

# Visszatérő Erő Sportegyesület

## honlapcsalád

Készítette: Tóth-Sárközi Viktória  
– 06135001 számú Webfejlesztő szakmai képzés

### **Bevezető**

Az alábbi weboldal a Visszatérő Erő Sport Egyesületet mutatja be. Magát az egyesület 2019. júniusában hoztuk létre, hogy esélyt adhassunk másoknak. Megalapítását Sztanek Sándor kezdeményezte, akit azok az élmények inspiráltak, amelyeket a versenyzés elkezdésekor átélt. Profi sportolóként pedig kötelességének érezte, hogy példát mutasson, és felkutassa az ifjú tehetségeket. A Visszatérő Erő egy teljesen hivatalos egyesület, amely magába foglalja a Karbirkózást és az Erőemelést. Az egyesületben társtulajdonosi státuszt töltök be.

A célja az egyesületnek és a honlapnak, hogy elérje azokat a szkanderozás iránt érdeklődőket, akik reményeink szerint az Egyesülettel együttműködnek majd, vagy tagok lesznek.

A honlap megtervezésekor (tartalmi és technikai szempontból egyaránt) kiemelten figyeltem, hogy életszerű, valódi problémára nyújtson megoldást. Erre is szolgál az egyesület bemutatása, a képek, hírek és a naptár, valamint a kapcsolatfelvételt elősegítő megoldás. A modern követelményeknek megfelelően adattárolási és -kezelési funkciókat is megvalósít, valamint tartalmaz RESTful architektúrának megfelelő szerver és kliens oldali komponenseket is. Továbbá szerveroldali adminisztrációs felületet is használ. A kliens oldali komponens egyaránt alkalmas asztali és mobil eszközökön történő használatra. Törekedtem arra, hogy a forráskód az ún. tiszta kód elveinek megfelelően készüljön.

## Technikai megoldások

A honlap reszponzív és tartalmaz javascripteket, valamint a formázások CSS-ben vannak. Mivel nem tárhelyen (tárhelyre) lettek kifejlesztve, hanem a vizsga kiírása szerint localhostra illetve Github-ra, így a menüpontokon található hivatkozások relatívak (pl. Főlap menü: ...\\web\\index.php). Ez nem szerencsés, de „élő”, valós esetben minden hivatkozás, link abszolút hivatkozás lenne (pl. Főlap menü: <https://www.domainnev.hu/index.php>).

A formázási beállítások túlnyomó többségét a visszateroero.css stílusállományban, a kódolási feladatokat pedig a visszateroero.js JavaScript fájlban végeztem el.

Az oldalak fejlécében többek között a szerzői és egyéb metaadatok, a nyelv és kódlap megadását szolgáló, valamint az oldal faviconját megjelenítő parancsokat, a bootstrep-re és visszateroero.css-re, valamint a weboldal végén a visszateroero.js-re mutató hivatkozásokat helyeztem el. Ugyanitt oldottam meg, hogy az egyesület neve jelenjen meg a böngésző címsorában megjelenő címként.

Kialakítottam továbbá a weboldal navigációs részét, létrehoztam a menüelemeket számozatlan lista segítségével, a lista külső HTML-eleme (<ul>) navbar-nav és ml-auto, a listaelemek pedig nav-item osztálykijelölővel lettek formázva. Az egyes listaelemekhez a hivatkozásokat is létrehoztam, melyek a külső, a honlap család más oldalain kívül a főoldal támogatóink, rolunk és kapcsolat dokumentum szintű azonosítóira mutatnak! A hivatkozásokat a nav-link, rounded és js-scroll-trigger osztálykijelölőkkel formáztam meg.

A visszatero.js fájlban egy olyan JavaScript kódot hoztam létre, ami a weboldal kapcsolat szekciójában található információkat (név, e-mail-cím, telefonszám, üzenet) elküldi a webszervernek, ha az „Üzenet küldése” parancsgombra kattintunk! A REST-kérés paraméterei: Kérés típusa: POST. URL: /api/kapcsolat. Az elküldött adat típusa: JSON. Az elküldött adat szerkezete: name:String, email:String, phone:String, message:String. A

választípus: JSON. A válaszüzenet sikeres küldés esetén: {feldolgozva:boolean}  
Amennyiben valamelyik mező értéke üres, akkor a „Kérjük, minden mezőt töltsön ki az űrlapon!” üzenetet jeleníti meg egy felugró ablakban, és nem küldi be az űrlap adatait a szervernek! Amennyiben a küldés sikeres (200-as státuszkód), törölje a weboldalon a beírt adatokat, és egy felugró ablakban jelenít meg üzenetet.

Továbbá tartalmaz a fájl egy olyan JavaScript programkódot, amely a számlalo azonosítójú <span> elembe 0-ról indulva fél másodpercenként egy 1-gyel növelt számot illeszt be és a weboldal betöltődésekor indul el. A fél másodperces intervallumhoz a setInterval(functionName, interval) függvényt használja.

### **A honlapcsalád felépítése**

#### **1. Főoldal (Kezdőlap)**

A honlapcsalád tagjai a Főoldal (Kezdőlap), a Naptár, a Galéria. Ahogy említettem a főlap tartalmaz hivatkozásokat a HTML fájl belsejébe is, a böngésző a hivatkozott résztől kezdi el megjeleníteni a dokumentumot. Ezek a „Rólunk”, a „Támogatók”, „Kapcsolat”. Ehhez a hivatkozás célpontjában megadtam a HTML tagek id attribútumait, vagyis a tagok egyedi azonosítóit.

Hangsúlyosan jelenik meg a „Visszatérő Erő Sport Egyesület” név is a kezdőoldalon. Ahogy a felhasználó lejjebb görget, a Híreknél a legaktuálisabb esemény, hír jelenik meg az egyesülettel kapcsolatban. A hír-rész php segítségével adatbázisból töltődik fel, amelyet az Adminisztrációs lapon keresztül lehet megadni.

A kezdőlapon láthatóak még a támogatók is, akik hozzásegítenek minket ahhoz, hogy tovább gyarapíthassuk a sikereinket. Az egyesületnek jelenleg négy támogatója van.

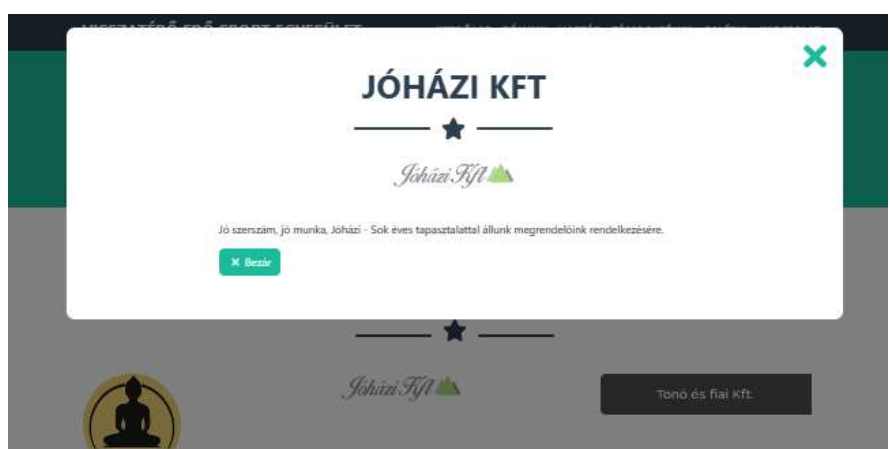
Az oldal legalján találhatóak azok a közösségi oldalak, ahol megtalálható az egyesület. Az első a Youtube, ahol versenyre való gyakorló videók vannak fent. Természetesen Facebookon is megtalálhatóak vagyunk, ahol még naprakészebbek



vagyunk, még több hír és kép található meg. A TikTok nagy sikerére tekintettel az egyesületnek is van TikTok oldala, felhasználója.

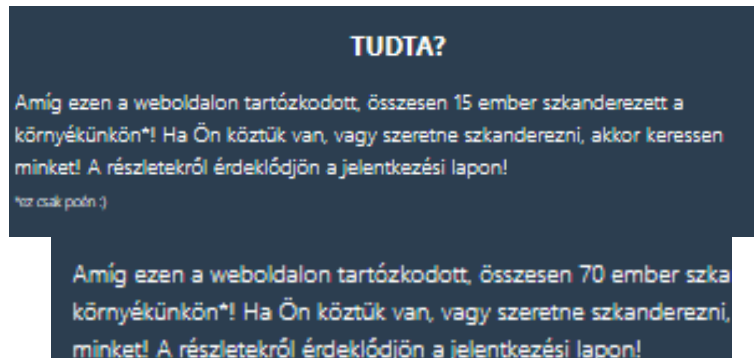
A Rólunk oldalrészben van az egyesület története és a tagsági díj is fel van tüntetve.

Szót szeretnék ejteni pár érdekesebb technikai megoldásról. Az egyik a speciális CSS/HTML kód által működő felugró kártyák, amelyek az adott támogató logójára kattintva lépnek működésbe.



Egy-egy kártyán rövid ismertető olvasható a cég tevékenységéről, valamint a logóra kattintva megnyílik a támogató weboldala.

A másik a Footerben lévő javascript alapú számláló, egy apró poén, geg.

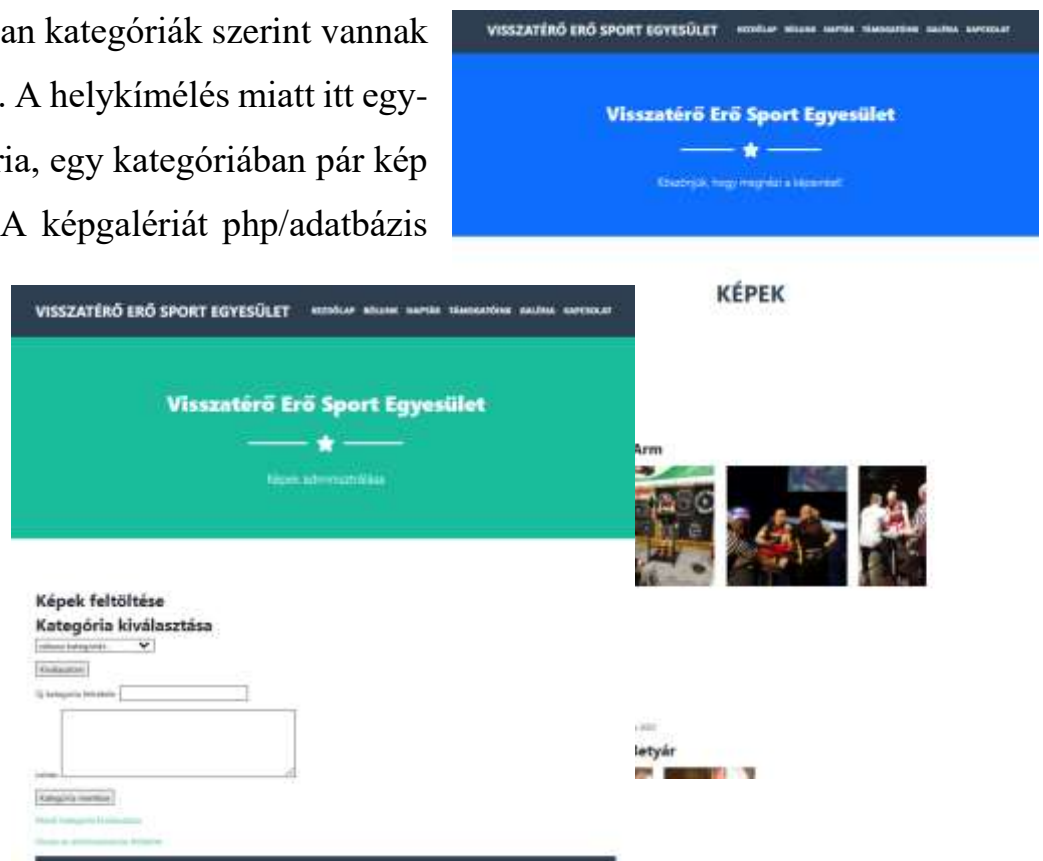


## 2. Naptár oldal

Az egyesület Gmail-es email címmel rendelkezik. A szolgáltatáscsomag része a Naptár, amelynél lehetőség van arra, hogy annak tartalmát megjelenítsük egy weboldalon. Ezt használtam ki. A reszponzivitás megőrzése érdekében a Google által biztosított „iframe” kapott tőlem egy „container” CSS osztályjelölőt.

## 3. Galéria

A Galériában kategóriák szerint vannak rendezve a képek. A helykímélés miatt itt egy-két minta kategória, egy kategóriában pár kép látható jelenleg. A képgalériát php/adatbázis alapon működik és az Adminisztrációs lapon lehet kezelni is.



#### 4. Adminisztrációs felület

A bejelentkezés után itt van lehetősége az arra jogosult felhasználónak, hogy képet, hírt feltöltsön, képkategóriákat létrehozzon, módosítson, vagy új adminisztrátorokat vegyen fel.

A honlapcsalád fenti részei adatbázison alapulnak. Ennek rövid bemutatásával zárom ezt a dokumentációt.

#### Az adatbázis felépítése és működése

Relációs adatmodellen alapuló adatbázist készítettem. A modell kialakítását alapvetően az indokolta, hogy az adatbázis megtervezésekor az adatokat a felhasználó számára áttekinthető és egyszerűen kezelhető táblázatokba lehessen elhelyezni, amely hasonló a hagyományos adatszervezés rekordstruktúrájához.

A XAMPP program által biztosított helyi (lokális) szerveren hoztam létre a *galeria* nevű adatbázist.

Az adatbázis a következő táblákat tartalmazza:

galeria hirek	galeria categoria	galeria kepek	galeria admin
id : int(11)	id : int(11)	id : int(11)	azon : varchar(100)
neve : varchar(100)	neve : varchar(100)	neve : varchar(250)	jelszo : varchar(250)
leiras : text	leiras : text	kategoria_id : int(11)	
aktualis : tinyint(4)	aktualis : tinyint(4)	sorszam : int(11)	

admin(azon, jelszo)

hirek(id, neve, leiras, aktualis)

kategoria(id, neve, leiras, aktualis)

kepek(id, neve, kategoria\_id, sorszam)

Idegen kulcsnak nem állítottam be a kategoria\_id-t, mert csak a törlésnél számít. Nem enged kategóriát törölni, ha van benne kép. A törlés úgy történik, hogy előtte az összes képet kitörli.

Táblák létrehozása:

```
CREATE TABLE admin (  
    azon varchar(100) NOT NULL,  
    jelszo varchar(250) NOT NULL  
) ENGINE=InnoDB
```

```
CREATE TABLE kategoria (  
    id int AUTO_INCREMENT PRIMARY KEY,  
    neve varchar(100) NOT NULL,  
    leiras text NOT NULL,  
    aktualis tinyint(1) NOT NULL DEFAULT '1'  
) ENGINE=InnoDB
```

```
CREATE TABLE kepek (  
    id int AUTO_INCREMENT PRIMARY KEY,  
    neve varchar(250) NOT NULL,  
    kategoria_id int(11) NOT NULL,  
    sorszam int(11) NOT NULL  
) ENGINE=InnoDB
```

```
CREATE TABLE hirek (  
    id int AUTO_INCREMENT PRIMARY KEY,  
    neve varchar(100) NOT NULL,  
    leiras text NOT NULL,  
    aktualis tinyint(1) NOT NULL DEFAULT '1'  
) ENGINE=InnoDB
```