

**Zsámbéki Premontrei Gimnázium, Szakgimnázium és
Szakközépiskola**

**Az 54 481 02 számú Gazdasági informatikus szakképesítés
záródolgozata**

**Személyre szabott weboldal
készítése egy természetvédelemmel
foglalkozó, fiktív, nonprofit
alapítvány számára**

Készítette:
Fodor Vince

**Gyermely
2017**

Tartalomjegyzék

1.	Bevezetés	3
1.1.	Témaválasztás indoklása	3
1.2.	Szakdolgozat célja.....	3
1.3.	Alkalmazott módszerek.....	3
2.	Weboldalak történelme, jellemzői.....	4
3.	Weboldalak struktúrája, működése, egyéb kapcsolódó leíró nyelvek	6
3.1.	HTML.....	6
3.2.	HTML5.....	7
3.3.	CSS	8
3.4.	PHP.....	10
3.5.	Java, Java applet, JavaScript.....	12
4.	A fiktív alapítvány igényei	14
5.	Weboldal készítés módszerei, lehetőségek.....	15
5.1.	Forráskód szerkesztők.....	15
5.2.	WYSIWYG szerkesztők	17
5.3.	Webes sablonok	18
5.4.	CMS rendszerek.....	20
6.	A személyre szabott weboldalak kialakításával kapcsolatos döntések	23
7.	Hibakeresés, tesztelés.....	25
8.	Domain név regisztráció.....	28
8.1.	Lehetőségek, szempontok.....	28
9.	Webtárhely kiválasztása.....	29
9.1.	Lehetőségek, szempontok.....	29
9.2.	Az optimális választás	30
10.	Honlap ismerteté tétele, keresőmarketing tevékenységek.....	32
10.1.	Keresőmarketing.....	32
10.2.	Honlap optimalizálása a keresőrobotok felé	33
10.3.	A domain név kiválasztásának jelentősége	35
10.4.	Külső hivatkozások.....	35
11.	A webhely jövőbeni továbbfejlesztési lehetőségei	37
12.	Irodalom és forrásjegyzék.....	38

1. Bevezetés

1.1. Témaválasztás indoklása

Témaválasztásomnál fontos szerepet játszott, hogy rajongok a gyönyörű és rezponzív weboldalakért. Az én meglátásom szerint ez egy külön művészeti ágazat a digitális világban. Közreműködött a webdesign és a webfejlesztés iránti érdeklődésem is a téma választásában.

1.2. Szakdolgozat célja

A szakdolgozatom célja hogy bemutassam a képzés során szerzett ismereteim a weboldallal és a hozzá kapcsolódó informatikai eszközökkel kapcsolatban, valamint ezek alapján egy személyre szabott weboldalt készítsek a kitalált cégem tematikájának megfelelően.

1.3. Alkalmazott módszerek

Szekunder kutatás: Internetes tájékozódás, információszerzés weboldalfejlesztéssel foglalkozó portálokról illetve fórumokról.

2. Weboldalak történelme, jellemzői

1991 derekán, amikor még csak a legelső szálát szőtte a Világháló a CERN és a FermiLab között, alig voltak néhányan, akik hallottak volna a World Wide Webről. Ma viszont, a kultúra minden területét behálózza bolygónkat, és átfogó kommunikációs közeget teremt az emberek millióinak.

Az az eszköz, amely az egymástól távol dolgozó kutatók közötti adatmegosztást volt hivatott segíteni, a nagyenergiájú fizika legjelentősebb modern vívmányaként tartható számon. A Web 1990-ben született meg, amikor a CERN (Európai Részecskefizikai Kutatóközpont) egyik informatikusa, Tim Berners-Lee, hozzáfogott egy olyan kódrendszer-
- protokoll - megalkotásához, melynek segítségével az egyes számítógépek más számítógépekkel kapcsolatba léphetnek anélkül, hogy foglalkozniuk kellene a különböző adatbázisok összeférhetetlenségéből és a nehézkes bejelentkezési procedúrákból adódó problémákkal

Kitalálta, hogy a hálózaton terjesztett információt hogyan lehetne az egész világ számára könnyen elérhetővé tenni, és hogyan kellene kidolgozni a módszer alapjait, azaz a protokollokat.

A Web első, belső kutatói használatra szánt változata a CERN fizikusainak munkáját tette könnyebbé: adatokat, eredményeket, dokumentumokat oszthattak meg egymással saját hálózatukon belül. A kutatóközpontból kifelé irányuló kapcsolat (például a CERN és a FermiLab között) és így az információ más tudósokkal való megosztása ekkor még nehéz feladatnak bizonyult.

Mielőtt Berners-Lee megírta volna a hipertext-továbbítási protokollokat, minden egyes számítógépre különböző módon kellett bejelentkezni. Ezek a kiszolgálók különböző felépítésű adatbázisokkal rendelkeztek, ami a rendszerek közti kommunikációt igen nehézkessé tette. Berners-Lee szoftvere (a Web) ledöntötte a hardvert (Internet) ilyen módon megosztó gátakat.

Az Egyesült Államokban Berners-Lee a rákövetkező nyáron tartotta első előadását a World Wide Web-ről.

Vicky White, a számítógépes részleg helyettes vezetője szerint "valójában Ruth érdemli az elismerést, amiért felfedezte a Web-et. Mindketten ismertük Tim Berners-Lee munkáját, de ő volt az, aki először érdeklődést mutatott iránta.

A Fermilab már a korai időszakban csatlakozott a World Wide Web hálózatra. A Sloan Digital Sky Survey elnevezésű, az égboltot feltérképező projekt -- amelyben a laboratórium is közreműködött -- az elsők között használta ki a Web lehetőségeit 1992-ben.

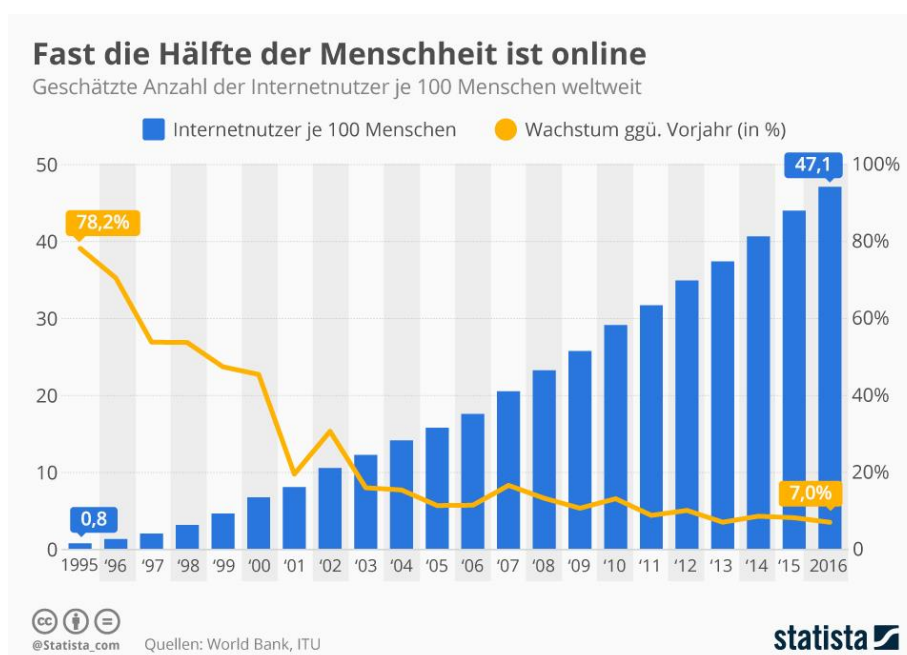
"A projektben egymástól nagyon távol lévő kutatók vettek részt. Úgy éreztük, hogy ha az információkhoz könnyen hozzáférhetnénk, akkor hatalmas segítséget kapnánk." -- így emlékszik vissza Pordes.

A nagyenergiák fizikáján túl

Berners-Lee-vel együtt mások is hamar rájöttek, hogy a Web felhasználási területe a nagyenergiák fizikáján túlmutatott.

"Mindezek után Berners-Lee elkezdett a World Wide Web számára protokollokat kidolgozni, amelyek segítségével nem csak a fizikusok kommunikálhattak egymással, hanem a világ minden tájáról elérhetővé váltak az információk." -- nyilatkozza a számítógépes részleg vezetője.

A következő lépés az volt, hogy a technológiát szélesebb kör számára tegyék elérhetővé. Manapság az Internet-használók általában valamelyik böngészőprogram (pl. a Netscape Navigator vagy a Web Explorer) segítségével szörföznek a World Wide Web hálózaton. 1992-ben azonban még egyik egeres-kattintós program sem létezett. Előfutáruk egy Mosaic nevű szoftver volt. (Eric Berger, 1996)



1. ábra Az Internethasználat alakulása 1995 - 2016.

https://infographic.statista.com/nomal/infografik_6897_anzahl_der_internetnutzer_je_100_menschen_weltweit_n.jp
g

3. Weboldalak struktúrája, működése, leíró nyelvek

A dolgozatom keretében elkészített honlap megalkotásánál a következőkben számba vett webes technológiákat alkalmaztam. A példa kódok az általam létrehozott honlap forráskódjaiból valók.

3.1. HTML

HTML (Hyper Text Markup Language) azaz hiperszöveges jelölőnyelv: Egy leíró nyelv, melyet weboldalak készítéséhez fejlesztettek ki, és mára már internetes szabvánnyá vált a W3C (World Wide Web Consortium) támogatásával. Az aktuális változata a 4.01, mely az SGML általános jelölőnyelv egy konkrét alkalmazása (azaz minden 4.01-es HTML dokumentum egyben az SGML dokumentumszabványnak is meg kell, hogy feleljen). Ezt tervek szerint lassan kiszorította volna az XHTML (ami a szintén SGML alapú XML leíró nyelven alapul), de az is lehetséges, hogy a HTML 5 veszi át a helyét. (wikipedia.org, 2016) Egy HTML állomány három fő részre bontható:

A Dokumentum Típus Definíció az állomány legelején, pl:

```
<!DOCTYPE html>
```

a HTML fejléc `<head></head>`, ami technikai és dokumentációs adatokat tartalmaz, amelyeket az internet böngésző nem jelenít meg, tehát átlag felhasználó ezeket nem látja és a HTML törzs `<body></body>`, amely a megjelenítendő információkat tartalmazza.

Tehát egy internetes oldal alapszerkezete a következőképpen nézhet ki:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<title>Természetvédelem</title>
```

```
<meta name="robots" content="all">
```

```

        <meta name="keywords" content="alapítvány, természetvédelem, erdő,
védett,,natura 2000, öko, erdővédelem, inváziós fajok, környezet, szennyeződés,
hulladékanyag, légszennyezés, levegő szennyezés, gépjárművek, élővilág, madarak,
denevérek, gomba, szarvasbogár, fakitermelés, erdészet, faanyag, biodiverzitás, nonprofit,
">

        <meta name="description" content="Weboldal természetvédelemről,
légszennyezésről, erdővédelemről">
        <meta http-equiv="Content-Type" content="text/html; charset=windows-
1250">
        <link rel="stylesheet" href="stilus.css" type="text/css"/>
    <head>
        <title>Öko gondolatok</title>
        <!--esetleges további fejléc-információk-->
    </head>
    <body>
        <p>első bekezdés</p>
        <p>második bekezdés</p>
    </body>
</html>

```

A HTML nem programozási nyelv, hanem kódnyelv, aminek segítségével akár a Jegyzetömb segítségével is alkothatóak weboldalak. A hipertext jelenti valójában az interneten található oldalakat (dokumentumokat), amelyek szöveg, kép, videó, hang, animáció, vagy ezek valamilyen kombinációjából áll. A HTML ezeknek a dokumentumoknak az elrendezését, formázását tartalmazza saját jelölőnyelvén. Bár a honlap készítő programok grafikai felületet is biztosítanak a weblap elkészítéséhez, mellette a legtöbb programnál hipertextként is kialakítható egy oldal. (Máté Balázs, 2016)

3.2. HTML5

A HTML5 a következő, jelentősen átdolgozott változata a HTML-nek (Hypertext Markup Language), a web fő jelölőnyelvének. Egyik fő tervezési célja, hogy a webes alkalmazásokhoz ne legyen szükség pluginek (pl. Adobe Flash, Microsoft Silverlight, Sun JavaFX) telepítésére. A specifikáció a HTML4 és az XHTML1 új verzióját jelenti, a hozzájuk

tartozó DOM2 HTML API-val együtt. A HTML5 specifikációban leírt formátumba történő migráció HTML4-ről, vagy XHTML1-ről a legtöbb esetben egyszerű, mivel a visszamenőleges kompatibilitás biztosított. A specifikáció a közeljövőben támogatni fogja a Web Forms 2.0 specifikációt is. HTML5-ben lehetőség van audio és video fájlok beillesztésére, erre a célra külön tagek vannak (<audio>, <video>). Továbbá létezik egy úgynevezett <canvas> tag, mely egy vászon. Erre Javascript segítségével lehet rajzolni, vagy képet beilleszteni. Fontos megemlíteni, hogy a HTML5-ből kivették az elrendezés megadó tulajdonságokat. Ezeket CSS-ben kell megadni. (wikipedia.org, 2016)

3.3. CSS

CSS (Cascading Style Sheet) egy stílusleíró nyelv, mely a HTML vagy az XHTML típusú strukturált dokumentumok megjelenését írja le. Ezen kívül használható bármilyen XML alapú dokumentum stílusának leírására is, mint például az SVG, XUL stb.

A CSS specifikációját a World Wide Web Consortium felügyeli.

A CSS-t a weblapok szerkesztői és olvasói egyaránt használhatják, hogy átállítsák vele a lapok színét, betűtípusait, elrendezését, és más megjelenéshez kapcsolódó elemeit. A tervezése során a legfontosabb szempont az volt, hogy elkülönítsék a dokumentumok struktúráját (melyet HTML vagy egy hasonló leíró nyelvben lehet megadni) a dokumentum megjelenésétől (melyet CSS-sel lehet megadni). Az ilyen elkülönítésnek több haszna is van, egyrészt növeli a weblapok használhatóságát, rugalmasságát és a megjelenés kezelhetőségét, másrészt csökkenti a dokumentum tartalmi struktúrájának komplexitását. A CSS ugyancsak alkalmas arra, hogy a dokumentum stílusát a megjelenítési módszer függvényében adja meg, így elkülöníthető a dokumentum formája a képernyőn, nyomtatási lapon, hangos böngészőben (mely beszédszintetizátor segítségével olvassa fel a weblapok szövegét), vagy Braille készüléken megjelenítve.

A CSS egyszerű szintaxissal rendelkezik, csak néhány angol nyelvű kulcsszót használ a stílusok tulajdonságaihoz. A stíluslap maga a stílust leíró szabályok sora. Minden szabályhoz tartozik egy szelektor és egy deklarációs szakasz. Ez utóbbi kapcsos zárójelek között pontosvesszővel elválasztott deklarációkat tartalmaz. A deklarációk formája a következő: a tulajdonság neve, egy kettőspont, majd az adott tulajdonság értéke. (wikipédia, 2016)

Például:

```
body{  
  background-color: #313A3F;  
  background-image: url("hatter.jpg");  
  background-size: auto;  
  background-position: top;  
  background-repeat: no-repeat;  
  background-attachment: fixed;  
  padding-bottom: 20px;  
}
```

```
div.oldal{  
  size: 90%;  
  width: 1152px;  
  size: 90%;  
  background-color: transparent;  
  margin-left: auto;  
  margin-right: auto;  
}
```

A stíluslapok már a HTML kezdete, 1990 óta jelen vannak valamilyen formában. A böngészők a stílus módosítására létrehozták saját leíró nyelvüket, melyet a webes dokumentumok megjelenésének módosítására lehetett használni. Eredetileg a stíluslapokat a felhasználók használták, mivel a HTML korai verziói még csak kevés prezentációs attribútumot tartalmaztak, így gyakran bízták a felhasználóra, hogy a webes dokumentumok hogyan jelenjenek meg.

A web fejlesztők igényei folyamatosan növekedtek a stilisztikai lehetőségek irányába, így a HTML nyelvbe egyre több ilyen elem került. Ilyen lehetőségek mellett a stíluslapok kevésbé voltak fontosak, és egyetlen külső stílusleíró nyelv sem lett széleskörűen elfogadva a CSS megjelenése előtt.

A CSS eredetileg Håkon Wium Lie ötlete volt 1994-ben. Bert Bos időközben egy Argo nevű böngészőn dolgozott, mely saját stíluslapokat használt; végül ők ketten döntöttek a CSS kifejlesztése mellett.

Ekkor már több stílusleíró nyelv is létezett, de a CSS volt az első, ami a kapcsolás ötletét használta fel, vagyis a dokumentum stílusa több különböző stíluslapból tevődhetett össze. Ezáltal lehetőség nyílt arra, hogy a felhasználó által megadott stílus bizonyos esetekben felülírja a szerző stílusát, míg a többi esetben örökli azt. A stíluslapok ilyen kapcsolása mind a szerző, mind a felhasználó számára rugalmas vezérlést biztosított, mivel megengedte a vegyes stilisztikai beállításokat.

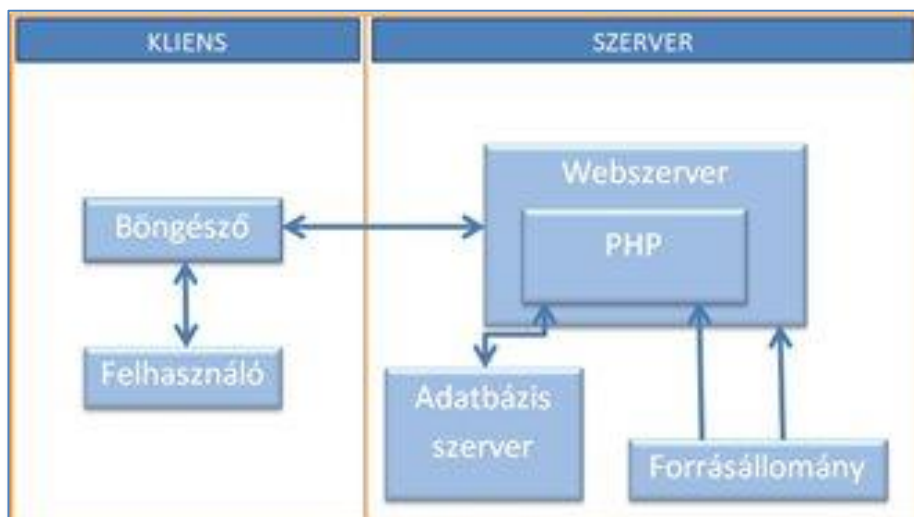
Håkon ajánlata a "Mosaic és a Web" konferencián került bemutatásra Chicagóban, először 1994-ben, majd később 1995-ben. Ebben az időben alakult meg a Word Wide Web Consortium is, mely később fellépett a CSS érdekében, és megalapított egy bizottságot a részletes kidolgozására. Håkon és Bert volt az elsődleges technikai vezetője a projektnek, melyhez további tagok csatlakoztak, többek között Thomas Reardon a Microsoft-tól. 1996 decemberében a CSS level 1 ajánlata hivatalosan is megjelent.

1997 elején a CSS egy Chris Lilley vezette csoporthoz került a W3C-nél. A csoport azokkal a problémákkal foglalkozott, melyeket a CSS Level 1 kihagyott. A CSS Level 2, mint hivatalos ajánlat, 1998 májusában jelent meg. A CSS Level 3 még jelenleg, 2014-ben is fejlesztés alatt áll, bár bizonyos funkcióit már használják a legújabb böngészők. (wikipédia, 2016)

3.4. PHP

PHP (Hypertext Preprocessor)

A PHP egy szerver-oldali szkript nyelv. A válaszlap formátumát leíró utasítások egy HTML dokumentumba ágyazva helyezkednek el. A Web-szerver a meghívott dokumentum URL-jéből ismeri fel, hogy a kérést a PHP szervernek kell feldolgoznia. Az adott dokumentum HTML utasításai változtatás nélkül átkerülnek a válaszlapra, a PHP specifikus elemeket a PHP végrehajtja, és a generált válasz kerül be az eredménylapra. Ugyan a PHP leíró nyelve nem adatbázis orientált, hanem a C általános programozási nyelvre épül, mégis tartalmaz adatbázis kapcsolódási kiegészítő modult, amely lehetővé teszi, hogy elérhessük, és a válaszlapon megjeleníthessük a legkülönbözőbb adatbázisokban tárolt adatokat is. (Balogh Csaba, 2014)



2. ábra A PHP sematikus működése

[http://www.tankonyvtar.hu/en/tartalom/tamop412A/2011-](http://www.tankonyvtar.hu/en/tartalom/tamop412A/2011-0003_06_szerver_oldal_programozas/1_2_php_nyelv_altalanos_jellemzoi_IWvRuyJk38w0T7De.html)

[0003_06_szerver_oldal_programozas/1_2_php_nyelv_altalanos_jellemzoi_IWvRuyJk38w0T7De.html](http://www.tankonyvtar.hu/en/tartalom/tamop412A/2011-0003_06_szerver_oldal_programozas/1_2_php_nyelv_altalanos_jellemzoi_IWvRuyJk38w0T7De.html)

A honlapom szerkezetében a fejléc, a menürész statikus, nem változik. A tartalom részben viszont változik az aktuális linknek megfelelő tartalom:

Index.php-ban szereplő tartalom meghívó programrészlet:

```
<div class="tartalom">
  <?php
    require("tartalom.php");
  ?>
</div>
```

tartalom.php eset szétválasztással (case szerkezettel) oldottam meg a változó oldalt.

```
<?php
  if(isset($_GET["center"]))
  {
    switch ($_GET["center"]){
      case "nyito": require_once("center/nyito.php"); break;
      case "index1": require_once("center/index1.php"); break;
      case "index2": require_once("center/index2.php"); break;
```

```

case "index3": require_once("center/index3.php"); break;
case "index4": require_once("center/index4.php"); break;
case "hirek":  require_once("center/hirek.php"); break;
case "terkep": require_once("center/terkep.php"); break;
case "index5": require_once("center/index5.php"); break;
}
}
else{
    if(isset($_GET["lap"]))
    {
        require_once("center/nyito.php");
    }
    else{
        require_once("center/nyito.php");
    }
}
?>

```

3.5. Java, Java applet, JavaScript

A programozási nyelvek fejlődését figyelemmel kísérő programozók fülében ismerősen cseng a Java szó. A Java napjaink talán legígéretesebb programozási nyelve, melyet a Sun Microsystems és a Netscape Communications Corporation együtt fejlesztettek ki. A Java nyelv nem más, mint egy hatalmas, tisztán objektum-orientált programozási nyelv, melynek nagy ereje, hogy segítségével platform független alkalmazások készíthetők. A Java nyelven megírt programokat egy ún. köztes kódra (byte kód) kell lefordítani, melyek különböző platformokon, különböző operációs rendszerek alatt futtathatóak - egy értelmező program segítségével.

A Java szóval napjainkban - a WWW világában - három területen találkozunk.

Egyrészt beszélünk **Java programokról**, mely programokat le kell fordítani és önállóan futnak. (Nem támaszkodnak más programokra.)

A **Java appletek** (programocskák) - kis Java programok -, melyek a Java-t támogató böngészőkből futtathatóak. Ezek tehát felhasználják a böngészők által nyújtott

szolgáltatásokat. Az appleteket is le kell fordítani és - ahogyan azt már korábban bemutattam - ezek HTML dokumentumokból hívhatók.

A **JavaScript** a Java nyelv "kis testvérkéje" - mondják sokan. A JavaScript-ek a HTML lap szövegébe épülnek be, csak a böngészőben képesek futni.

A JavaScript, mint neve is mutatja, egy script nyelv, amit legelőször a Netscape Navigator 2.0 támogatott. A JavaScript-eket HTML lapokba lehet beágyazni, a lappal együtt töltődnek le, majd a böngésző értelmezi és futtatja azokat. Hasonlít a Java-ra, de ugyanakkor sokkal kisebb, egyszerűbb, korlátozottabb működésű is annál.

(nejanet, 2012)

Példa:

```
LightboxOptions = (function() {  
  function LightboxOptions() {  
    this.fileLoadingImage = 'images/loading.gif';  
    this.fileCloseImage = 'images/close.png';  
    this.resizeDuration = 700;  
    this.fadeDuration = 500;  
    this.labelImage = "Ke'p";  
    this.labelOf = "/";  
  }  
}
```

A galériához egy szabadon felhasználható javascript-es galériamintát használtam, amit az igényekhez mérten módosítottam.

4. A fiktív alapítvány igényei

A honlapot egy nonprofit természetvédelemmel foglalkozó alapítvány számára kell elkészíteni. Az alapítvány elsősorban könnyen áttekinthető, tetszetős honlapot szeretne, amely bizalmat ébreszt a leendő partnerekben. A dizájnr a letisztultság és átláthatóság legyen jellemző. Ezen kívül legyen a honlap informatív, és mutassa be a szervezet eddigi nagyobb referenciáit, munkáit. Legyenek benne minőségi fotók, amelyek tükrözik az alapítvány igényességét a munkáik tekintetében. Az oldalak töltődjenek be gyorsan, amely szintén közvetetten reprezentálja a szervezet minőségi tevékenységén kívül a gyorsaságot, és hatékonyságot. Nem cél egy dinamikus, állandóan frissülő honlap, hanem fontosabb, hogy a leendő partner, vagy akár csak egy egyszerű érdeklődő tájékozódni tudjon az alapítvány tevékenységeiről, és eddigi munkáiról. Valamint az is fontos természetesen, hogy az érdeklődők fel tudják venni a kapcsolatot az alapítvánnyal, megtalálják térképen.

Ahhoz, hogy a honlap megfelelő helyen helyezkedjen el az internetes keresők – legfőképpen a Google – találati listáján. Az Internetes jelenlét korunk alapvető elvárásává vált, és az is fontos, hogy az érdeklődők rátaláljanak a szervezetre. Manapság a legtöbb ember az Internetet használja tájékozódásra. A szakdolgozatom 10. fejezetében részletesebben is ismertetem a keresőoptimalizáláshoz tartozó módszereket.

A honlap a fenti igények figyelembevételével született, és a következő fejezetben ismertetem a szóba jöhető lehetőségeket egy, a mai felhasználói igényeket kielégítő honlap elkészítéséhez, továbbá ezek előnyeit és hátrányait, valamint megindokolom a döntésemet, hogy miért választottam a honlap elkészítésének a jelenlegi módját.

Összefoglalva tehát az elvárások:

- Egyediség
- Tetszetős, igényes külalak
- Gyors betöltődésű oldalak
- Alacsony költségek
- Informatív, de leginkább reprezentációs cél
- A lehetőségek szerinti legnagyobb rezponzivitás

5. Weboldal készítés módszerei, lehetőségek

Egy weboldalt, vagy inkább egy website-ot többféle módszerrel is létre lehet hozni. Vannak direkt erre a célra készített weboldal szerkesztő programok. Vannak az Interneten elérhető weboldalkészítő webhelyek, ahol sablonok alapján létre tudunk hozni honlapokat. Egy viszonylag új technológia a két előbb felsorolt mellett a tartalomszerkesztők, vagy más néven CMS-ek (Content Management System) használata, ahol úgynevezett portál szerű honlapot tudunk készíteni.

5.1. Forráskód szerkesztők

Ezen programok viszonylag egyszerű alkalmazások, melyekkel a HTML ismeretekkel rendelkezők számára forráskód kiegészítő, és forráskód kiemelő funkciók segítik a weblap létrehozását. Elengedhetetlen ezeknél a HTML, CSS haladó szintű ismerete. Persze ez attól is függ, milyen minőségű honlapot kívánunk létrehozni. Ezek fő előnye, hogy nagyon tiszta, felesleges kódok nélküli forráskódot tudunk produkálni. Ez azt az előnyt hordozza magában, hogy a létrehozott kód gyorsabban fog betöltődni, és a forráskód is kevésbé lesz bonyolult.

Notepad ++

Az egyik ilyen ismert ingyenes forráskód szerkesztő program a Notepad ++ nevű alkalmazás. Tulajdonképpen egy hasznos funkciókkal kiegészített jegyzettömb program. Erre utal a nevében is a Notepad. Windows platformra készült, de Akár Linux vagy Mac OS X alatt is futtatható, a Wine segítségével. A Notepad++ egyik fő előnye a Windowsbeépített szerkesztőjével szemben a fülekkel ellátott felület, amely több fájl párhuzamos szerkesztését engedi meg. A Notepad++ támogatja a makrókat és plugineket. Jelenleg 27 hivatalos plugin létezik, ebből 10 -et tartalmaz alapkiépítésben. Szabad szoftverként terjesztik.

Néhány alapvető fontosságú funkció a teljesség igénye nélkül:

- Füles interfész
- Fogd és vidd
- Többszörös vágólap
- Osztott képernyős szerkesztés és szinkronizált görgetés
- Keresés és csere több dokumentumon keresztül

- Fájlok összehasonlítása
- Nagyítás/kicsinyítés

Fejlett képességek a kódszerkesztés segítésére:

- Automatikus kiegészítés
- Könyvjelzők
- Szintaxis kiemelés
- Blokk és bekezdés kiemelés
- Beszédfelismerés
- FTP böngésző (a szükséges plugint az alapterepítés tartalmazza)
- Makró rögzítés és végrehajtás
- Különféle eszközök, mint sorrendezés, szövegekódolás konverzió, szöveg becsomagolás

A Notepad++ a fent felsorolt funkciókat 48 programozási, szkriptelési és jelölőnyelvhez támogatja. Automatikusan megpróbálja felismerni a megnyitott fájlban használt nyelvet, amihez egy testre szabható kiterjesztéslistát használ. A felhasználó manuálisan is átállíthatja az éppen használt nyelvet. (Wikipédia, 2016)

PsPad

A PSPad egy könnyen használható ingyenes weblapszerkesztő editor, mely az adott programnyelv szintaxisát különböző színezéssel jeleníti meg, ezzel segítve a programozót a szintaktikai hibák elkerülésében. A beépített FTP kliensnek köszönhetően weblapjainkat közvetlenül a webszerveren is szerkeszthetjük a PsPad használatával. Az elkészült weblapot Internet Explorer és Firefox alatt is megtekinthetjük. Az editor beépülő elemek segítségével teljesen a weblapszerkesztő igényei szerint egyénre szabható. (KISS HONLAPKÉSZÍTÉS, 2015)

EditPlus

Windows platformra készült szövegszerkesztő. Nem ingyenes szoftver. Természetesen alapvető funkciók közé tartozik a forráskód kiemelés, körülbelül azokat tudja, mint a két előző program.

További kevésbé ismert szerkesztők a teljesség igénye nélkül: UltraEdit, KomodoEdit, Eclipse, TextMate stb

5.2. WYSIWYG szerkesztők

Ezeknél a programoknál is szükség van megfelelő jártasságra a program használatában, hogy megfelelő eredményt tudjunk elérni. A következő programok nem csak egy weboldal elkészítésében nyújtanak hatékony segítséget, hanem az egész web helyet elkészíthetjük a segítségükkel. Támogatják az oldalak linkelését húzásos gumivonalas módszerrel, és a létrehozott dokumentum kinézetének ellenőrzését is különböző böngészőprogramokban.

Adobe Dreamweaver CC

A szoftver a fizetős programok közé tartozik. Az oldalszerkesztéseket valós időben lehet ellenőrizni a böngésző frissítése nélkül, és a webhelyek előnézetét több eszközön is (táblagép, mobiltelefon) megtekinthetjük egy időben. Adobe Stock közvetlenül be van építve a programba. Az Adobe Stock az Adobe Creative Cloud része, amely egy alkalmazáson belüli szolgáltatás, ami képek, grafikák, videók millióit tartalmazza, amelyek kereshetők, és könnyen beépíthetők a weboldalba. A Creative Cloud az Adobe cég grafikai, illetve videó, és egyéb kreatív munkák készítését segítő programok gyűjtőneve.

Front Page

A program a Microsoft Office programcsomag része volt. Kereskedelmi szoftver. Fejlesztésével 2006-ban leálltak. A programot a Sharepoint Designer, és a Microsoft Expression Web alkalmazások váltották fel.

Microsoft Expression Web

A program freeware. Az alkalmazással web oldalakat lehet tervezni és fejleszteni. Használja a HTML 5-öt, CSS 3-at, ASP.NET-et, PHP-t, JavaScript-et, XML+XSLT-t, XHTML-t. Az utolsó kiadása 2012-ben volt, és ezt a programot sem fejlesztik már tovább. Támogatása 2020-ig fog tartani. .NET Framework 4.0 és Silverlight szükséges a futtatásához, illetve a telepítéséhez.

Sharepoint Designer

Az alkalmazás a Microsoft Expression Web testvérprogramja. Főként a tervezésre és a dizájnrá helyezi a hangsúlyt a Microsoft SharePoint oldalak tekintetében. Legutolsó verziója a 2013-as kiadású. Freeware szoftver.

KompoZer

A program ingyenes, és elérhető magyar nyelven is. A többi weboldalszerkesztőhöz hasonlóan van benne FTP kliens is, amivel fel lehet tölteni a tárhelyre az elkészült fájlokat. Erős támogatása van formok, táblázatok és sablonok tekintetében. A program elérhető Windows, Linux és Macintosh rendszerekhez is. (Compozer, 2017)

5.3. Webes sablonok

Vannak olyan weboldalszerkesztő webhelyek, amelyek többféle sablont tartalmaznak, és ezek segítségével viszonylag gyorsan el tudunk készíteni programozási tudás és weblapkészítő alkalmazásban való jártasság nélkül is tetszetős webhelyeket, weboldalakat. A web sablonok általában viszonylag modern, tetszetős dizájnnal, és trendi grafikákkal rendelkeznek. Úgy alakították ki ezeket, hogy általános informatikai tudással is létre lehessen hozni webhelyeket. Hátrányuk, hogy nem annyira testre szabhatóak, mint a weblapszerkesztők vagy a tartalomkezelők által készített oldalak. Előfordulhat, hogy olyan dizájnt választunk, amit még szintén többen is használnak. Ez esetleg ronthatja a cég renoméját a partnerek előtt. Persze ez üzlet, illetve vállalkozástípustól is függ. A sablonok nem nyílt forráskódúak, így átalakításuk, megváltoztatásuk általában csak a webhely üzemeltetői által lehetséges.

A legtöbb webes sablon használatánál ingyen igénybe vehetjük az alapszolgáltatásokat, de ha többet szeretnénk – pl. saját domain, több e-mail cím, beépülő modulok, még több sablon, alkalmazások – azért már fizetnünk kell.

Wix.com

A Wix egy ingyenes online weboldal építő rendszer, amivel fogd, és vidd módszerrel lehet weboldalakat létrehozni.

Néhány jellemzője a WIX-nek:

- 500+ sablon

- Fogd és vidd módszer
- 223 applikáció
- Reszponzív weboldal
- Webáruház készítés
- Jó felépítés SEO (kereső optimalizálás) szempontból
- 24/7 ügyfélszolgálat

A WIX hátránya, hogy magyar nyelvű támogatása gyakorlatilag nincs. A Wix nem nyitott kódolású, csak a készítőik tudják alakítani, fejleszteni. Ez azt jelenti, hogy csak a Wix csapata adhat hozzá új eszközöket, így ezek az eszközök sokkal kisebb valószínűséggel lesznek hibásak, bugosak. (Nagy Attila, 2017)

Weebly

A weboldalak összerakását illetően modulokkal dolgozhatunk, kényünk-kedvünk szerint válogathatjuk össze a rendelkezésünkre álló mozaikokból a végleges honlapot. Át kell esnünk egy ingyenes regisztráción, amelyért cserébe 500 MB tárhelyet is kapunk, amelyen felépíthetjük oldalunkat, s a végeredmény egy sajátnev.weebly.com URL című honlap lesz. A szerkesztőfelület itt elérhető böngészőből, így a gépünkre nem kell telepítenünk semmit. De a Google Play áruházból vagy az App Store-ból letölthetünk androidos, illetve iPhone-os appot is, amelyben összerakhatjuk Weebly-s weblapunkat. Ha szeretnénk, bővíthetjük lehetőségeinket előfizetéssel, melyért cserébe saját domain, keresőoptimalizálást, Weebly-logó mentesítést kapunk. (Gorgo, 2016)

Webnode

A Webnode online weblapszerkesztő a weboldalak 3 különböző típusát kínálja: személyes weboldal, üzleti weboldal és webáruház.

A rendszer futtatható a legtöbb böngészőn (például Internet Explorer, Mozilla Firefox, Netscape, Google Chrome, Safari és Opera). A weboldalak létrehozása és szerkesztése akár internetkapcsolattal rendelkező okostelefonokról is lehetséges. A rendszer a Freemium üzleti modell jegyében működik, ami azt jelenti, hogy az alapszolgáltatások az ingyenes verzióban is elérhetőek, de a felhasználónak bármikor lehetősége van Prémium csomag vásárlására, amely nagyobb tárhelyet, magasabb adatforgalmi korlátot és további haladó funkciókat

kínál. Lehetőség van a saját domain regisztrációjára is, a cég továbbá személyre szabott marketingszolgáltatásokat is kínál. (webnode.hu, 2017)

Nagy előnyei közé tartozik, hogy magyar nyelvű változat is van, magyar nyelvű technikai támogatást is nyújtanak.

További ismertebb online sablonnal is rendelkező weboldalszerkesztők még többek között a GoDaddy és a Jimbdo.

5.4. CMS rendszerek

A tartalomkezelők bizonyos fokú jártasságot igényelnek az adott rendszerben. Kell egy bizonyos idő, amíg megtanulja ezeket használni a felhasználó. Cserébe jóval nagyobb szabadságot adnak, mint a webes sablonok. Előnyük még, hogy különböző jogosultságú felhasználók használhatják a létrehozott webhelyet. Lehetnek tervezők, dizájnerek, vagy csak egyszerű hírszerkesztők, vagy beküldők. Szükség van egy háttér adatbázisra is, amely a legtöbbször MySQL. Ezen kívül a szerveren, amin elhelyezzük, szükség van a PHP nyelvre, és természetesen egy web szerver programra, ami a legtöbb esetben az Apache. Sablonok használatára is lehetőség van, és ezeket kellő szakértelem ismeretében meg is változtathatjuk, ha akarjuk. Így teljességgel egyénre szabott honlapokat tudunk létrehozni.

Mivel sokféle igényt kell kielégíteniük, a tartalomkezelő rendszerek rengeteg fájl, kódot, képet stb. tartalmaznak. Ez a nagymennyiségű adat mindenképpen tárolva kell, hogy legyen a szerveren és a sokszor fölösleges kódok lassítják az oldal betöltődését. A tartalomkezelőkkel készített weboldalnak általában nehezebb, bonyolultabb, tehát időigényesebb a keresőoptimalizálása, mint egy hagyományos weboldalnak.

Mivel a tartalomkezelő rendszerek forráskódja könnyedén hozzáférhető bárki számára, így az internetes bűnözők is kedvükre tanulmányozhatják őket. Ez pedig biztonsági problémákat vet fel, mert könnyen ki tudják fürkészni gyenge pontjaikat, és támadást indítani ellenük. Sokszor észrevehető az ilyen weboldalak esetében, hogy bizonyos funkciók nem működnek, ha valamelyik, kevésbé elterjedt böngészővel próbáljuk használni őket. Egy-egy komolyabb böngészőprogram verzióváltása után (például amikor megjelent az Internet Explorer 8) pedig a meglévő weboldalakat sok esetben hozzá kell igazítani az új verzióhoz, hogy a már azt használók számára is tökéletesen jelenjen meg. Míg ez a kiigazítás egy egyedi készítésű weboldal esetében - a készítőjével kötött

megállapodástól függően - napokon vagy akár órákon belül megtörténik, addig például egy Joomla weboldalnál a weblapkészítő tehetetlen mindaddig, amíg meg nem jelenik az a CMS frissítés, ami már kompatibilis az új böngészővel. (awasweb.hu, 2017)

Számos tartalomkezelő rendszer érhető el az interneten. Ezek közül a legtöbb felhasználó négy CMS-t részesít előnyben, amiknek a népszerűsége messze nagyobb, mint az összes többinek. (DIMA, 2013)

WordPress

Az utóbbi évek nagy nyertese, viszonylag rövid idő alatt hatalmas népszerűségre tett szert. Blog motorként indult, de mára már szinte bármilyen típusú honlapot (portált, fórumot, webáruházat) létre lehet vele hozni. Az adminisztrációs felülete nagyon felhasználóbarát, ebben a tekintetben lekörözi a konkurenciáit. Tud magyarul is. (DIMA, 2013)

A WordPress CMS használatához igénybe vehetünk egy tárhelyszolgáltató céget, vagy akár saját kiszolgáltót is üzemeltethetünk. A legtöbb hazai tárhelyszolgáltatónál egyszerűen csak annyit kell tennünk, hogy letöltjük a WordPress magyar nyelvű változatának legfrissebb kiadását a hu.wordpress.org oldalról, majd feltöltjük a szolgáltatónk által biztosított mappába. Vannak olyan szolgáltatók is, amelyek rögtön kész WordPress oldalakat kínálnak. Ilyen például a WordPress fejlesztéséért is felelős Automattic által működtetett wordpress.com, amely egy alap WordPress oldalt kínál ingyenesen, az extra szolgáltatások igénybe vételéért azonban fizetnünk kell. (Wphu.org, 2016)

CMS Made Simple

Átlátható adminisztrátori felülettel rendelkezik. Minimális erőforrást igényel, beépített és online súgó érhető el hozzá, egyszerű felhasználó-kezelés, számos előretelepített modul tartozik hozzá (keresés, szavazás, képgaléria, blog stb.).

A CMSMS elkülöníti egymástól a tartalmat a formától, vagy ha úgy jobban tetszik, a struktúrát a design-tól. Ez lehetővé teszi a designernek, hogy a design-ra koncentráljon, a Szerkesztőnek pedig, hogy tartalommal töltse fel az oldalakat anélkül, hogy egymás tevékenységét zavarnák. Miután az oldal formája végleges lett, a tartalmak hozzáadása azt már nem érinti. Az Adminisztrátor menedzseli a felhasználókat pl. ki milyen jogosultságokat kapjon, és milyen tevékenységet folytathasson. Csak az Adminisztrátor tud további

kiterjesztéseket telepíteni és további funkciókat hozzáadni az oldalhoz. Ez a három szerep - Designer, Szerkesztő, Adminisztrátor - egymástól elkülönül, akár személyében is. (CMS Made Simple.hu, 2011)

Drupal

Stabil és biztonságos rendszer, amit kormányzati oldalak is használnak (pl. a Fehér Ház). Nagymértékben testre szabható, elérhető hozzá számos kiegészítő és sablon is. Támogatja a felhasználóbarát URL címeket és a többnyelvűséget is. (DIMA, 2013)

Joomla

A legnépszerűbb, ezért ehhez érhető el a legtöbb kiegészítő és sablon is. Érdemes már a 3-as verzióját használni, mert számos javítást tartalmaz a 2-es változathoz képest. Hátránya, hogy elég nehézkes az adminisztrátor felületének használata, sok tapasztalatot igényel. Elérhető hozzá magyar nyelv is. (DIMA, 2013)

6. A személyre szabott weboldalak kialakításával kapcsolatos döntések

A fiktív alapítvány igényeit figyelembe véve, amely számára a webhely készült, a szóba jöhető lehetőségek közül a forráskód szerkesztő programmal létrehozott honlap mellett döntöttem. Az érveim, amely ezt a döntést segítették elő a következők:

- Olyan weboldalakra volt szükség, amelyek gyorsan betöltődnek a böngészőkbe, így olyan kódot kellett alkalmazni, amely teljesen mentes a felesleges részekről. A WYSIWYG weboldal szerkesztők, a websablonok, és a CMS-ek rengeteg felesleges részt tartalmazhatnak, amire a cél szempontjából semmi szükség. Ez a feleslegesen bonyolult kód különböző mértékben ugyan, de mindenképp lassítja az oldalak betöltődését.
- A fiktív szervezet szeretett volna egyéni weboldalt, így a websablonok eleve kiestek a választási lehetőségek közül, mivel az ezekkel létrehozott webhelyek eléggé hasonlítanak egymásra, úgymond egy kaptafára készültek. A CMS-ekkel létrehozott weboldalaknál is lehetőség van úgynevezett template-ek, vagy más néven sablonok használatára, és ezek jobban személyre is szabhatók, azonban ezek átalakításához nagyobb jártasság szükséges a tartalomkezelő rendszerek használatában, így ezen rendszerek alkalmazását is elvetettem. Ha felmerülnek a jelenlegitől különböző igények, viszont elképzelhető, hogy a honlap továbbfejlesztési lehetőségei között szerepelhet egy olyan átalakítás, amely valamelyik CMS segítségével valósítható meg a leghatékosabban.
- A forráskód szerkesztő alkalmazása mellett szólt még az az érv is, hogy a gazdasági informatikus képzésen jártasságot szereztem a HTML, CSS, JavaScript nyelvek használatában, így ennek a lehetőségnek az előnyét tudtam hasznosítani a webhely megalkotásában. Mivel előzetes informatikai ismereteim, és a weboldalszerkesztésben, és grafikai munkákban szerzett gyakorlataim ezt lehetővé tették, ez is abba az irányba vitt, hogy a megszerzett ismereteket ezen programok valamelyikének a segítségével kamatoztassam a weboldalak megalkotásánál.
- Mivel az alapítvány nonprofit jellegű, ezért a létrehozási költségek egyáltalán nem voltak mellékesek. Ez is leszűkítette a szóba jöhető lehetőségek számát, például fizetős termékek, és drága webhelyek használatát a projekthez.

Végül a fiktív alapítvány igényeit figyelembe véve, és az optimális lehetőségeket számba véve a Notepad++ forráskód szerkesztő használata mellett döntöttem. A weboldal létrehozásánál alkalmaztam a HTML, CSS, valamint JavaScript adta lehetőségeket. A képgaléria létrehozásában egy Internetről szabadon letölthető sablont is felhasználtam, amit az igényekhez, és megvalósítandó célokhoz igazítva átalakítottam.

7. Hibakeresés, tesztelés

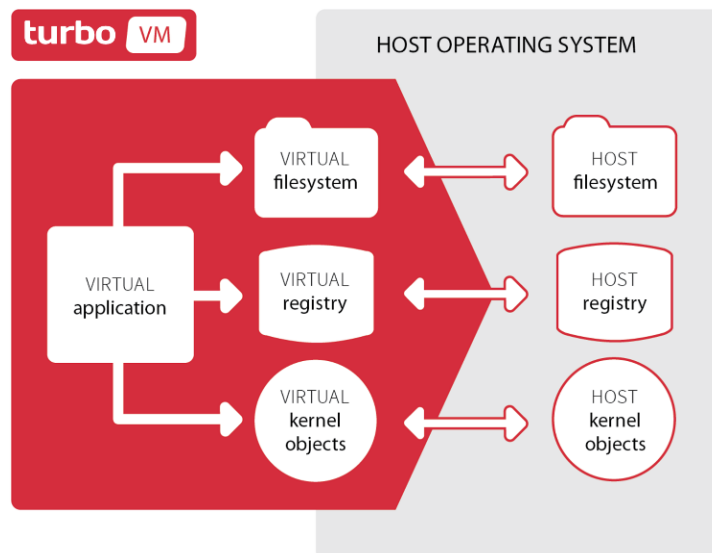
A weboldalak tesztelésének alapvető feladata, hogy megvizsgáljuk, hogyan mutatnak, hogyan viselkednek a weblapjaink különböző böngésző programokban. A böngészők közötti különbségek azt eredményezhetik, hogy ugyanaz a weboldal kicsit másként néz ki egy másik böngészőben. Ha a különbség nem annyira zavaró, vagy csak nagyon kicsi, akkor elfogadható a weblap, viszont, ha hibásan jelenik meg egy-egy grafikai elem, vagy akár szöveges rész, akkor sajnos alakítani kell még az oldalt, hogy az adott böngészőben is elfogadható legyen a megjelenés. Az is előfordulhat, hogy néhány funkció, vagy dizájn, illetve animációs elem egyáltalán nem működik az adott böngészőben - mert például egyáltalán nem támogatja a Jávát. Ilyenkor megoldásokat kell találni a problémára.

Virtuális gépek

A különböző böngészőkben végzett tesztelés során nagy segítség lehet a virtuális gépek használata, ahová különböző böngészőket telepíthetünk fel, hogy kipróbáljuk, hogy viselkednek a weboldalaink ezekben. Ezeket a gépeket fel kell konfigurálnunk megfelelően, és a programokat fel kell telepítenünk ezekre. Előfordulhat, hogy több operációs rendszert is fel kell telepítenünk, hogy az azonos nevű böngészőprogramok különböző verzióit is ki tudjuk próbálni az oldalaink viselkedése szempontjából. Ilyen virtuális gép szoftverek a VMware, VirtulaBox, KVM.

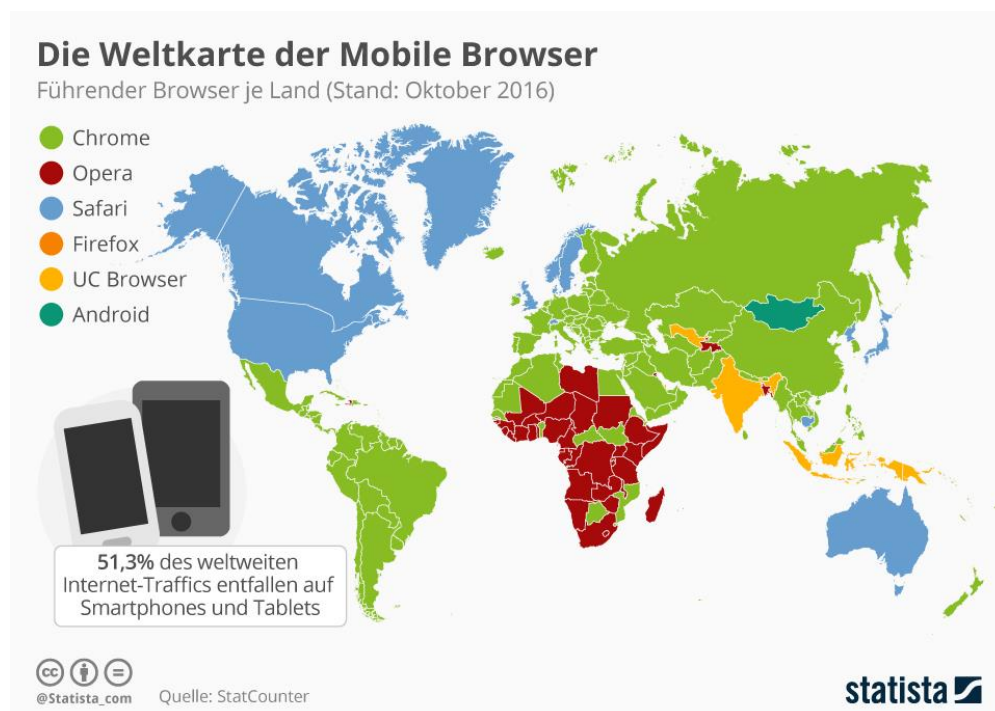
Turbo.net

A tesztelés szempontjából segítséget jelenthet a **turbo.net** weboldal, ahonnan különböző programokat futtathatunk virtuálisan anélkül, hogy a gépünkre telepítenénk ezeket. Ezek között a programok között különböző böngészőprogramok is vannak. A programokat lehet futtatni a felhőből is egy szerverről, és lehet a saját számítógépünkre letöltve. A weboldal csak egy regisztrációt kér tőlünk, és ezen kívül le kell töltenünk egy virtuális futtató környezetet (Spoon Virtual Machine Engine), amiben tulajdonképpen futnak a kipróbálásra szánt programok. A programokat a weboldalról indíthatjuk, és használhatjuk őket a saját számítógépünkön. Teszteléshez, programok kipróbálásához nagy segítséget nyújthat. Nagy előnye, hogy nem terheljük annyira gépünk erőforrásait (merevlemez, memória stb.), mintha virtuális gépeket telepítenénk rá.



3. ábra A Turbo Virtual Machine sematikus működése
<https://turbo.net/docs/getting-started/what-is-turbo#how-does-it-work>

A másik feladat, hogy megvizsgáljuk, leellenőrizzük, hogy hogyan néznek ki weblapjaink a különböző mobil eszközökön, mint pl. mobiltelefon vagy tablet. A reszponzivitás fontos, hiszen ilyen eszközökön egyre többen interneteznek, egyre többen használják internetezésre, tájékozódásra ezeket az eszközöket, hiszen ezekkel szinte bárhol, bármikor elérhető az Internet, nem kell gép előtt ülnünk, hogy információkat szerezhessünk valamiről.



4. ábra Mobil eszközökön lévő böngészőprogramok használata földrajzi eloszlásban
https://infographic.statista.com/normal/infografik_6812_die_weltkarte_der_mobile_browser_n.jpg

A Google soktényezős keresési algoritmusába 2015. április 21-től beépült a responsive, mobil barát jelleg is, mely 2016. májusától a mobilról indított kereséseknél még kiemeltebb szerepet kapott. Amennyiben szeretnénk a Google organikus találati listájában pozíciónkat megőrizni, érdemes erre felkészülni és a saját weboldalunkat a Google erre kialakított ellenőrző eszközével mielőbb ellenőrizni. Az ellenőrzéshez mindössze annyi a dolgunk, hogy a Google webmester eszközök "Mobilbarát teszt" oldalán beírjuk a weboldalunk címét és az ellenőrzésre kattintva megvizsgáljuk azt reszponzivitás szempontjából. (A link: <https://search.google.com/search-console/mobile-friendly?hl=hu>) (WebFolio Webdesign, 2016)

Egy weboldallal szükséges rendszeresen foglalkozni, több-kevesebb idő elteltével a megfelelő lépéseket és az időközben felmerült szükséges átalakításokat megtenni. Ezeket leggyakrabban a felhasználói visszajelzések tükrében eszközöljük.

8. Domain név regisztráció

A domain név (tartománynév) a számítógépek azonosítására szolgálnak. Az Internetre csatlakozó valamennyi számítógépnek egyedi azonosítója, ún. IP-címe (Internet Protocol) van (pl. 216.151.212.175). A domain nevek ezen nehezen megjegyezhető IP címeket teszik könnyen megjegyezhetővé, és egyben egyedivé.

A legfelsőbb szintű domain nevek – végződésük alapján – alapvetően két nagy csoportba oszthatók: nemzetközi fődomének és nemzeti domének. Például a **.com** - kereskedelmi tevékenységet, a **.net** – hálózatot, az **.org** - szervezetet, az **.eu** - uniós államot jelöl. A nemzeti domének végződései egy-egy ország nevéből képzett rövidítések, mint **.hu** - Magyarország.

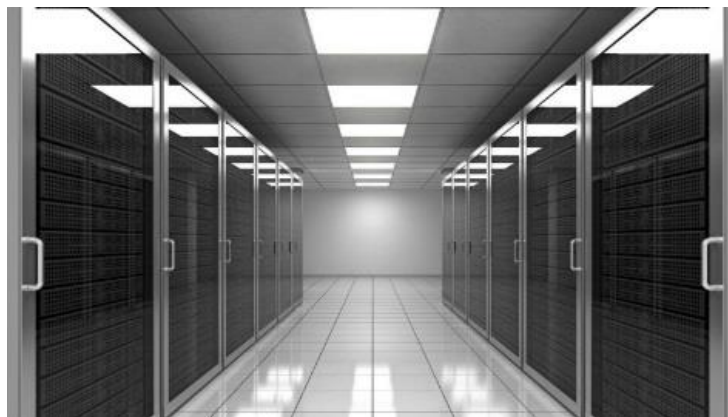
8.1. Lehetőségek, szempontok

Ami lényeges a domain név választásnál főként üzleti tevékenység esetében:

- legyen frappáns / egyedi
- könnyen megjegyezhető
- ne legyen túl hosszú
- kerüljük a kötőjeleket a domain névben
- foglaljuk le a domain ékezetes változatát is, ha van
- Magyarországon érdemes .hu végű domaint választani
- minden esetben ellenőrizzük a domain név jelenlegi foglaltságát pl itt: <http://www.domain.hu/domain/szabad-e/>
- Külföldi oldalak esetében itt: <https://www.onlydomains.com/>
- illetve azt, hogy a kiválasztott név nem esik-e valamilyen védelem alá (Webfolio, 2014)

9. Webtárhely kiválasztása

A megfelelő tárhely kiválasztása nagyon nagy jelentőséggel bír, úgymint a domain név megválasztása is. A webtárhely szolgáltatók általában nagykapacitású, nagy sebességű kapcsolattal rendelkező számítógépeken (szervereken) lévő tárhelyeket ajánlják fel. Ezek a gépek biztonságos gépteremben találhatóak, ahol az üzemeltetők gondoskodnak az állandó hőmérsékletről, páratartalomról, steril körülményekről, biztosítják az szünetmentes áramellátást és internetkapcsolatot. Úgynevezett szerverfarm kiépítése és fenntartása hatalmas költségekkel jár, ezért a piacon lévő megbízható tárhelyszolgáltató legtöbbjéből (hostingolás) ezeket a rendszereket (dedikált szerverbérlet).



9.1. Lehetőségek, szempontok

Sokszor a leendő weboldal tulajdonos nehezen becsüli meg, mekkora tárhelyre lesz szüksége egy adott tevékenységhez. A tapasztalat azt mutatja, hogy egy egyszerű, statikus weboldal 100MB tárhely területen is üzemképes. Ha azonban folyamatosan épülő tartalom, blog is szerepel a tervek között, akkor 250MB-os csomag méret körül érdemes kezdő csomagot keresnünk. (Webfolio, 2014)

Az **ingyenes** tárhely szolgáltatásokat legfeljebb tesztelés szintjén érdemes kipróbálni. Ezen szerverek üzemeltetői általában reklámmal próbálják meg fenntartani magukat. Honlapunk egy részén véletlenszerűen generált, sokszor az oldal témájához nem releváns tartalmú reklámok jelennek meg. Továbbá sokszor használnak – pl. atw.hu – régi elavult frame-s szerkezetet, amit a keresők már nem is indexelnek be. Továbbá a cégünk minőségi

internetes megjelenésére is rossz megítélést válthat ki: nem fizetőképessé benyomást kelthet, ha néhány ezer forintot is spórolunk.

A **fizetős** szolgáltatók kiválasztásánál pedig a következő szempontokat kell figyelembe vennünk:

- Szolgáltató cég nagysága, megbízható anyagi háttere.
- A szolgáltatás technikai követelményei, internetkapcsolat sebessége, háttérprogramok, amik nélkül a mai honlapok nem futnának. Pl.: MySQL, PHP... Mennyire biztosítja a technikai fejlődéssel járó frissítéseket.
- Mennyibe kerül az előfizetési csomag, és mit tartalmaz (email-cím, tárhely nagysága, napi mentés, webstatisztika, ügyfélkapu, vírusszűrés, egyéb szolgáltatások).
- Ügyfélszolgálat (support) minősége, megfelelő ügyfél tájékoztatás, ügyintézés minősége, gyorsasága, karbantartási értesítések küldése, adatbiztonság, műszaki problémák gyors megoldása.
- A tárhely szolgáltatók összehasonlításánál a tárhely rendelkezésre állást érdemes még összehasonlítani, minél inkább közelíti a statisztika vagy a szolgáltató vállalása a 100%-ot, annál ritkábban fog előfordulni, hogy nem elérhető a honlapunk.
- Fontos még, hogy a weboldalunkról a szolgáltató napi mentést is készítsen, melyből adatvesztéssel kapcsolatos probléma esetén a honlap helyreállítható.

9.2. Az optimális választás

Stabil megbízható szolgáltató kiválasztása nem könnyű feladat, mert a piaconsok cég hirdeti magát. A kiválasztásnál nem a legolcsóbb ajánlat a „nyerő”, hanem a biztonságot és megbízhatóságot kínáló szolgáltatóé. Nincs annál kellemetlenebb, ha vírustámadás, vagy rosszul működő szervergépek miatti leállás miatt nem elérhető a honlapunk, és annak szolgáltatásai. Az ilyen „szünetekből” adódó üzleti károk utólagos jogi rendezése a szolgáltatók felé pedig bonyodalmas.

Több internetszolgáltató kínálatát áttekintve, valamint egyéb informális csatornán keresztül (ismerős informatikus, más cégek internetes tapasztalatai, újságcikkek, statisztikai adatok, internetes fórumok, a szolgáltató honlapján közzétett referenciák) tájékozódva érdemes kiválasztani a szolgáltatót. A legmegfelelőbbnek a **Mediacenter** bizonyult.

Legfontosabb döntési tényezők:

- Tőkeerős cég, több éve a piacon van, megbízható infrastrukturális háttérrel rendelkezik.
- Folyamatosan emeli a szolgáltatások körét, az innovációkat követi.
- Ár – szolgáltatás aránya megfelelő.
- Brand Intel Quad Xeon (8 magos), minőségi szerverpark.
- 1 Gbit/sec sávszélesség szerverenként.
- Weblap elhelyezés tükrözött tárterületeken.
- 0-24h ügyeletes rendszergazdai felügyelet.
- Kiváló referenciákkal rendelkezik, pl. Joomla.hu, GSM, SAP.

10. Honlap ismertté tétele, keresőmarketing tevékenységek

Az alapítvány természetes igénye volt még, hogy az új honlap megfelelő helyen helyezkedjen el az internetes keresők találati listáján, ha valaki a természetvédelemre keres rá. Erre a legcélszerűbb módszer, ha egy direkt erre szakosodott céget kérünk fel, de sok mindent magunk is megtehetünk, hogy a honlapunk a keresők találati listáján előrébb kerüljön. Megfelelő anyagi források híján egyelőre ilyen keresőoptimalizálásra szakosodott cég bevonására nincs lehetőség, ezért az alábbiakban szeretném ismertetni az ezzel kapcsolatos teendőket, módszereket.

Az interneten jelenlevő vállalkozásoknak piaci sikereik eléréséhez ugyanúgy kell marketing tevékenységeket folytatni, mint az offline, valóvilágban. Jelen dolgozatomban a webmarketing tevékenységekből a keresőmarketing jelentőségét emelem ki.

10.1. Keresőmarketing

A keresőmarketing (Search Engine Marketing) többféle tevékenységséget takar, melyek céltudatos alkalmazásával webes jelenlétünk sikerének növelését érhetjük el. A siker alapjai: a látogatók számának növelése, ezáltal a cég tevékenységének és forgalmazott termékeinek széleskörű ismertté tétele, és a cégről, üzletről pozitív kép kialakítása

A keresőmarketing központi eleme a keresőoptimalizálás. Olyan technikák és eszközök alkalmazását jelenti, mellyel a keresők működésének ismeretében, azokhoz illesztve szerkesztjük meg, alakítjuk ki weboldalaink formáját, tartalmát. A keresőoptimalizálás lényege, hogy az általunk megcélzott, elérni kívánt felhasználók az interneten való keresésük eredményeként az első 10 - 30 találat között, vagy éppen az első helyen jelenítsék meg honlapunkat a keresőrendszerek. Azaz a lehető legtöbbször a legjobb helyeken és legegyszerűbben megtalálják honlapunkat.

Az optimalizálás eredménye fél - egy éves időtávlatban mutatkozik meg. Nem lehet kimutatni érdemi eredményt egy - két hónap alatt. A tartós eredményhez pedig folyamatos munkára van szükség. Egy weboldalt egyébként sem lehet magára hagyni, mert könnyen eltűnik az internet láthatatlan bitmélységeiben.

10.2. Honlap optimalizálása a keresőrobotok felé

Honlap készítésénél, kódolásánál alapvető feladat, hogy a honlap ergonómiája, forráskódja és tartalma megfeleljen felhasználók igényeinek, elvárásainak és illeszkedjen a keresőrendszerek működéséhez.

A keresőkhöz való adaptálás legelső foka a **World Wide Web Consortium** által meghatározott webes szabványoknak való megfelelés. A HTML kiértékelőkkel egyszerűen ellenőrizhetjük a dokumentumok helyességét a deklarált dokumentum-típusnak (DOCTYPE) megfelelően írtuk-e meg a forráskódot. A szabványok betartása azért alapvető, mert a robotok nem ismerik fel, vagy csak részben ismerik fel a nem szabványos oldalakat. A robotok számára vannak problémás technológiák, amiket nem tudnak értelmezni. Ilyenek a frames szerkezetű oldalak, és a Flash technológia alkalmazása.

HEAD

A keresőoptimalizálás fontos területe a HTML dokumentumon belül **HEAD** rész. Sok robot az itt található leírások, elemek alapján indexeli és értékeli az oldalt, vagy éppen mi magunk irányíthatjuk a robotok működését.

TITLE

A **<TITLE>** elemmel a dokumentumunk címét kell megadni. A keresők szempontjából nagyon fontos, hogy az oldal témáját lefedő szavakat használjunk, mert nagyon sok kereső ezt indexeli be kulcsszóként adatbázisába. Ezeket jelenítik meg - legtöbbször klikkelhető formában - a keresők találati listáiban. Mindenképpen figyelemfelkeltőnek, tájékoztató jellegűnek és az oldal témájához relevánsnak kell lennie. Az oldal címét a felhasználó a megjelenítő browser ablakában látja. Ezzel olyan többlet információt adunk a felhasználóknak, ami alapján el tudja dönteni, hogy szüksége van-e az adott dokumentumra, vagy sem. Minden oldalra más címszavakat kell alkalmazni, mert így tudja a látogató az egyes ablakok közötti váltásokat is nyomon követni. A honlap címe kerül be alapesetben a Kedvencek közé való felvételkor is, amikor a látogatók az oldalt bookmarkolják. Sokan elkövetik azt a hibát, hogy címsor nélküli dokumentumként "Untitled dokumentumnak" hagyják meg.

META

A **META** tagekkel közlünk információkat a feldolgozóprogramoknak és a robotoknak. De a keresőrobotokat lehet irányítani is a forráskódok fejrészében elhelyezett META tagokkal. A legtöbb kereső minden oldalt archivál cache -be, melyben minden oldalrégebbi változatát is meg lehet találni, és a hozzá tartozó link is minden találati eredmény megadásakor megjelenik. Ha csak a legfrissebb oldalakra van szükségünk, el lehet ezt kerülni:

`<META NAME="ROBOTS" CONTENT="NOARCHIVE">`

`<META NAME="GOOGLEBOT" CONTENT="NOARCHIVE">`

(Google robotját megszólítva)

Kérhetjük a robotokat, hogy indexeljék és kövessék a linkeket:

`<META NAME="robots" CONTENT="index, follow">`

Kulcsszavak megadására ad lehetőséget a következő META elem:

`<META NAME= "keywords" CONTENT=" ... ">`

Lehetőségünk van a dokumentumok tartalmának rövid leírására is. Szintén nagy jelentősége van, mert a találati listában megjelenítik egyes keresők az itt leírtakat. Az így megjelenő kivonatok figyelemfelkeltőnek kell lenniük, hogy felkeltsék a reménybeli látogatók/vásárlók érdeklődését.

`<META NAME="description" CONTENT=" ... ">`

A böngészők számára meg kell adni azt is, hogy milyen kódkészlet alapján értelmezze a megjelenítendő lapot. A magyar ékezetes betűket az ISO Latin-1 tartalmazza.

`<META http-equiv="Content-Type"CONTENT="text/html; charset=iso-8859-2">`

Ajánlatos megadnunk lapunk milyen nyelvű, ami szintén fontos információ kereső-robotoknak indexeli eljárásukkor.

`<META http-equiv ="Content-Language" CONTENT="hu">`

BODY

A **<BODY>** utáni rész a megjelenítendő oldalt írja le. Szövegek, szavak és utasítások halmaza. A keresők szöveges információt olvasnak, és szöveges linkeket követnek. Alapvető fontosságú ezért, hogy mennyi szöveg található az oldalon és mennyire hordozzák, közvetítik az információt mind a keresőmotorok felé, mind a látogatók felé. Nem szabad megtéveszteni a keresőmotorokat azzal, hogy más tartalmat mutatunk feléjük, mint amit az olvasóknak mutatunk. A csalások, megtévesztések sajnos az online világban is elterjedtek. Ennek a jelenségnek a neve, **cloaking**, azaz köpönyegforgatás. Érdekes a tartalomban minél többféleképpen leírni egy-egy szót, ragozva, összetett kifejezésben, vagy minél több szinonimát alkalmazni, hiszen a magyar nyelv gazdag szókészletű, és ezt ki kell használni.

Kulcsszavak

A tartalom megfogalmazásában, kialakításában fontos szerepet kapnak a **kulcsszavak**, olyan szavak vagy kifejezések, amelyek biztosítják a releváns találatokat a felhasználók kereséseire. Törekedni kell arra, hogy kulcsszavakat használjunk képek, mappák, hivatkozások, címsorok leírásában, elnevezéseiben is.

10.3. A domain név kiválasztásának jelentősége

A keresőoptimalizálás szempontjai szerint megírt honlapunknak lehetőség szerint olyan **domain nevet** kell adnunk, ami szintén tükrözi a tartalmat, a cég tevékenységét. A keresőrendszereknél nagy súlyt kapnak a jól megválasztott nevek. Képesek az ékezet nélküli nevekben is megtalálni a keresett szót, és a szótövezésre is, azaz ha a keresett kulcsszó csak a domain név töredéke, akkor is figyelembe vételre kerül. A legtöbb keresőben a domain névben megtalált kulcsszóval első helyezést lehet elérni a találati listákban.

10.4. Külső hivatkozások

A keresőmarketing utolsó lépéseként **külső hivatkozásokat** kell kiépítenünk, azaz minél több oldalra kell felíratni weboldalunkra mutató hivatkozásokat. Minél több oldal mutat weboldalunkra, annál jobb helyezést érhetünk el a keresőkben. Ez az állítás különösen igaz a Google -ra, amely az optimalizált honlap értékének mérésére bevezette **PR= PageRank** (oldalérték) algoritmusát. Olyan rekurzív számítási közelítést takar, mely az

egyes honlapokra mutató linkek száma alapján súlyozza, rangsorolja az oldalakat. De figyelembe veszi azt is, hogy a ráhivatkozó link milyen PR értékű weboldalon szerepel. Ebből a szempontból nagyobb súlya van a mi PR értékünkre, ha kevés magas minősítésű oldal hivatkozik ránk, mint a ha sok jelentéktelen oldalról mutat link az oldalainkra. Az alapelve a módszernek tehát, hogy egy adott dokumentum rangját azok a dokumentumok adják, melyek reá hivatkoznak. Az értéke 1-10 -ig terjed. A magas oldalértékkel rendelkező honlapok jobb helyezést érnek el a találati eredményekben. A legnagyobb cégek honlapjainak van csak 10-es PR értékük, mint Google, Apple. Az induló webhelyek, míg nem kerülnek be a keresők adatbázisaiba, s míg nem mutat rájuk külső hivatkozás addig a PR értékük= 0.

11. A webhely jövőbeni továbbfejlesztési lehetőségei

A továbbfejlesztési lehetőségek annak függvényében változhatnak, hogy az alapítványnak hogyan növekednek az igényei az internetes jelenléttel kapcsolatban. Egyelőre egy reprezentációs honlap megalkotása volt a cél, és ezt figyelembe véve választottam ki az eszközöket.

Amennyiben a szervezet közösséget is szeretne építeni, akkor további szolgáltatások bevezetése is szükségessé válhat. Ilyen lehet például, ha hírleveleket szeretne küldeni a honlapon regisztrált tagoknak az alapítvány. Az is elképzelhető, hogy időközönként, vagy egyre gyakrabban szeretne híreket közölni a honlapon. Esetleg fórumtémákat is indítana, közvélemény kutatást is tarthatna különböző témákban.

A különböző fajok, főként a védelem alatt álló állatok élőhelyeire webkamerákat is lehetne telepíteni, amennyiben a környező infrastruktúra ezt lehetővé teszi. Például nagy népszerűségnek örvendett annak idején az a webkamera, amelyen keresztül egy fehér gólya család életébe lehattünk be minden nap folyamatosan.

Elképzelhető, hogy lelkes természetvédők blogot is indítanának és vezetnék a honlap segítségével.

Amennyiben az alapítvány érdeklődik az felől, hogy az oldalát mennyien látogatták eddig, egy számlálót is el lehetne helyezni az oldalon.

Amennyiben ezek az igények idővel felmerülnének, akkor mindenképpen egy CMS alkalmazása, megfelelő beépülő modulokkal való megvalósítása lehetne a következő lépcsője az oldal továbbfejlesztésének. A különböző igények kielégítésére az ismert, és a dolgozatomban felsorolt tartalomkezelő rendszerek mindegyike rengeteg előre elkészített modult, plugint tartalmaz, amelyből csak ki kell választani az igényeknek megfelelőt, és feltelepíteni az alap CMS megfelelő könyvtárába.

12. Irodalom és forrásjegyzék

- awasweb.hu. (2017). *CMS vagy egyedi fejlesztés? Avagy a Joomla előnyei és hátrányai*. Forrás: awasweb.hu: <https://www.awasweb.hu/cms-vagy-egyedi-webshop>
- Balogh Csaba. (2014). *Szerver oldali programozás*. Forrás: tankonyvtar.hu: http://www.tankonyvtar.hu/en/tartalom/tamop412A/2011-0003_06_szerver_oldali_programozas/1_2_php_nyelv_altalanos_jellemzoi_IWvRuyJk38w0T7De.html
- CMS Made Simple.hu. (2011). *Dokumentáció*. Forrás: CMS Made Simple.hu: <http://cmsmadesimple.hu/dokumentacio/kezdo-lepesekek/>
- Compozer. (2017). *Kompozer - Features*. Forrás: Kompozer: <http://www.kompozer.net/features.php>
- DIMA. (2013). *Melyik ingyenes CMS-t használjam?* Forrás: DIMA: <https://www.dima.hu/tarhely-blog/melyik-ingyenes-cms-t-hasznaljam/>
- Eric Berger. (1996). *A World Wide Web születéséről*. Forrás: www.kfki.hu: <http://www.kfki.hu/~cheminfo/hun/teazo/webtori.html>
- Gorgo. (2016). *LÁTVÁNYOS WEBOLDAL KÉSZÍTÉSE INGYEN, AKÁR A MOBILUNKON IS!* Forrás: Techwork.hu: <http://techwok.hu/2016/03/15/latvanyos-weboldal-keszítése-ingyen-mobilon/>
- KISS HONLAPKÉSZÍTÉS. (2015). *AZ 5 LEGJOBB WEBLAPSZERKESZTŐ*. Forrás: Honlapkészítés: <http://kiss honlapkeszites.hu/honlapkeszites-blog/az-5-legjobb-weblapszerkeszto>
- Máté Balázs. (2016). *Mi az a HTML, mire jó, magyarázat és leírás*. Forrás: matebalazs.hu: <http://www.matebalazs.hu/html.html>
- Nagy Attila. (2017). *WordPress vagy Wix? Melyik való a Te céljaidhoz?* Forrás: wpkurzus.hu: <https://wpkurzus.hu/wordpress-vagy-wix/>
- nejanet. (2012). *Mi az a JavaScript?* Forrás: Nejanet: <http://www.nejanet.hu/~ksanyi/java/JAVAS.HTM>
- Webfolio. (2014). *EGY ÚJ HONLAP INDÍTÁS LÉPÉSEI*. Forrás: webfolio: <https://webfolio.hu/uj-honlap-inditas-lepesei/>
- WebFolio Webdesign. (2016). *A RESPONSIVE WEBOLDAL VISELKEDÉS FONTOSSÁGA ÉS SAJÁT HONLAP ELLENŐRZÉSE*. Forrás: webfolio.hu: <https://webfolio.hu/responsive-weboldal-viselkedes-fontossaga-es-ellenorzese/>
- webnode.hu. (2017). *A Webnode előnyei*. Forrás: webnode.hu: <https://www.webnode.hu/okos-funkciok/>

wikipédia. (2016). *Cascading Style Sheets*. Forrás: wikipedia.org:
https://hu.wikipedia.org/wiki/Cascading_Style_Sheets

Wikipédia. (2016). *Notepad++*. Forrás: Wikipédia:
https://hu.wikipedia.org/wiki/Notepad%2B%2B#cite_note-Gilmore-1

wikipedia.org. (2016). *HTML*. Forrás: wikipedia.org: <https://hu.wikipedia.org/wiki/HTML>

Wphu.org. (2016). *3.1. A WordPress telepítése*. Forrás: wphu.org: <https://wphu.org/konyv/3-1-a-wordpress-telepitese/>