
<input type="text" name="name" required placeholder="Felhasználónév" maxlength="20" class="box" oninput="this.value = this.value.replace(/s/g, '')">
<input type="password" name="pass" required placeholder="Jelszó" maxlength="20" class="box" oninput="this.value = this.value.replace(/s/g, '')">
<input type="password" name="cpass" required placeholder="Jelszó megerősítése" maxlength="20" class="box" oninput="this.value = this.value.replace(/s/g, '')">
<input type="submit" value="Regisztráció" class="btn" name="submit">
</form>
</section>
```

Gombunkra issetelünk, amivel több dolgot is ellenőrizünk általa. A kódsor figyel, hogy ne legyen két azonos nevű adminisztrátori fiók, másrészt, hogy a két jelszó biztosan megegyezik-e.

Ha sikeres, erről is kapunk üzenetet, ismét.

```
if(isset($_POST['submit'])){
    $name = $_POST['name'];
    $name = filter_var($name, FILTER_SANITIZE_STRING);
    $pass = sha1($_POST['pass']);
    $pass = filter_var($pass, FILTER_SANITIZE_STRING);
    $cpass = sha1($_POST['cpass']);
    $cpass = filter_var($cpass, FILTER_SANITIZE_STRING);

    $select_admin = $conn->prepare("SELECT * FROM `admins` WHERE name = ?");
    $select_admin->execute([$name]);

    if($select_admin->rowCount() > 0){
        $message[] = 'Felhasználónév már létezik!';
    }else{
        if($pass != $cpass){
            $message[] = 'Jelszavak nem egyeznek meg!';
        }else{
            $insert_admin = $conn->prepare("INSERT INTO `admins` (name, password) VALUES(?,?)");
            $insert_admin->execute([$name, $cpass]);
            $message[] = 'Sikeres admin regisztráció!';
        }
    }
}
```

Regisztrált admin fiók az alap admin fiók mellett:

<div>Új fiók létrehozása</div> <div>Admin Regisztráció</div>	<div>id : 1</div> <div>név : admin</div> <div>Törlés Szerkesztés</div>	<div>id : 3</div> <div>név : teszt</div> <div>Törlés</div>
--	--	--

Szerkesztés/törlés:

Fő adminisztrátori fióknál van csak lehetőségünk a szerkesztésre, a később regisztrált fiókoknál csak a törlés opció van játékban.

Ezt a program az isset betöltése segítségével tudja megállapítani, ahogy mondtam, minden fiókhoz van saját azonosító, így belépés után tudjuk, hogy melyik admin fiók kié, illetve, hogy melyik az alap fiók.

```
$admin_id = $_SESSION['admin_id'];

if(!isset($admin_id)){
    header('location:admin_login.php');
}

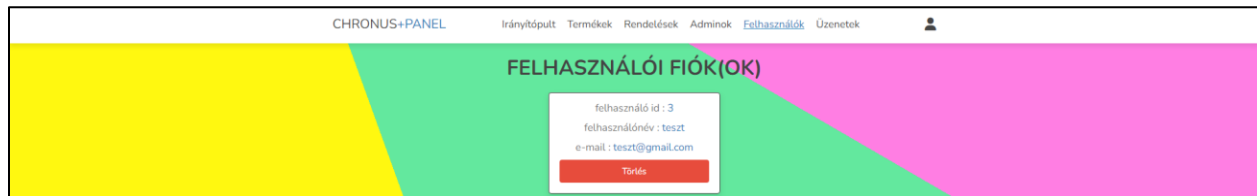
if(isset($_GET['delete'])){
    $delete_id = $_GET['delete'];
    $delete_admins = $conn->prepare("DELETE FROM `admins` WHERE id = ?");
    $delete_admins->execute([$delete_id]);
    header('location:admin_accounts.php');
}

?>
```

```
<a href="admin_accounts.php?delete=?= $fetch_accounts['id']; ?>" onclick="return confirm('Törli ezt a fiókot?')" class="delete-btn">Törlés</a>
```

Admin panel (Felhasználók):

Ennél a panel résznél a regisztrált fiókokat láthatjuk, természetesen a személyes adatok nélkül (jelszó).



<p> Tageknél látható, hogy az adatokat fetch_accounts segítségével a megadott adatokat behúzzuk az oldalra, jelen esetben itt a felhasználó id-je, neve és e-mail címe kell számunkra.

```
<section class="accounts">
  <h1 class="heading">Felhasználói fiók(ok)</h1>
  <div class="box-container">
    <?php
      $select_accounts = $conn->prepare("SELECT * FROM 'users'");
      $select_accounts->execute();
      if($select_accounts->rowCount() > 0){
        while($fetch_accounts = $select_accounts->fetch(PDO::FETCH_ASSOC)){
          <div class="box">
            <p> felhasználó id : <span>= $fetch_accounts['id']; ?&gt;&lt;/span&gt; &lt;/p&gt;
            &lt;p&gt; felhasználónév : &lt;span&gt;<?= $fetch_accounts['name']; ?&gt;&lt;/span&gt; &lt;/p&gt;
            &lt;p&gt; e-mail : &lt;span&gt;<?= $fetch_accounts['email']; ?&gt;&lt;/span&gt; &lt;/p&gt;
            &lt;a href="users_accounts.php?delete=?= $fetch_accounts['id']; ?&gt;" onclick="return confirm('Törli ezt a fiókot? Felhasználóval kapcsolatos információk is törlésre kerülnek!')" class="delete-btn"&gt;Törlés
          &lt;/div&gt;
        }
      }
      else{
        echo 'Nincs elérhető felhasználói fiók!';
      }
    &lt;/?php&gt;
  &lt;/div&gt;
&lt;/section&gt;</pre

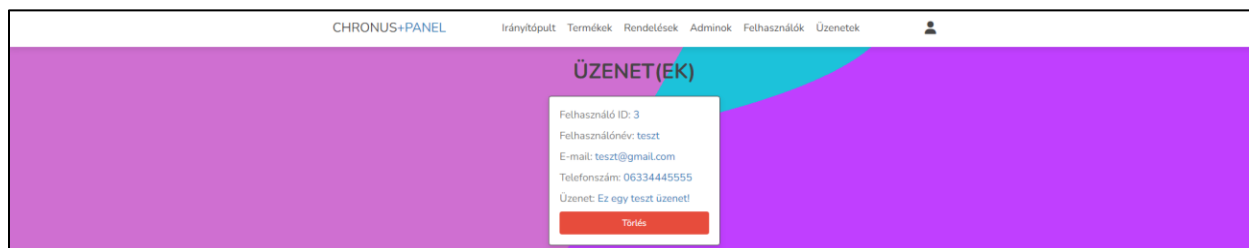
```

Törlés gomb törli az egész fiókot a fiókhoz kapcsolt ID által (\$fetch_accounts['id'] , így egy adat alapján tudjuk az egész fiókot kitörölni az adatbázisból. Az SQL lekérdezések végén mindig az id-t keresi, például \$delete_user=\$conn->prepare("DELETE FROM 'users' WHERE id = ?");

```
if(isset($_GET['delete'])){
  $delete_id = $_GET['delete'];
  $delete_user = $conn->prepare("DELETE FROM `users` WHERE id = ?");
  $delete_user->execute([$delete_id]);
  $delete_orders = $conn->prepare("DELETE FROM `orders` WHERE user_id = ?");
  $delete_orders->execute([$delete_id]);
  $delete_messages = $conn->prepare("DELETE FROM `messages` WHERE user_id = ?");
  $delete_messages->execute([$delete_id]);
  $delete_cart = $conn->prepare("DELETE FROM `cart` WHERE user_id = ?");
  $delete_cart->execute([$delete_id]);
  $delete_wishlist = $conn->prepare("DELETE FROM `wishlist` WHERE user_id = ?");
  $delete_wishlist->execute([$delete_id]);
  header('location:users_accounts.php');
}
```

Admin panel (Üzenetek):

Ez a panel szinte azonos elven működik admin részről, mint az előbb leírt 'Felhasználók' panel, azonos módon megkapjuk a felhasználói oldalról a felhasználói adatokat és az írt üzenetet, bármilyen gondtal kapcsolatban.



Megjelenik előttünk a felhasználó azonosítója, a felhasználóneve, e-mail címe, telefonszáma és az általa írt üzenet, ezeket mind \$fetch_message segítségével kapjuk meg a 'messages' SQL táblából, ahova a főoldalon a felhasználó feltöltötte, itt pedig lekérjük ezeket.

Ha nincs semmilyen üzenet a felhasználók által elküldve, akkor csak egy üres sor fogad minket, **'Nincs megjeleníthető üzenet'!**.

```
<section class="contacts">

<h1 class="heading">Üzenet(ek)</h1>

<div class="box-container">

    <?php
        $select_messages = $conn->prepare("SELECT * FROM `messages`");
        $select_messages->execute();
        if($select_messages->rowCount() > 0){
            while($fetch_message = $select_messages->fetch(PDO::FETCH_ASSOC)){
                <div class="box">
                    <p>Felhasználó ID: <span><?= $fetch_message['user_id']; ?></span></p>
                    <p>Felhasználónév: <span><?= $fetch_message['name']; ?></span></p>
                    <p>E-mail: <span><?= $fetch_message['email']; ?></span></p>
                    <p>Telefonszám: <span><?= $fetch_message['number']; ?></span></p>
                    <p>Üzenet: <span><?= $fetch_message['message']; ?></span></p>
                    <a href="messages.php?delete=<?= $fetch_message['id']; ?>" onclick="return confirm('Törli ezt az üzenetet?');" class="delete-btn">Törés</a>
                </div>
            <?php
            }
        }
        else{
            echo 'Nincs megjeleníthető üzenet!</p>';
        }
    <?php
    </div>

</section>
```

```
if(isset($_GET['delete'])){
    $delete_id = $_GET['delete'];
    $delete_message = $conn->prepare("DELETE FROM `messages` WHERE id = ?");
    $delete_message->execute([$delete_id]);
    header('location:messages.php');
}
```

Törés gombra nyomva pedig \$delete_message-ben meg van írva, hogy törölje a 'messages' táblából az adott ID-val ellátott

üzenet, ezt a felhasználó észre sem veszi.

Jövőben fejleszthető részek:

Az oldal nem tökéletes, de pár dolgot a jövőre nézve lehet fejleszteni benne:

Az első ami eszembe jut, hogy létre lehetne hozni az oldalra egy automatikus botot, ami fel van töltve alap kérdésekkel és válaszokkal (hasonló, mint pl egy Facebook oldal esetén) ami közül a vásárló tud választgatni és a bot automatikus választ ad pár másodpercen belül,

Véleményeket tudnának írni a regisztrált felhasználók a termékekről, oldalról.

Több nyelvre fordítani az oldalt és egy nyelvválasztót beépíteni az oldal header részébe, hogy a külföldi vásárlók is tudjanak eligazodni az oldalon.

Több nyelv mellett több pénznemet is felvinni az adatok közé és szintén a header részbe egy devizaváltó lehetőséget beállítani.

Lehetőség szerint az oldalhoz adni egy automatikus e-mail küldő plugint beépíteni, hogy a regisztrált felhasználók miután elfogadják a feltételeket, értesülni tudjanak az aktuális akciókról vagy új termékekről, márkákról.

Felhasznált források:

<https://www.youtube.com/?hl=hu&gl=HU>

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

<https://getbootstrap.com/>

<https://blog.prototypr.io/css-only-multi-color-backgrounds-4d96a5569a20>

<https://www.remove.bg/>