

Szakdolgozat



Kmetti László

**Debrecen
2010**

Debreceni Egyetem
Informatikai Kar

Web design
(Web design és web-marketing)

Belső Témavezető:
Dr. Zichar Marianna
Egyetemi adjunktus

Külső Témavezető:
Vágvölgyi Csaba
Főiskolai adjunktus

Készítette:
Kmetti László
PTI Bsc
Levelező tagozat

Debrecen
2010

Tartalomjegyzék:

1. Bevezetés.....	7
1.1 Kutatási téma bemutatása.....	7
1.2 Kitűzött célok, motiváció	8
1.3 Internet kialakulása, fejlődése.	9
2. Technológiák, szoftverek (HTML, CSS, NetObject Fusion).....	11
2.1 Bevezetés:.....	11
2.2 HTML (Hyper Text Markup Language)	11
2.2.1 A HTML dokumentumokról	11
2.2.2 A HTML dokumentum fejléce	13
2.2.3 A HTML dokumentum szövegtestének felépítése:.....	14
2.2.4 A HTML dokumentum címszintjeinek felépítése	15
2.2.5 Bekezdések a HTML dokumentumban	16
2.2.6 Kereszthivatkozások HTML dokumentumok között	17
2.2.7 Karakterformátumok a HTML dokumentumban	18
2.2.8 Képek elhelyezése a HTML dokumentumban	19
2.2.9 Különböző listaformátumok a HTML dokumentumban	19
2.2.10 A HTML formátum táblázatai.....	20
2.2.11 A HTML dokumentumablak felosztása	21
2.2.12 A HTML speciális karakterei	22
2.2.13 A JavaScript használata a HTML dokumentumban.....	25
2.2.14 A HTML és a multimédia	26
2.3 CSS (Cascading Style Sheets).....	27
2.3.1 Bevezetés.....	27
2.3.2 Karakterkódolás beállítása	27
2.3.3 CSS kommentek.....	27
2.3.4 Stíluslap beágyazása HTML dokumentumba	28
2.3.5 Színek definiálása a weboldalakon.....	29
2.3.6 Weboldal háttérének beállítása.....	29
2.3.7 A dobozmodell	30
2.3.8 Linkek.....	31
2.3.9 Törött linkek ellenőrzése	32
2.3.10 Relatív és abszolút linkek.....	32
2.3.11 Kép megadása linkként	33
2.3.12 Hivatkozás letölthető fájlokra	33
2.3.13 Ugrás a honlap egy bizonyos pontjához.....	33
2.3.14 Linkek formázása stíluslappal	33
2.3.15 Karakter entitások	34
2.3.16 A szöveg részeinek logikai elkülönítése	34
2.3.17 Idézetek és címhivatkozások	34
2.3.18 Programnyelvek forráskódjának jelölése	34
2.3.19 Felső- és alsó index	35
2.3.20 Lakcímek vagy egyéb címek.....	35
2.3.21 Bekezdésen belüli szöveg formázása	36
2.3.22 Listák	37
2.3.23 Táblázatok	38
2.3.23.1 Sor- és oszlopátnyúlások megadása	39
2.3.23.2 Egy cella több sort is átfoghat:	39
2.3.23.3 Táblázat fejléce és címe	40

2.3.23.4 Rövid leírás megadása a táblázathoz.....	40
2.3.24 Méretezés	41
2.3.25 Pozicionálás.....	41
2.3.26 Blokk-szintű elemek és sorbeli elemek, elemek megjelenítése	42
2.3.27 Gördítősáv hozzáadása egy elemhez	42
2.3.28 A float és a clear tulajdonság	42
2.3.29 Formok (űrlapok)	43
2.3.29.1 Beviteli mező.....	43
2.3.29.2 Elküldő gomb	44
2.3.29.3 Jelszóbevitel	44
2.3.29.4 Többsoros beviteli mező	45
2.3.29.5 Rádiógombok	45
2.3.29.6 Jelölőnégyzetek	45
2.4 NetObjects Fusion	46
2.4.1 Bevezetés.....	46
2.4.2 A programról:	47
2.4.2.1 Verziók és a kiadás dátuma:	47
2.4.2.2 Program előnyei:	47
2.4.3 NetObject Fusson 11 indítása, kezdeti lépések	48
2.4.4 „Site” menüpont	51
2.4.4.1 Oldalunk felépítése:.....	51
2.4.4.2 Oldalak létrehozása:	52
2.4.4.3 Oldalak átnevezése:	52
2.4.4.4 Oldalak törlése:	53
2.4.4.5 Ugrás az oldalra:.....	53
2.4.5 „Page” menüpont.....	54
2.4.5.1 Keretek módosítása:	56
2.4.5.2 Háttér módosítása:	56
2.4.5.3 A honlapunk menürendszere:	56
2.4.5.4 Banner megadása:	57
2.4.5.5 Szöveg beviteli mező:	58
2.4.5.6 Szöveg formázása:.....	59
2.4.5.7 Kép beszúrása:.....	59
2.4.5.8 Linkek:	59
2.4.5.9 Táblázat:	60
2.4.5.10 Táblázat importálása:	60
2.4.5.11 Form elem:	60
2.4.5.12 Elválasztó elem:	60
2.4.5.13 Tervező nézet:	61
2.4.5.14 Kód nézet:	61
2.4.5.15 Gyors oldalnézet:.....	61
2.4.6 „Style” menüpont	62
2.4.7 „Assets” menüpont	63
2.4.9 „Preview” menüpont	64
2.4.10 „Publish Site” menüpont	64
3. Akadálymentesítés és Ergonómia	65
3.1 Bevezetés.....	65
3.2 Akadálymentesítés	65
3.2.1 Fogyatékossgal élők	66
3.2.1.1 Vakok	66

3.2.1.2 Gyengénlátók	67
3.2.1.3 Színvakok és hallássérültek	67
3.2.1.4 Mozgássérültek	68
3.2.1.5 Értelmileg visszamaradottak	69
3.2.1.6 Diszlexiások (és diszgráfiások)	69
3.2.1.7 (Fényérzékeny) epilepsziában szenvedők	69
3.2.1.8 Kulturálisan elszigetelt emberek	69
3.2.2 Technológiailag megkülönböztetettek	70
3.2.2.1 Eltérő képernyőméret	70
3.2.2.2 Elavult böngészők	70
3.2.2.3 Gyenge, lassú hardver	70
3.2.3 Speciális célcsoportok	71
3.2.3.1 Gyerekek	71
3.2.3.2 Idősek és informatikai szempontból alacsony képzettségűek	71
3.2.4 Működés ellenőrzése	71
3.2.5 W3C Validator	71
3.2.6 W3C – WAI	72
3.2.6.1 W3C	72
3.2.6.2 WCAG 1.0.....	73
3.2.6.3 WCAG 2.0.....	75
3.3 Ergonómia /felhasználóbarát oldalak/ (usability).....	76
3.3.1 Tippek egy hatékony weblap kialakításához:.....	76
3.3.2 Mitől nőhet egy felhasználó komfortérzete?	78
3.3.2.1 Honlap letöltési idő:	78
3.3.2.2 Web design:	78
3.3.2.3 Használhatóság:.....	78
3.3.2.4 Tartalom:	79
3.3.2.5 Kellemes, harmonikus színek használata:	79
3.3.3 Ergonómiai hibák	80
3.3.3.1 „I. Ergonómiai hibák okai	80
3.3.3.2 II. létező problémák.....	81
4. Grafika, megjelenés.....	82
4.1 Bevezetés:.....	82
4.2 Fontosabb formátumok:	82
4.2.1 JPEG (Joint Photographic Experts Group).....	82
4.2.2 GIF (Graphics Interchange Format)	83
4.2.3 PNG (Portable Network Graphic)	83
4.2.4 BMP (Bitmap)	84
4.2.5 SVG (Scalable Vector Graphics)	85
4.3 Templatek.....	85
4.3.1 Mik is a templatek? Mire jók?.....	85
4.3.2 Legnagyobb előnyük:	85
4.4 Grafikák.....	86
4.4.1 Animált gif-ek	86
4.4.2 Logók	86
4.4.3 Texturák	86
4.4.4 Fotók.....	86
4.5 Fontok.....	87
4.6 Színek, színekódok.....	87
4.6.1 Színekről.....	87

4.6.1.1 A színek csoportosítása	87
4.6.2 Színek nevei, hexa és RGB kódjaik:	89
4.7 Grafikai szoftverek:	90
4.7.1 Bevezetés.....	90
4.7.2 CorelDraw	91
4.7.3 Photoshop	91
5. Webes marketing, keresőmarketing	93
5.1 Bevezetés.....	93
5.2 Web-marketing	93
5.3 Keresőmarketing	93
5.3.1 Keresőoptimalizálás (SEO - search engine optimization).....	94
5.3.1.1 Benchmark, felmérés a jelenlegi helyzetről (meglévő weboldal esetén):	95
5.3.1.2 Kulcsszóelemzés	96
5.3.1.3 Domain választás.....	97
5.3.1.4 Tartalom	97
5.3.1.5 Weboldal struktúra, belső linkelés	98
5.3.1.6 Technikai elemzés, weboldal leprogramozása	99
5.3.1.7 Title és Meta Tag-ek megadása.....	99
5.3.1.8 Regisztrálás keresőkbe	100
5.3.1.9 Linképítés	100
5.3.1.10 Riport.....	101
5.3.2 Keresőhirdetés (PPC - Pay Per Click).....	102
5.3.3 Fizetett listázás (paid inclusion).....	102
6. Összefoglalás.....	103
6.1 Mikről esett szó az előző fejezetekben?	103
6.2 Mi volt a célom?.....	103
6.3 Fejlesztési lehetőségek a vizsgált szoftverben (NetObject Fusion)	103
Köszönetnyilvánítás:	105
Irodalomjegyzék:.....	106
Felhasznált könyvek	106
Internet, hivatkozások	106
Internet háttéréhez felhasznált irodalom:	106
Technológia részhez felhasznált irodalom:	106
Akadálymentesítés, ergonómia részhez felhasznált irodalom:	106
Grafika, megjelenés részhez felhasznált irodalom:.....	107
Marketing részhez felhasznált irodalom:	108
Függelék:	109

1. Bevezetés

1.1 Kutatási téma bemutatása

Az általam választott téma címe a Web design és web-marketing.

Úgy érzem a fenti témák, ahogy telnek - múlnak az évek, egyre fontosabbak lesznek. Hogy miért? Mert manapság már egyetlen komoly cég sem engedheti meg magának azt, hogy az interneten ne jelenjen meg. Arról nem is beszélve, hogy a cégeknek az internet egy viszonylag olcsó reklámfelületet is biztosít. Továbbá ahogy teret nyer az e-kereskedelem jelentős bevételtől is el lehet esni, ha nem használjuk ki internet adta lehetőségeket.

Nem mindegy azonban, hogy jelenünk meg a neten. Egy elhibázott, átgondolatlan és rosszul összerakott, megszerkesztett oldal ronthat egy cég image-én.

Nagyon sok dolog van, amiről beszélni lehetne a Web design kapcsán, amire első körben nem is gondolnánk.

A Web design egy elég összetett téma. Sok dolog tartozhat bele attól függően milyen részletesen vizsgáljuk. Sok mindent beleérthetünk.

Mint például:

- Milyen technológiát használunk a weboldalunk létrehozásához?

Ez alapvetően hathat az oldalunk kinézetére is. Jelen szakdolgozatban a HTML és stíluslapokkal (CSS) kívánok foglalkozni. A HTML-t azért vettem bele, mert úgy érzem, mielőtt a stíluslapokról szót ejtenék jó, ha az olvasó megismerkedik a HTML alapokkal. Szerintem a Web design egyik elengedhetetlen eszköze a stíluslapok használata. Ezért került a szakdolgozatomba ez a rész. A szakdolgozatom második fejezetében lesz ezekről a témákról szó.

- Milyen szoftverrel hozzuk létre a weboldalunkat (weboldalfejlesztő eszközök)?

Sokfajta szoftver van a piacon. Egyik legelterjedtebb és legjobb az Adobe cég által készített Dreamweaver, de az elég összetett és bonyolult.

Éppen ezért egy másik szoftvert mutatnék be még hozzá a NetObject Fusion-t, ami egyszerű és könnyen használható. Szintén a szakdolgozatom második fejezetében írok erről a témákról szó.

- Hogyan helyezzük el a weboldalon az anyagainkat?

Ha már adatokat teszünk fel, akkor azokat célszerű akadálymentesen és ergonomikusan elhelyezni, hogy minél több felhasználó megtekinthesse az oldalunkat. Ez természetesen mindkettő fél érdeke. Az oldalt fejlesztő és a felhasználóké is.

Ezekre a szempontokra is jó, ha odafigyelünk 2010-ben egy weboldalfejlesztés kapcsán. Szakdolgozatom harmadik fejezete foglalkozik ezekkel a témákkal.

- Lényeges szempont természetesen az oldalon alkalmazott színek, fontok, grafikák is.

Szakdolgozatom következő (negyedik) fejezete arról fog szólni, hogy milyen grafikai elemeket használhatunk az oldalunkon. Milyen elterjedt formátumok vannak jelenleg. Milyen színeket használunk, hogy olvasható legyen (és szép) az oldalunk. Hogyan tudunk gyorsan szép weboldalt létrehozni (templatek). Ezen témán belül szót ejtenék az egyik legnépszerűbb grafikai szoftverről is az Adobe cég által készített Photoshopról.

- Milyen web-marketinges eszközöket használjunk?

Ha már készen vagyunk és elkészítettük a weboldalunkat, akkor természetes elvárás, hogy oldalunk látogatott legyen. Hiába egy jól összerakott és megszerkesztett és minden szempontból design-os oldal, ha a senki sem látogatja.

Éppen ezért úgy éreztem, hogy habár nem tartozik közvetlenül a web-marketing a Web design-hoz, de célszerű erről is beszélni, ha sikeres és látogatott oldalt szeretnénk magunkénak mondani. Szakdolgozatom ötödik fejezete ezt a témát járja körbe.

1.2 Kitűzött célok, motiváció

Mint említettem a szakdolgozatom választott témája elég szerteágazó. Minden egyes témához külön könyvek, tananyagok állnak rendelkezésre, amelyek csak az adott témával foglalkoznak. Jelen szakdolgozatnak nem célja minden egyes terület elejétől a végéig való bemutatása, mivel az meghaladná a szakdolgozat formai követelményeit, helyette inkább ízelítőt szeretnék adni a fent nevezett témákból. Rávilágítani bizonyos dolgokra. Remélem, hasznos kis olvasmányt teszek le kedves olvasóm elé.

Kezdjük is bele hát. Első körben úgy illik, hogy egy kis történeti kitekintéssel kezdjem. A Web design elválaszthatatlan az internettől. Éppen ezért kezdjük is el a témát egy kis internetes történelemmel.

1.3 Internet kialakulása, fejlődése.

Szakedolgozatom jelen részében az internetről és annak kialakulásáról lesz szó. Azért gondoltam, hogy ezzel kezdeném, mert a szakedolgozatom témájához szorosan kapcsolódik az internet. Ezen fejezetben leírom röviden mi is az internet, hogyan és mi okból jött létre.

Elsőként felmerülő kérdés, hogy mi is az internet?

Az internet egy világméretű számítógépes hálózat, mely sok kis számítógépes hálózat között teremt kapcsolatot egységes hálózati protokoll (IP - Internet Protokoll) segítségével.

Az internet kifejezés az angol internetwork szóból ered, ami magyarul körülbelül azt jelenti, hogy „hálózatok hálózata”.^[3]

Hogyan alakult ki?

Története 1969-re vezethető vissza. 1969-ben az USA Hadügyminisztériuma létrehozta kísérleti jelleggel az ARPANET-et (Advanced Research Project Agency Network). Ez egy nagy távolságot áthidaló csomagkapcsolt hálózat volt. A számítógépeket többszörös utakkal kapcsolták össze azért, hogy akkor is működőképes legyen, ha egyes számítógépek vagy az őket összekötő kapcsolat kiesne. A felhasználó a hálózat bármely pontján beléphet. Elérhet a hálózat más pontján lévő számítógépet. A két számítógép között átküldendő üzeneteket csomagokra bontják. Mindegyik csomag viseli a feladó és a címzett számítógép címét. A hálózat bármely része képes adatcsomagok feladására, fogadására és küldésére.

Kezdetben 4 csomóponttal rendelkezett, amik a következők voltak: UCLA (University of California at Los Angeles), SRI (Stanford Research Institute), UCSB (University of California at Santa Barbara) és University of Utah.

Az ARPANET mellett létrejön a hasonló technológiával működő MILnet (Military Network).

1971-ben újabb oktatási és kutatási intézmények kapcsolódnak (BBN, CWRU, MIT, CMU, RAND, SDC, Harvard, UIUC), Lincoln Lab, NASA/Ames). Immár 15 csomópontra bővül a hálózat.^[1]

1973-ban kapcsolódik az ARPANET-hez az első nemzetközi kapcsolat (Anglia és Norvégia)

1982-ban összekapcsolódik az ARPANET és a MILnet.

További hálózatok is kapcsolódnak: EARN (European Academic Research Network), MINET (MILnet európai megfelelője), EUNet, USENET, BITNET, NFSNET (National Science Foundation Network - Kisebb intézmények a hozzájuk legközelebb eső nagyobb intézethez bérelt vonalon keresztül csatlakoznak. A nagyobb intézmények meg a nagyobb intézményekhez kapcsolódnak).

Az így kialakult hálózat az 1980-as évektől az intézmények közötti kapcsolattartás fő eszköze lett. Az egyre jelentősebbé váló adatforgalom indokoltá teszi a helyi hálózatok tehermentesítését. Dedikált számítógépek, routerek jelennek meg, amik leválasztják a helyi hálózatot és az átmenő forgalmat önállóan kezelik. Kialakul egy gerinchálózat (backbone).^[2]

1989-ben az ARPANET megszűnik, hogy helyét átadja a fejlettebb gerinchálózatoknak.

90-es években az internet fejlődése felgyorsul. A számítógépek és elektronikai alkatrészek árai csökkennek. Egyre több számítógép kapcsolódik a hálózatra. A 90-es

években már nem csak katonai, oktatási és tudományos intézmények, hanem a nagy szolgáltató központok is elérhetőek lesznek interneten keresztül.
Kialakul a WWW (World Wide Web – magyarul világháló).

Miről is van szó?

Ez nem más, mint interneten működő, egymással hiperlinkekkel összekötött dokumentumok rendszere, amelyet web böngésző programmal lehet elérni. Ez a program képes megjeleníteni a weblapokat. A felhasználók a hiperlinkek segítségével újabb lapokat kérhet le, amelyeken szintén lehetnek további hiperlinkek. A technológia alapjait Tim Berners-Lee dolgozta ki 1989-ben. 1993-ban a CERN bejelenti, hogy a világháló mindenki számára ingyenes. 1993-ban elindul az első magyarországi WWW szerver is a www.fsz.bme.hu.

A világháló három szabványra épül.

- 1 – URL (Uniform Resource Locator). Ez leírja, milyen egyedi címmel kell rendelkeznie az oldalnak.
- 2 – HTTP (Hyper Text Transfer Protocol) amely leírja, hogyan küldenek egymásnak adatot a böngésző és a kiszolgáló.
- 3 – HTML (Hyper Text Markup Language) az információkódolás eljárása, amellyel az oldal a különböző eszközökön megjeleníthetővé válik.^[4]

90-es években az alkalmazások (üzleti, kereskedelmi) számának robbanásszerű növekedése figyelhető meg. Online szolgáltatók, adatbázis szolgáltatók, adatbázis előállítók lesznek elérhetőek. Megjelenik a tartalomszolgáltatás, közhasznú információk szolgáltatása és a szórakoztatás (csevegés, hálózati játékok, érdekességek, letöltések {képek, zenék, videók}). Az internet globalizálódik. Tovább nő a felhasználó emberek köre. Nő a sávszélesség. Megjelennek a földrajzi határokon átvelő szolgáltatások. Magánszemélyek számára is elérhetővé válik otthonról az Internet.

2000-es években tovább nő a felhasználók tábora azáltal, hogy tovább terjednek a számítástechnikai eszközök és az internet-szolgáltatás is egyre több helyen válik elérhetővé egyre olcsóbban. Egyre több számítógépet kapcsolnak be a hálózatba.

Internet ma

Manapság az internetes technológiákat 1.4 milliárd ember használja személyes illetve üzleti célokra (például: adatok keresése, szolgáltatások és áruk megrendelése).

Jelenleg kutatások zajlanak az internet technológiai problémáinak megoldására (például az IPv6 címzési rendszer bevezetése). A jelenleg használt IPv4-es címzési rendszer ugyanis kifogyóban van, erre jelentene megoldást az IPv6-os címzési rendszer. Az internetet elérhetővé kell tenni további 5 milliárd ember számára a jelenlegi 1.4-1.5 milliárd helyett.

Magyarországon 2006-ban a háztartások körülbelül 32%-ban volt internet elérés, 2007-ben körülbelül 38%-ban. Ez időszak alatt nőtt a szélessávú^[*] internet elérések aránya is 22%-ról 33%-ra. Ezen időszak alatt az európai uniós átlag a 42%. Ebben a listában országunk a 18. helyen áll.

A függelék részbe található néhány táblázat, ami a hazai internetes helyzetet mutatja be.

* A szélessáv mást jelent az EU-ban és mást jelent az USA-ban.

2. Technológiák, szoftverek (HTML, CSS, NetObject Fusion)

2.1 Bevezetés:

Most, hogy áttekintettük az internet rövid történetét, ebben a fejezetben megismerkedünk a web design-hoz szorosan kötődő két nyelvel. Egész pontosan a HTML (mint alapnyelv) és a CSS (mint stílusnyelv) nyelvekről lesz szó. Ebben a fejezetben szó esik még egy web design szoftverről is aminek a neve NetObject Fusion.

Kezdjük akkor a fejezetet a HTML nyelv ismertetésével, amihez később hozzájön majd a CSS stílusnyelv ismertetése is.

2.2 HTML (Hyper Text Markup Language)

2.2.1 A HTML dokumentumokról

A HTML nyelv a weboldal szerkezetét leíró nyelv. Weboldalak készítéséhez fejlesztették ki. Mára már szabvánnyá vált W3C támogatásával. Aktuális változata a 4.01. A nyelvet folyamatosan fejlesztik. Tervek szerint a XHTML kiszorítja, de az is lehet, hogy a HTML 5 veszi át a helyét^[*]. Álljon itt a HTML verzióiról egy kis történelmi áttekintés:

A HTML verziói

Verzió	Melyik évben vált ajánlássá?	Jelenlegi státusz
HTML 2.0	1995. november	Elavult
HTML 3.2	1997. január	Elavult
HTML 4.0	1997. december	Elavult
HTML 4.01	1999. december	Ajánlás
HTML 5	-	Munkavázlat

A HTML fájl egy szöveges fájl. Bármely szövegszerkesztővel létrehozható, ill. módosítható. Szövegen kívül tartalmaz úgynevezett „HTML-tag”-eket — más néven formázóutasításokat —, valamint megjelenítendő objektumokra történő hivatkozásokat is. Ezek a HTML formázóutasítások (más szóval: parancsok, elemek) befolyásolják a dokumentum megjelenítését, kapcsolatait. Ezeket az utasításokat a böngészőprogram értelmezi és végrehajtja.

A HTML utasításokat a szövegben „<” és „>” jelek közé kell zárni. Egy-egy utasítás - HTML parancs, HTML elem - hatását általában a záró utasításpárja szünteti meg, amely megegyezik a nyitó utasítással, csak a „/” jel vezeti be (természetesen a „<” és a „>” jelek között). Az utasítások nagy része opcionális elemeket is tartalmazhat, melyek csak a nyitóutasításban szerepelhetnek, a záróban nem. Az opciók értékadásánál az idézőjel nem mindig kötelező, csak ajánlott. A HTML utasítás kulcsszávaiban nem különböztetjük meg kisbetűket és nagybetűket.

* Friss hírek szerint valószínű, hogy a HTML 5 megnyeri ezt a csatát.

Négyfajta leíró elem található a HTML-ben:

- Strukturális elemek

Leírják az adott szöveg „célját” például `<h1>Téma 1</h1>` mint első szintű címsor (alcím).

- Prezentációs szimbólumok

Leírják, az adott szöveg hogy nézzen ki. Elavult. Helyette CSS használata javasolt.

- Hiperszöveg (hypertext)

Segítségével kapcsolat létesíthető a dokumentum egyes elemei és más dokumentumok között.

- Eszköz elemek

Segítségükkel gombok, listák, beviteli mezők hozhatók létre. ^[2]

HTML dokumentum felépítése

Dokumentum kezdete és vége:

Minden HTML formátumú szövegfájl a `<HTML>` utasítással kezdődik és a `</HTML>` záró utasítással végződik. Ezen elemek közé kell zárni a teljes dokumentumot - formázóutasításokkal és hivatkozásokkal együtt.

Fejléc:

A fejlécet a `<HEAD>` utasítás jelzi és a `</HEAD>` utasítás zárja.

Dokumentumtörzs:

Ezt a részt jeleníti meg a web böngésző. Kezdetét a `<BODY>` és végét a `</BODY>` utasítások jelzik.

Egy tipikus HTML dokumentum felépítése:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title>A dokumentum neve</title>
  </head>
  <body>
    <H1> Ez egy alcím </H1>
    <p> Első bekezdés </p>
    <p> Második bekezdés </p>
    <!--HTML comment (nem jelenik meg a dokumentumban)-->
  </body>
</html>
```

2.2.2 A HTML dokumentum fejléce

Mint fentebb említettem a fejléct a <HEAD>, és a végét </HEAD> utasítás jelzi.

A fejlécelemekre jellemző, hogy a böngészőablakban nem jelennek meg! Ezért a World Wide Weben szereplő dokumentumok fejlécének csak nagyon kis hányada tartalmaz a címen kívül egyéb információkat.^[1]

A fejlécben használható további utasítások:

<TITLE> és a </TITLE>

dokumentum címe

<BASE HREF="protokoll://gépnév/elérési_út">

Utasításban szereplő URL határozza meg a báziscímet, melyből a relatív címeket értelmezni kell. Az intelligens kiszolgálók korában nem kötelező megadni.

<BASEFONT SIZE="szám">

Ezzel az utasítással állítható be a dokumentumban az alapértelmezett betűméret.

<ISINDEX>

Jelzi a keresőrendszerek számára, hogy index-dokumentumról van szó.

<LINK>

Dokumentum kapcsolatait más dokumentumokkal, stíluslappal, címszalaggal, stb.

<META NAME="mező" CONTENT="érték">

Jelezheti a keresőrendszerek számára a dokumentum-adatbázisba kerülő adatokat, pl. a dokumentum alkotóját, a létrehozó programot, rövid tartalmat, stb.

Fejlécre példa:

<HTML>

<HEAD>

<TITLE>Fejléc címe</TITLE>

<META NAME="Author" CONTENT="Kmetti László">

<BASEFONT SIZE=8>

</HEAD>

<BODY>

A dokumentum törzse.

</BODY>

</HTML>

2.2.3 A HTML dokumentum szövegtestének felépítése:

Minden HTML formátumú szövegfájl a <BODY> és a </BODY> utasításokkal közrezárt részében tartalmazza a megjelenítendő részét.

Ezen elemek között kell elhelyezni mindent: a szöveget, hivatkozásokat, képeket, stb.

A dokumentum törzsében használható további utasítások:

```
<BODY      BACKGROUND="fájlnév.kit"      BGCOLOR="színkód"
TEXT="színkód" LINK="színkód" VLINK="színkód" ALINK="színkód">
```

A dokumentumtörzsre vonatkozó fenti előírások is szerepelhetnek opcióként.

- A BACKGROUND="elérési_út/fájlnév.kit" Mi legyen a dokumentum háttere. Ezt állíthatjuk be.

- BGCOLOR="színkód" opcióval a háttérszín állíthatjuk be.

- TEXT="színkód" opcióval a dokumentum szöveg színét jelölhetjük ki.

- LINK="színkód" opció a hivatkozások megjelenési színét határozza meg.

- VLINK="színkód" pedig, a már bejárt hivatkozásokat jelölő színt határozza meg.

Természetesen egyszerre több opció is szerepelhet, de egyik sem kötelező.

Dokumentumtörzsre példa:

```
<BODY BACKGROUND="pelda3.gif"
      BGCOLOR="#FF3333"
      TEXT="#000099"
      LINK="#993399"
      VLINK="#009999">
```

```
<P> A szemléltetés kedvéért az alábbi szöveg széles szegéllyel határolt! <P>
```

```
<TABLE BORDER=12>
```

```
<TH><TD>
```

```
<H1><A HREF="#pelda">Ez itt egy hivatkozás,</A> kattints rá!</H1>
```

```
</TABLE>
```

```
<P> <IMG SRC="k11.gif" ALT="Görgetősáv"> <P>
```

```
<A NAME="pelda"> Ide mutat a hivatkozás!</A>
```

```
</BODY>
```

2.2.4 A HTML dokumentum címszintjeinek felépítése

A HTML formátumú szövegfájlban 6 mélységig definiálhatunk címeket.

A legfelső szintű címet a `<H1 ALIGN="hely">` és a `</H1>` utasítaspárral kell közrezárni.

A második szintet a `<H2 ALIGN="hely">` és a `</H2>` utasítások határolják, és így tovább a hatodik szintig.

Minden szint más-más betűformátumban jelenik meg a dokumentumban, a böngészőprogram beállításától függően.

ALIGN opcióval igazíthatjuk a címeket, melynek lehetséges értékei:

- left
- center
- right

NOWRAP opcióval megakadályozhatjuk a cím betördelését több sorba.

`<BANNER>`, `</BANNER>` egy különleges címszalagot jelöl. Tulajdonsága, hogy nem lehet görgetni, a dokumentum többi részével ellentétben.

A címek csak a szemlélő számára keltik a tagoltság érzetét, a valóságban nem tagolják fizikailag szakaszokra a dokumentumot. Ezt a tagolást a `<DIV CLASS="osztály" ALIGN="hely">`, `</DIV>` utasításokkal lehet meghatározni, ahol a CLASS opció sorolja a megfelelő SGML osztályba a szakaszt, az ALIGN pedig a szakasz igazítási formátumát írja elő. Az automatikus tördelést itt is megakadályozza a NOWRAP opció, ez esetben a szakasz tördelését a `<P>` vagy a `
` utasításokkal lehet szabályozni.

Példa a címszintekre:

`<BANNER>Címszalag</BANNER>`

`<H1 ALIGN="left">Legfelső szintű címsor</H1>`

`<H2 ALIGN="center">Második szintű alcímsor</H2>`

`<H3 ALIGN="right">Harmadik szintű alcímsor</H3>`

`<H4 NOWRAP>Negyedik szintű alcímsor</H4>`

`<H5>Ötödik szintű alcímsor</H5>`

`<H6>Hatodik szintű alcímsor</H6>`

`<DIV ALIGN="center">`

Ez egy szakasz, melyben a szöveg elvileg azonos módon

- középre igazítva - kezelendő.

`</DIV>`

2.2.5 Bekezdések a HTML dokumentumban

A HTML dokumentumok alapvetően bekezdésekre tagolódnak.

A bekezdések kezdetét a <P> utasítás jelzi a böngészőprogram számára. A bekezdés a következő bekezdés kezdetét jelző utasításig tart, tehát nincs záró utasításpárja.

A legtöbb böngészőprogram két bekezdés között egy üres sort szúr be megjelenítéskor!

A bekezdés elem hordozhat magában a bekezdés stílusát meghatározó opciókat.

A bekezdés igazítását a <P ALIGN="hely"> formájú utasítással szabályozhatjuk.

NOWRAP opcióval az automatikus tördelést lehet megtiltani a böngészőprogram számára.

Amennyiben tördelhetetlen szóközt igényel a szöveg, akkor az egyszerű szóköz helyett alkalmazzuk a különleges karaktert.

Amennyiben egy bekezdésen belül mindenképpen új sort szeretnénk kezdeni, akkor a
 utasítást kell használni. (Nincs záró párja!)

Példa a bekezdések használatára:

```
<P>Ez alapértelmezett (balra igazított) bekezdés.  
  <P ALIGN=left>Ez balra igazított bekezdés.  
  <P ALIGN=center>Ez középre igazított bekezdés.  
  <P ALIGN=right>Ez jobbra igazított bekezdés.  
  <P ALIGN=justify>Ez sorkizárt bekezdés lenne.  
<P>  
<NOBR>Ez itt egy tördeletlen - jó hosszú - bekezdés,  
      amelynek eltűnik a vége a semmiben  
      .....</NOBR>  
<P>  
Itt egy sortörés elem található,  
<BR>  
melynek hatására új sorban folytatódik a szöveg,  
és nem maradt&nbsp;ki&nbsp;üres&nbsp;sor.
```

2.2.6 Kereszthivatkozások HTML dokumentumok között

A HTML dokumentumokban használhatunk egymás dokumentumaira való hivatkozásokat (linket). Ezekre kattintva, ha aktiváljuk őket, akkor a hivatkozással összefüggésben lévő szöveghez jutunk el. A hivatkozó utasítások megjelenési formája sokféle lehet.

Hivatkozás egy másik fájlra/dokumentumra:

`` utasítás jelzi a hivatkozás kezdetét, és a `` utasításelem zárja le a végét.

Hivatkozás egy másik fájlra/dokumentumra adott részére:

`` utasítás jelzi a hivatkozás kezdetét, és a `` utasításelem zárja le a végét.

Ebben az esetben a hivatkozott fájl kell, hogy tartalmazzon egy olyan részt (könyvjelzőt), ahová a hivatkozás mutat. Ezt a részt a `` és a `` utasítások határolják.

2.2.7 Karakterformátumok a HTML dokumentumban

A HTML formátumú szövegfájlban is használhatjuk a szövegszerkesztőkben megszokott karakterformátumokat. Az alábbi táblázat a formázás kezdő és záró utasítása között a mintát is tartalmazza.

Kezdő elem	Ilyen betűformátumot eredményez	Záró elem
	Félkövér betűformátumot eredményez	
<I>	Dőltbetűs formátumot eredményez	</I>
<U>	Aláhúzott formátumot eredményez	</U>
<S>	Áthúzott formátumot eredményez	</S>
<TT>	Fixpontos betűket eredményez	</TT>
	Kiemeli a szöveget	
<CITE>	Idézetekre használható	</CITE>
<VAR>	Változónevet jelöl	</VAR>
	Ez is egy kiemelési lehetőség	
<CODE>	Kódoknál használjuk	</CODE>
<SAMPLE>	Minták jelzésére használjuk	</SAMPLE>
<KBD>	Billentyűfelirat jelzése	</KBD>
<BQ>	Idézet megjelenítése	</BQ>
<BIG>	Nagyméretű betűformátumot eredményez	</BIG>
<SMALL>	Kisméretű betűformátumot eredményez	</SMALL>
_{	Alsóindexet eredményez	}
^{	Felsőindexet eredményez	}
<BLINK>	Villogó szöveget eredményez	</BLINK>
	A részleteket lásd lentebb	

A , utasításpárral direkt módon előírhatók a megjelenő szöveg betűinek a jellemzői. A FACE opciót nem szokás használni, mert nem valószínű, hogy minden rendszerben rendelkezésre áll pl. az ARIAL CE FÉLKÖVÉR betűtípus. A COLOR opció pontosan meghatározza a megjelenítendő szöveg színét. A SIZE opcióban egy számot megadva a betűméretet határozza meg direkt módon. (A SIZE opcióban előjeles szám is szerepelhet, ami az alapbetűtípushoz viszonyított méretet jelöl.)

2.2.8 Képek elhelyezése a HTML dokumentumban

A HTML formátumú dokumentumban képeket - grafikákat - is elhelyezhetünk.

Utasítás formája egyszerű esetben:

```
<IMG SRC="elérési_út/fájlnev.kit">
```

Utasítás formája, ha formázási elemeket is beleteszünk:

```
<IMG ALIGN="hely" SRC="elérési_út/fájlnev.kit" HSPACE="vszám"
VSPACE="fszám" WIDTH="szméret" HEIGHT="mmeret" UNITS="egység"
USEMAP="#jelző" ISMAP ALT="szöveg">
```

- ALIGN opció meghatározza a kép igazításának módját.

Lehetséges értékei: top, middle, bottom, left, right.

- HSPACE a kép melletti vízszintes térközt határozza meg.

- VSPACE pedig a függőleges térközt határozza meg.

- WIDTH a szélességét határozza meg a képnek.

- HEIGHT a magasságát adja a képnek.

- UNITS milyen egységben (pixel vagy en)?

- ALT azt a szöveget adja meg, amelyet nem grafikus böngészők használata esetén meg fog jelenni a kép helyett.

- USEMAP, ISMAP összetartozó opciók, a kép különböző területeihez különböző hipertext hivatkozásokat rendelhetnek.

2.2.9 Különböző listafomátumok a HTML dokumentumban

A HTML formátumú szövegfájlban használhatunk listákat is.

Léteznek számozott, számozatlan és leíró listák. A leíró listában az elemekhez tartozhat egy hosszabb leírás is.

Utasítások:

, Számozott listára.

, Számozatlan listára.

<LH>, </LH> Listafejléc számozott és számozatlan listára.

 Listák sorai.

A listasorok egy utasítástól a következő utasításig, ill. a lista záró eleméig tartanak.

<DIR>, </DIR> Könyvtár típus (számozatlan listák esetén).

<MENU>, </MENU> Menülista (számozatlan listák esetén).

<DL>, </DL> Leíró listára.

<DT> Leíró lista esetén, a listák egyes alkotóelemeinek a kezdetét jelöli.

<DD> Leíró lista esetén, a listák egyes alkotóelemeinek a kezdetéhez tartozó leírás kezdetét jelöli.

2.2.10 A HTML formátum táblázatai

A HTML formátumnak ez az utasításcsoportja képes a legváltozatosabb szöveg-, és képmegjelenítési formák előállítására.

Utasítások:

<TABLE>, </TABLE> Táblázat kezdete és vége

<CAPTION>, </CAPTION> Táblázat címe.

A cím <CAPTION ALIGN="hely"> formájú megadással igazítható.

<TR> Táblázat sora. Következő <TR>-ig, ill. a táblázat végéig tart.

<TH> Oszlopfejléc.

<TD> Táblázat cellája.

A táblázat nyitóutasítása részletesen:

<TABLE BORDER="szám" ALIGN="hely" COLSPEC="oszlopjellemzők"
UNITS="egység" NOWRAP CELLPADDING="pszám"
CELLSPACING="kszám" BGCOLOR="színkód">

- BORDER rácsozat szélességét határozza meg. (0 esetén nincs rácsozat.)
- ALIGN a teljes tábla elhelyezkedését határozza meg (left, right, center lehet).
- COLSPEC egy oszlop igazítását és szélességét adja meg. Egy oszlopra vonatkozóan egy betű és szám egybeírva (pl.: L12 C24 R10), melytől a következő oszlop értékeit egy köz választja el.
- UNITS a számokhoz tartozó mértékegységet jelöli ki (en, relative - oszlopszélességhez -, pixel).
- NOWRAP opció a cellák szövegének tördelését tiltja le.
- BGCOLOR a táblázat háttérszínét határozza meg.

2.2.11 A HTML dokumentumablak felosztása

Egyetlen böngészőablakban több HTML dokumentum is megjeleníthető a következő utasítások segítségével:

Utasítások:

```
<FRAMESET>, </FRAMESET>  
<FRAME>
```

<FRAMESET ROWS="oszlophatárok"> Képernyő függőleges felosztása.

<FRAMESET COLS="sorhatárok"> Képernyő vízszintes felosztása.

<FRAME SRC="objektum"> Betölt egy megadott objektumot, mely objektum lehet egy teljes HTML fájl, annak egy meghatározott része, ill. egy kép

```
<FRAME NAME="név" SRC="objektum" SCROLLING="érték"  
MARGINWIDTH="szám" MARGINHEIGHT="szám">
```

Kitöltendő keret viselkedését szabályozza.

- NAME opció, a szkrollozást letilthatja SCROLLING="no" kiegészítés (ezenkívül a „yes” és az „auto” értékeket veheti fel a SCROLLING opció),
- MARGINWIDTH Kereten belüli margók szélessége.
- MARGINHEIGHT Kereten belüli margók magassága.

A <FRAMESET>, <FRAMESET> utasításpárral határolt területnek meg kell előznie a <BODY> utasítással kijelölt dokumentumtörzset! Sőt egy <NOFRAMES> utasítással kell jelezni a dokumentum azon részének kezdetét, amelyet akkor kell a böngészőnek megjelenítenie csak, ha nem ismeri a keretutasításokat. És csak ez a <NOFRAMES>-mel kezdődő rész tartalmazhatja a <BODY> és a </BODY> utasításpárt.

2.2.12 A HTML speciális karakterei

A HTML dokumentumban nem csak az angol ABC alfanumerikus jeleit alkalmazhatjuk, hanem ékezetes betűket vagy speciális jeleket is használhatunk. Ahhoz, hogy ezeket a karaktereket elérjük, a HTML speciális jeleit kell használnunk.

Lehetséges az ESCAPE szekvencia alapján történő jelölése és ISO-kód szerinti megadása is ezen speciális jeleknek.

Az ABC betűi és kódjaik								
Elnevezés	Jel	ESC	ISO		Elnevezés	Jel	ESC	ISO
Nagy A betű	A	A	A		Kis a betű	a	a	a
Nagy Á betű	Á	Á	Á		Kis á betű	á	á	á
Nagy B betű	B	B	B		Kis b betű	b	b	b
Nagy C betű	C	C	C		Kis c betű	c	c	c
Nagy D betű	D	D	D		Kis d betű	d	d	d
Nagy E betű	E	E	E		Kis e betű	e	e	e
Nagy É betű	É	É	É		Kis é betű	é	é	é
Nagy F betű	F	F	F		Kis f betű	f	f	f
Nagy G betű	G	G	G		Kis g betű	g	g	g
Nagy H betű	H	H	H		Kis h betű	h	h	h
Nagy I betű	I	I	I		Kis i betű	i	i	i
Nagy Í betű	Í	Í	Í		Kis í betű	í	í	í
Nagy J betű	J	J	J		Kis j betű	j	j	j
Nagy K betű	K	K	K		Kis k betű	k	k	k
Nagy L betű	L	L	L		Kis l betű	l	l	l
Nagy M betű	M	M	M		Kis m betű	m	m	m
Nagy N betű	N	N	N		Kis n betű	n	n	n
Nagy O betű	O	O	O		Kis o betű	o	o	o
Nagy Ó betű	Ó	Ó	Ó		Kis ó betű	ó	ó	ó
Nagy Ö betű	Ö	Ö	Ö		Kis ö betű	ö	ö	ö
Nagy Ő betű	Ő	Õ	Ô		Kis ő betű	ő	õ	ô
Nagy P betű	P	P	P		Kis p betű	p	p	p
Nagy Q betű	Q	Q	Q		Kis q betű	q	q	q
Nagy R betű	R	R	R		Kis r betű	r	r	r

Nagy S betű	S	S	S	Kis s betű	s	s	s
Nagy T betű	T	T	T	Kis t betű	t	t	t
Nagy U betű	U	U	U	Kis u betű	u	u	u
Nagy Ú betű	Ú	Ú	Ú	Kis ú betű	ú	ú	ú
Nagy Ü betű	Ü	Ü	Ü	Kis ü betű	ü	ü	ü
Nagy Ű betű	Ű	Û	Û	Kis ű betű	ű	û	û
Nagy V betű	V	V	V	Kis v betű	v	v	v
Nagy W betű	W	W	W	Kis w betű	w	w	w
Nagy X betű	X	X	X	Kis x betű	x	x	x
Nagy Y betű	Y	Y	Y	Kis y betű	y	y	y
Nagy Z betű	Z	Z	Z	Kis z betű	z	z	z

Néhány speciális jel és kódjaik							
Elnevezés	Jel	ESC	ISO	Elnevezés	Jel	ESC	ISO
Tabulátor					Soremelés		
	
Szóköz		 		Felkiáltójel	!	!	!
Idézőjel	"	"	"	Számjel	#	#	#
Dollárjel	\$	$	\$	Százalékjel	%	%	%
Angol és jel	&	&	&	Aposztróf	'	'	&
Bal zárójel	(((Jobb zárójel)))
Aszteriszk	*	*	*	Pluszjel	+	+	+
Vessző	,	,	,	Kötőjel	-	-	-
Pont	.	.	.	Perjel	/	/	/
Nullás	0	0	0	Egyes	1	1	1
Kettes	2	2	2	Hármas	3	3	3
Négyes	4	4	4	Ötös	5	5	5
Hatos	6	6	6	Hetes	7	7	7
Nyolcas	8	8	8	Kilences	9	9	9
Kettőspont	:	:	:	Pontosvessző	;	;	;
Kisebb jel	<	<	<	Egyenlőségjel	=	=	=
Nagyobb jel	>	>	>	Kérdőjel	?	?	?
Kukac	@	@	@	Bal zárójel	[[[
Visszaper jel	\	\	\	Jobb zárójel]]]

Hatványjel	^	^	^	Aláhúzás	_	_	_
Vissza aposztróf	`	`	`	Bal kapocs	{	{	{
Függőleges		|		Jobb kapocs	}	}	}
Tilde jel	~	~	~	Alsó aposztróf	,	‚	
Alsó idézőjel	„	„		Kereszt	†	†	
Kettős kereszt	‡	‡		Ezrelék	‰	‰	
Felső vessző	‘	‘		Felső vessző	’	’	
Kettős vessző	“	“		Kettős vessző	”	”	
Szorás jel	•	•		Mínusz előjel	–	–	
Kivonás jel	—	—		Trade Mark	™	™	
Cent jele	¢	¢	¢	Font jele	£	£	£
Csővezeték jel		¦	¦	Paragrafus jel	§	§	§
Umlaut	¨	¨	¨	Copyright	©	©	©
Bal tört idézet	«	«	«	Lágy kötőjel		­	­
Registered TM	®	®	®	Fok jele	°	°	°
Plusz-mínusz	±	±	±	Ékezet	´	´	´
Mikro	μ	µ	µ	Bekezdés vége	¶	¶	¶
Középen pont	·	·	·	Jobb tört idézet	»	»	»
Szorás kereszt	×	×		Scharfes s	ß	ß	ß
Osztás jel	÷	÷					

A következő táblázat tartalmazza a magyar ékezetes betűk Escape-szekvenciáit és ISO-kódját:

Jel ESCAPE ISO-kód

Á Á Á
 É É É
 Í Í Í
 Ó Ó Ó
 Ő Õ Ô
 Ö Ö Ö
 Ú Ú Ú
 Ű Û Û
 Ü Ü Ü

Jel ESCAPE ISO-kód

á á á
 é é é
 í í í
 ó ó ó
 ő õ ô
 ö ö ö
 ú ú ú
 û û û
 ü ü ü

2.2.13 A JavaScript használata a HTML dokumentumban

Egy HTML formátumú szövegfájl tartalmazhat JavaScript programnyelven megírt kódsorokat is a következő utasítások között:

```
<SCRIPT LANGUAGE="JavaScript"> , </SCRIPT>
```

A JavaScriptről röviden:

- A JavaScript változókat és függvényeket a dokumentum fejlécében szokás definiálni a <HEAD> és a </HEAD> utasítások között.

Az így definiált függvényeket meghívhatjuk, változókat használhatjuk (lehet hivatkozni rá) a szöveg HTML elemeiben. A JavaScript programocskákat ne tévesszük össze a JAVA programozási nyelvvel.

Az alábbi JavaScript programocska faktoriális számol:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
  <HEAD>
    <TITLE>Javascript munkalap</TITLE>
    <META content="text/html; charset=windows-1250" http-
equiv=Content-Type>
    <SCRIPT language=JavaScript>
      <!--
      function nfakt(a)
      {
        if (a==0) {return 1}
        else {return a*nfakt(a-1)}
      }
      //-->
    </SCRIPT>
    <META content="Microsoft FrontPage 5.0" name=GENERATOR>
  </HEAD>
  <BODY>
    <B><FONT size=+2>
    <CENTER>n-faktoriális </CENTER></FONT></B><FONT size=+1>
    <CENTER>
    <P>
    <FORM name=szamelmelet><INPUT maxLength=8 name=egyik
size=8 type=number>! <INPUT name=egy
onclick=document.szamelmelet.ertek.value=nfakt(document.szamelmel
et.egyik.value) type=button value=" = ">
    <INPUT maxLength=40 name=ertek size=40 type=number>
    <HR>
    </FORM>
    <P></P>
    </CENTER>
    </FONT>
  </BODY>
</HTML>
```

2.2.14 A HTML és a multimédia

A HTML formátumú szövegfájlban nemcsak képeket helyezhetünk el, hanem multimédiás fájlokra történő utasítást is.

Utasítások:

<BGSOUND SRC="hangfájl.wav" LOOP="szám">

A letöltés közben lejátszandó hangeffektust határozza meg.

- LOOP-ban megadhatjuk, mennyiszer ismételjen.

<SOUND SRC="hangfájl.wav">

A böngésző megjelenít egy hivatkozást a lejátszandó hangeffektusra, amelyet aktiválva az adott hangfájlt lejátszsa.

Mozgóképet játszhat le vele.

 utasítással is kombinálható.

<EMBED SRC="fájlnév.kit" WIDTH="vszám" HEIGHT="fszám">

Ezzel az utasítással a HTML-ben külön nem támogatott típusú fájlok beágyazhatók a dokumentumba. Az így beágyazott objektumok saját megjelenítő programot igényelnek, amelyet a böngésző feladata meghívni. ^[3]

Mostanra átnéztük a HTML alapjait. Bővítsük akkor tovább a HTML lehetőségeit a CSS nyelv (stíluslapok) használatával^[*].

* Megjegyzés: A következő részben nem csak tisztán CSS kódok fognak szerepelni. Elő fognak benne fordulni HTML kódok is.

2.3 CSS (Cascading Style Sheets)

2.3.1 Bevezetés

CSS jelentése „Cascading Style Sheets”, azaz egymásba ágyazott stíluslapok.

HTML oldalainkat tudjuk vele megváltoztatni. Meghatározhatjuk, hogyan jelenjenek meg az egyes HTML elemek. Milyen legyen a színük, méretük, stb.

A CSS egyik alapvető tulajdonsága a folyamatos stíluslap - HTML lap kapcsolat. A lapok szerzői az általuk kedvelt stílust egyszer rögzítik, és hozzákapcsolhatják minden általuk készített HTML laphoz. Stíluslappal tervezni nem nehéz, de szükséges hozzá némi alapvető HTML ismeret. Az alábbi pár oldalon megpróbálom minél rövidebben ismertetni a CSS lehetőségeit. Ennek megfelelően nem leszek túl bőbeszédű és próbálok csak a lényegre koncentrálni.

2.3.2 Karakterkódolás beállítása

A stíluslap-fájl legelső sorába ajánlott a karakterkódolásra vonatkozó információt írni. Itt is az UTF-8 kódolást ajánlom, mint a HTML esetében. Ekkor így kell kezdődnie a stíluslapnak:

```
@charset "utf-8";
```

2.3.3 CSS kommentek

CSS-ben a kommenteket /* és a */ jelek közé tehetjük.

Egy megjegyzés több sort is átfoghat. A megjegyzéseket nem veszi figyelembe a böngésző, semmilyen hatással nincsenek a kinézetre

Példa:

```
@charset "utf-8";
```

```
/*Ez egy megjegyzés.*/
```

```
h1, h2, h3 {  
    color: red; /*Ez egy megjegyzés.*/  
}/*Ez  
egy  
többsoros megjegyzés.*/
```

2.3.4 Stíluslap beágyazása HTML dokumentumba

Az elkészült stíluslap bármennyi HTML oldalhoz használható egyidejűleg. Viszont ahhoz, hogy a stíluslap hatással legyen rá, össze kell kötni a HTML fájlt és a stíluslapot. Ezt háromféleképpen tehetjük meg:

1. Külső stíluslapok

Ha egy HTML fájlra alkalmazni kívánunk egy stíluslapot, akkor a fejrészbe (<head>...</head>) a következőt írjuk, ha a két fájl azonos mappában van:

```
<head>
  <link href="stiluslapneve.css" rel="stylesheet" type="text/css">
</head>
```

- A rel és a type attribútumnak mindig ez kell, hogy legyen az értéke, ha stíluslapot ágyazunk be.
- A href attribútumnak a stíluslap elérési útvonalát kell tartalmaznia a HTML fájlhoz képest.

2. Belső stíluslapok

Olyan stílusok megadásakor érdemes használni ezt a módszert, aminél tudjuk, hogy csak egyetlen weboldalhoz fogjuk felhasználni. Ugyanis ha egy stíluslapot már kettő, vagy több weboldalhoz is felhasználunk, akkor ajánlott külön stíluslapban tárolni őket, és a fenti 1. módszerrel társítani a weboldalhoz. A <head>...</head> részbe kell beírni a stíluslapot a <style>...</style> címkék közé.

```
<head>
  <style type="text/css">
    h1 {
      color: red;
      text-align: center;
    }

    h2 {
      color: blue;
      text-decoration: underline;
      margin-bottom: 1em;
    }
  </style>
</head>
```

3. Szövegszámítógépes stílusok

A style attribútumot bármelyik címkén belül használhatjuk. Ezt akkor érdemes használni, ha egy stílus csak egyszer, vagy nagyon kevésszer fordul elő a weboldalon, és ezért nem akarjuk külön kiírni a külső stíluslapban.

Ha mondjuk alkalmaztuk a fenti egyszerű stíluslapot az oldalunkra, és minden címsor piros lett és középre igazított, de mi egyetlen címsornál azt szeretnénk, ha mégis kék és jobbra igazított lenne, akkor ezt így adhatjuk meg a kívánt címsornál (a szövegszámítógépes stílus felülbírálja az összes többi stílust, erről később szó lesz):

```
<h1 style="color: blue; text-align: right;">
```

2.3.5 Színek definiálása a weboldalakon

A weboldal megjelenítésére képes eszközök (pl. monitor, nyomtató) a színeket három alapszínből, a piros-zöld-kék (R-G-B) kombinációból keveri ki.

Azt, hogy melyik színből mennyit akarunk felhasználni, háromféleképpen adhatjuk meg:

- Hexadecimális számokkal.

Az első kettő szám a piros, a középső kettő a zöld, az utolsó kettő pedig a kék mennyiségét jelzi 16-os számrendszerben.

pl. #1E0EE2

- RGB értékek százalékos formában történő megadása.

Ekkor azt mondjuk meg, hogy a három alapszínből százalékosan milyen arányban forduljon elő a színünkben.

Pl. color: rgb(30%,20%,50%);

- RGB értékek 0 és 255 közti tízes számrendszerbeli számmal való megadása.

Pl. color: rgb(34,55,201);

- Használhatunk előre definiált angol színneveket is.

Ezeket a W3C határozta meg a szabványban, ezek nevei alább láthatóak.

Az alábbi példák mindegyike a fehér színt állítja elő a h1-es címsorra:

(Persze valós példánál csak egyszer adjunk meg a color értékét!)

```
h1 {
  color: #FFFFFF;
  color: rgb(100%,100%,100%);
  color: rgb(255,255,255);
  color: white;
}
```

2.3.6 Weboldal háttérének beállítása

background-color

Egy elem háttérszínét adja meg.

Lehetséges értékei: color-rgb, color-hex, color-name, transparent.

background-image

Háttérképet állít be. Lehetséges értékei: url(URL), none.

background-repeat

Beállítja, hogy a háttérkép vízszintesen, függőlegesen, mindkét módon, vagy ahogy se ismétlődjön. Lehetséges értékei: repeat, repeat-x, repeat-y, no-repeat.

background-position

A háttérkép pozícióját állítja be.

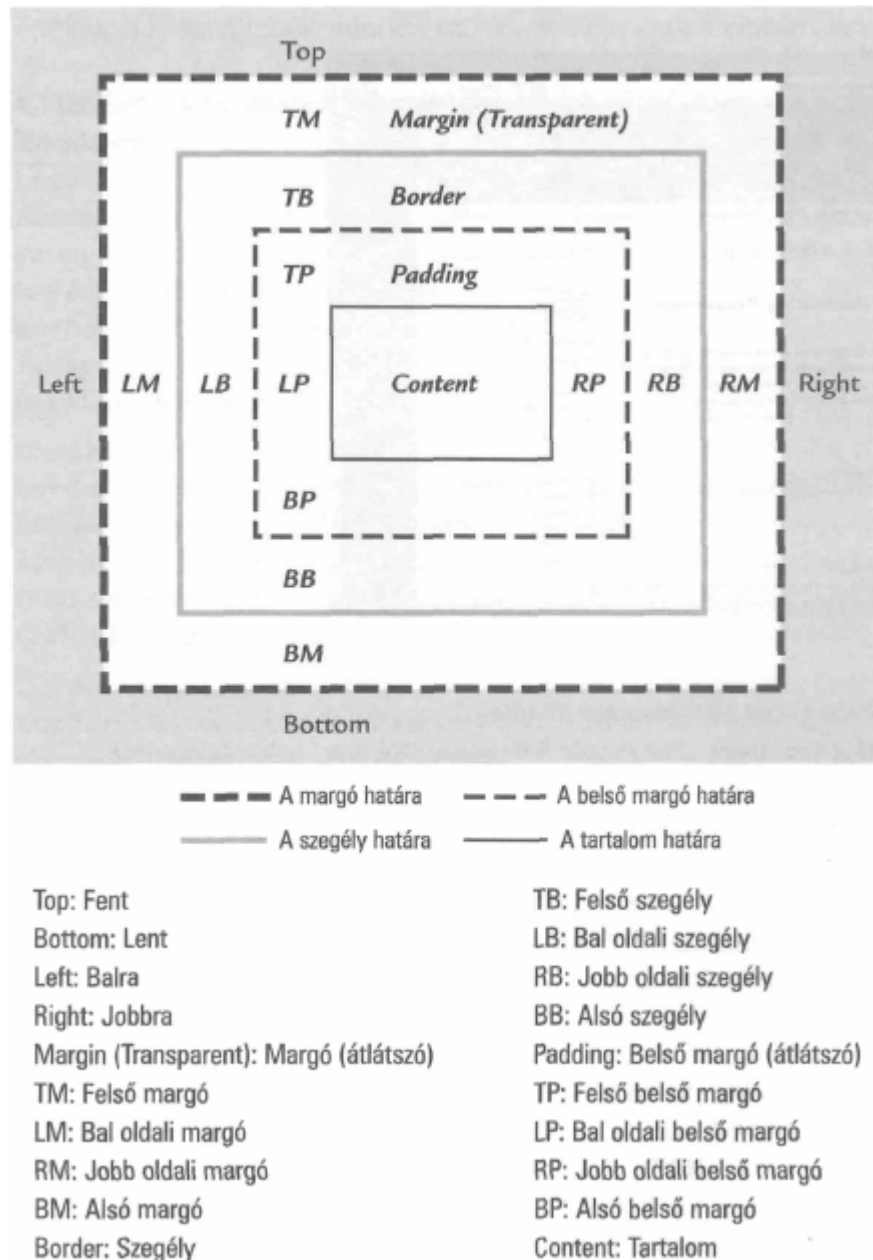
Lehetséges értékei: top left, top center, top right, center left, center center, center right, bottom left, center right, bottom left, bottom center, bottom right, x% y%, xpos, ypos.

2.3.7 A dobozmodell

A weboldal minden eleme egy téglalap alakú területen jelenik meg.

PL: bekezdés, táblázatot, képet, stb. Ezek az elemek akkor ezek egy téglalap alakú területet, egy „dobozt” fognak elfoglalni. Ezekhez a dobozhoz hozzárendelhető: belső margó (padding), szegélyt (border), illetve margót (margin).

Az alábbi ábra illusztrálja a dobozmodellt.



Forrás: Virginia DeBolt: HTML és CSS webszerkesztés stílusosan.
2005 Kiskapu Kft. ^[1]

Keretek (tulajdonság, leírás és lehetséges értékek)

`border-style`, `border-top-style`, `border-right-style`, `border-bottom-style`, `border-left-style`

Keret stílusának megadása. Ha nem állítjuk be, nem fog látszani a keret!

Lehetséges értékei: `none`, `hidden`, `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset`, `outset`.

`border-width`, `border-top-width`, `border-right-width`, `border-bottom-width`, `border-left-width`

Keret vastagságának megadása.

Lehetséges értékei: `thin`, `medium`, `thick`, `length`.

`border-color`, `border-top-color`, `border-right-color`, `border-bottom-color`, `border-left-color`

Keret színének megadása.

Lehetséges értékei: `color-rgb`, `color-hex`, `color-name`.

`border`, `border-top`, `border-right`, `border-bottom`, `border-left`

Rövidebben lehet megadni a keret tulajdonságait, ebben a sorrendben: vastagság, stílus, szín.

Lehetséges értékei: `border-width`, `border-style`, `border-color`.

Belső margó (tulajdonság, leírás és lehetséges értékek)

`padding`, `padding-top`, `padding-right`, `padding-bottom`, `padding-left`

Belső margó megadására szolgál.

Lehetséges értékei: `length`, `%`.

Külső margó

`margin`, `margin-top`, `margin-right`, `margin-bottom`, `margin-left`

Külső margó megadására szolgál.

Lehetséges értékei: `auto`, `length`, `%`.

2.3.8 Linkek

A linkek (hivatkozások) kötik össze a weboldalakat. Az egész Internet linkekből áll. Az alábbi példa egy Google-ra mutató link:

```
<a href="http://www.google.hu/" title="Erre a linkre kattintva a Google magyar nyelvű oldala fog megjelenni">Látogasd meg a Google-t!</a>
```

Magyarázat:

```
<a> (anchor, azaz horgony), </a>
```

Ezzel a paranccsal hozhatunk létre linket.

`href` attribútum

Ezután adjuk meg annak a webhelynek a címét, ahová a link mutat, a fenti példában ez a Google magyar webhelye. Az `<a>...` címkék közé írt szöveg fog megjelenni a link szövegeként.

`title` attribútum

Segítségével segédszöveget írhatunk a linkhez, ami akkor fog megjelenni, ha az egérmutatót kis ideig pihentetjük az adott linken.

2.3.9 Törött linkek ellenőrzése

Ajánlott a linkeket átnézni, mielőtt feltöltjük honlapunkat az Internetre.

Ha olyan oldalra mutat a linkünk, ami nem található, akkor a látogató a „404 - A keresett oldal nem található” üzenetet fogja kapni. Az ilyen üzenet külső oldal esetén még csak kellemetlen, de ha a link a saját oldalunkra mutat, és akkor kap a látogató ilyen üzenetet, akkor valószínűleg nagyon hamar el fogják elhagyni az oldalt.

Ezt az ellenőrzést szerencsére nem kell manuálisan megtenni, mert van rá program:

<http://validator.w3.org/checklink>

2.3.10 Relatív és abszolút linkek

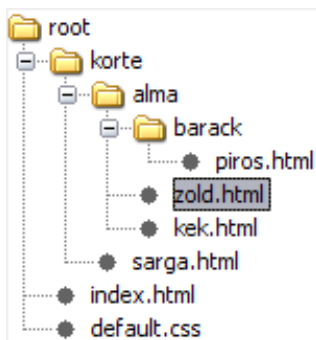
Kétféle link létezik:

- Relatív link
- Abszolút link

Abszolút linkekre már láttunk példát pár sorral ezelőtt így nézzük a relatív linkeket.

Relatív link az a link, amelyik a rá hivatkozott dokumentumhoz képest valahol máshol helyezkedik el a webszerveren belül.

Tekintsük az alábbi könyvtárszerkezetet egy webszerveren belül:



Ebben a könyvtárszerkezetben a relatív linkek így alakulnak:

`Menj a kek.html oldalra!`

Hivatkozás alkönyvtárban az azonos könyvtárban lévő fájlra.

`Menj a piros.html oldalra!`

Hivatkozás alkönyvtárban lévő fájlra:

`Menj a sarga.html oldalra!`

Hivatkozás egy könyvtárral feljebb lévő fájlra.

`Menj az index.html oldalra!`

Hivatkozás két könyvtárral feljebb lévő fájlra.

2.3.11 Kép megadása linkként

Kép is megadható link-ként.

Íme, egy példa rá:

```
<a href="http://validator.w3.org/check?uri=referer"></a>
```

2.3.12 Hivatkozás letölthető fájlokra

Ugyancsak link segítségével kínálhatunk letöltésre akármilyen kiterjesztésű fájlt.

Például PDF dokumentumot így tudunk belinkelni:

```
<a href="mypdf.pdf">Kattintson ide a PDF letöltéséhez!</a>
```

2.3.13 Ugrás a honlap egy bizonyos pontjához

Címkével honlapunk egy bizonyos pontjához tudjunk ugrani, ha rákattintunk a linkre.

Az alábbi példával a 4. és a 7. fejezethez lehet ugrani:

```
<p><a href="#f4">Ugrás a 4. fejezethez!</a></p>
<p><a href="#f7">Ugrás a 7. fejezethez!</a></p>
```

```
<h2>1. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2>2. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2>3. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2 id="f4">4. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2>5. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2>6. fejezet</h2>
<p>Ez itt a fejezet tartalma stb. stb.</p>
<h2 id="f7">7. fejezet</h2>
```

2.3.14 Linkek formázása stílussal

4 állapotunk lehet. Ezekre definiáltak alosztályokat. Nem kötelező mind a négy állapotra külön stílust megadni. Azonban ha megadjuk őket, akkor pontosan a fenti példában látható sorrendben alkalmazzuk őket! Ha nem így teszünk, akkor felülírhatják egymást az egyes állapotok.

A 4 állapot a következő:

:link

A még meg nem látogatott linkeket választja ki.

:visited

A már meglátogatott linkeket választja ki.

:hover

Azt a linket választja ki, amelyik felett éppen áll az egér.

:active

Azt linket választja ki, amelyiken állva az egérrel a felhasználó lenyomta az egérgombot, de még nem engedte fel.

2.3.15 Karakter entitások

Vannak speciális karakterek, amiket ha meg akarnánk jeleníteni a böngészőben, bajban lennénk. Ilyen például a „<” vagy „>” jel. Ahhoz hogy ezeket a karaktereket meg tudjuk jeleníteni erre találták ki a karakter entitásokat.

Ez azt jelenti, hogy ha „<” jelet akarunk kiírni, akkor a forráskódba a következőt kell beírni: „<”. Az entitás & jellel kezdődik, aztán szóköz nélkül jön az entitás neve, majd a pontosvessző következik, ami az entitás végét jelzi. Hivatkozhatunk név helyett számmal is egy entitásra.

Ebben az esetben a következő formát használjuk: „&#” Az általános szintaktika tehát a következő: &entitásneve; vagy &#entitásszáma. Az entitás nevét ajánlott használni inkább, mint az entitás számát.

2.3.16 A szöveg részeinek logikai elkülönítése

Ha logikailag szeretnénk elkülöníteni bizonyos részeket, akkor a HTML kódban bizonyos címkéket kell használnunk. Erre jó példa a strong és az em címke. A ... címkék közé kell tenni az erősen kihangsúlyozott szöveget, az ... címkék közé pedig a gyengébben kihangsúlyozott szöveget. A strong közé tett szöveg vastagon, az em közé tett pedig dőlt betűvel fog megjelenni.

Példa:

```
<p><strong>Ez egy nagyon fontos információ</strong></p><p><em>Ez egy fontos információ</em></p>
```

2.3.17 Idézetek és címhivatkozások

```
<blockquote>...</blockquote>
```

Hosszabb rész idézésére használatos.

```
<q>...</q>
```

Kisebb, pár soros idézeteket a <q>...</q> címkék közé kell írni.

```
<cite>...</cite>
```

Az idézett személy nevét a <cite>...</cite> címkék közé kell írni!

Példák:

```
<p><cite>Petőfi</cite> írta apjáról: „<q>Szemében »<q>mesterségem</q>« /  
Most is nagy szálla még;<br> / Előítéletét az / Évek nem szüntették.</q>”</p>
```

```
<p><cite>Ady Endre</cite> írta:</p>
```

```
<blockquote>
```

```
<p>Nem az a fontos, hogy meddig élünk, / Hogy meddig lobog vérünk, / Hogy  
csókot meddig kérünk és adunk, / Hanem az, hogy volt egy napunk, / Amiért  
érdemes volt élni.</p>
```

```
</blockquote>
```

2.3.18 Programnyelvek forráskódjának jelölése

```
<code>...</code>
```

A code címke programnyelvi kód jelölésére szolgál. Gyakran használják a

```
<pre>...</pre>
```

címkével együtt formázásra.

Példa:

```
<p>Ez a lehető legegyszerűbb Java program:</p>
```

```
<pre>
```

```
<code>
```

```
public class Main {
```

```

        public static void main(String[] args) {
            System.out.println("Helló világ!");
        }
    }
</code>
</pre>

```

<kbd>...</kbd>

A kbd címke olyan szöveg jelölésére szolgál, amit a felhasználónak kell begépelnie.

Példa:

```

    <p>A fenti program lefordításához, majd futtatásához ezt kell beírni a
    parancssorba:</p>
    <p><kbd>javac Main.java<br>
    java Main</kbd></p>
    A fenti program lefordításához, majd futtatásához ezt kell beírni a
    parancssorba:
    javac Main.java
    java Main
    A samp címke egy kód kimenetét mutatja meg.
    <p>A fenti program kimenete:</p>
    <p><samp>Helló világ!</samp></p>

```

<var>...</var>

A var címke egy program változójának jelölésére szolgál.

Példa:

```

    <p>A programban az <var>args</var> változó a program indításakor
    esetlegesen megadott paramétereket tartalmazza.</p>

```

A programban az args változó a program indításakor esetlegesen megadott paramétereket tartalmazza.

2.3.19 Felső- és alsó index

^{...}

Felső index jelölésére.

_{...}

Alsó index jelölésére.

Példa:

```

    <p>Az angol így jelöli a sorszámokat:
    1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup>.</p>
    <p>A víz vegyjele: H<sub>2</sub>O</p>

```

2.3.20 Lakcímek vagy egyéb címek

<address>...</address>

Lakcímet, vagy egyéb címeket az address címkével kell jelölni.

Példa:

```

    <p>Irodánk elérhetősége:</p>
    <address>
    1111 Budapest<br>

```

Pitypang utca 111/C.
</address>

2.3.21 Bekezdésen belüli szöveg formázása

Az alábbi utasításokkal szövegrészeket emeljünk ki, színezzünk át stb. Az ilyen formázandó szövegrészek kijelölésére kell használni a span címkét. Ehhez a címkéhez mindig adjuk meg a class értéket, hiszen így fogjuk tudni később elérni a stíluslapból.

color

Szöveg színének megadása.

Lehetséges értékei: color-rgb, color-hex, color-name.

text-align

Szövegigazítás: bal, jobb, közép, sorkizárt.

Lehetséges értékei: left, right, center, justify.

text-decoration

Aláhúzást, felülhúzást, áthúzást, vagy villogást állít be a szövegre.

Lehetséges értékei: none, underline, overline, line-through, blink.

text-indent

Bekezdés első sorának beljebb kezdése.

Lehetséges értékei: length, %.

line-height

Sorközök megadására szolgál.

Lehetséges értékei: normal, number, length, %.

letter-spacing

A karakterek közti szóköz nagyságát szabályozza.

Lehetséges értékei: normal, length.

word-spacing

A szavak közti szóköz nagyságát szabályozza.

Lehetséges értékei: normal, length.

text-transform

Szöveget kisbetűssé, nagybetűssé, vagy minden szót nagy kezdőbetűssé változtat.

Lehetséges értékei: none, capitalize, uppercase, lowercase.

font-family

A használandó betűcsaládról ad prioritásos listát: Ha az első megadott font nem található, akkor próbálkozik a másodikkal és így tovább.

Lehetséges értékei: Betűcsalád neve, pl: „Arial”, „Times New Roman”, „Verdana”; Vagy: általános típus: serif, sans-serif, cursive, fantasy, monospace

font-style

Dőlt betűs szöveg megjelenítését teszi lehetővé.

Lehetséges értékei: normal, italic, oblique.

font-size

Szöveg méretét lehet megadni.

Lehetséges értékei: xx-small, x-small, small, medium, large, x-large, xx-large, pt, px, smaller.

font-weight

Szöveg vastagságát adja meg.

Lehetséges értékei: normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900.

font-variant

Kiskapitális szöveg megadása.

Lehetséges értékei: normal, small-caps.

font

Rövidítő tulajdonság az összes eddigi tulajdonság beállításához, a jobb oldali sorrendben.

Lehetséges értékei: font-style, font-variant, font-weight, font-size/line-height, font-family.

2.3.22 Listák

Háromféle listát készíthetünk. Számozatlan, számozott és definíciós listákat.

Számozatlan lista

A számozatlan lista egy felsorolás, minden listaelem előtt egy kis pöttyel.

Parancsszava:

`` (unordered list, azaz számozatlan lista)

Listaelemeket a `` (list element, azaz listaelem) és a `` címkék közé kell tenni.

Példa:

```
<ul>
<li>Listaelem 1</li>
<li>Listaelem 2</li>
<li>Listaelem 3</li>
</ul>
```

Számozott lista

Számozott listát a `` (ordered list, azaz számozott lista) címke vezeti be.

Példa:

```
<ol>
<li>Listaelem 1</li>
<li>Listaelem 2</li>
<li>Listaelem 3</li>
</ol>
```

Definíciós lista

Definíciós listákat a `<dl>` (definition list, azaz definíciós lista) címke vezeti be és a `</dl>` címke zárja. Fogalmat a `<dt>` és `</dt>` címkék közé tesszük.

Magyarozatát pedig a `<dd>` és `</dd>` címkék közé helyezzük.

Példa:

```
<dl>
  <dt>Listaelem 1</dt>
  <dd>Listaelem 1 magyarázata</dd>

  <dt>Listaelem 2</dt>
  <dd>Listaelem 2 magyarázata</dd>

  <dt>Listaelem 3</dt>
  <dd>Listaelem 3 magyarázata</dd>
</dl>
```

Beágyazott listák

Listákat egymásba is ágyazhatjuk.

Listák formázása stíluslappal

list-style-type

A listaelemek előtti kis jel megadására szolgál.

Lehetséges értékek: disc, circle, square, none, decimal, lower-alpha, upper-alpha, lower-roman, upper-roman.

list-style-image

Kép beállítása a listaelemek jelzésére.

Lehetséges értékek: none, url.

list-style-position

A listaelemek előtti kis jel vagy kép helyzetének megadására szolgál.

Lehetséges értékek: outside, inside.

list-style

Rövidítő tulajdonság.

Lehetséges értékek: list-style-type, list-style-position, list-style-image.

2.3.23 Táblázatok

A táblázatot a <table>...</table> címkék közé kell írni. A táblázat egy sora a <tr>...</tr>, ezen belül pedig a cellák a <td>...</td> címkék közé kerülnek.

Példa:

```
<table>
  <tr>
    <td>első sor, első oszlop</td>
    <td>első sor, második oszlop</td>
    <td>első sor, harmadik oszlop</td>
  </tr>
  <tr>
    <td>második sor, első oszlop</td>
    <td>második sor, második oszlop</td>
    <td>második sor, harmadik oszlop</td>
  </tr>
</table>
```

```

<tr>
  <td>harmadik sor, első oszlop</td>
  <td>harmadik sor, második oszlop</td>
  <td>harmadik sor, harmadik oszlop</td>
</tr>
</table>

```

Ha azt akarjuk, hogy egy cella üresen maradjon, akkor a tartalma legyen (non-breaking space).

Például:

```

<table>
  <tr>
    <td>első sor, első oszlop</td>
    <td>&nbsp;</td>
    <td>első sor, harmadik oszlop</td>
  </tr>
  <tr>
    <td>&nbsp;</td>
    <td>második sor, második oszlop</td>
    <td>második sor, harmadik oszlop</td>
  </tr>
</table>

```

2.3.23.1 Sor- és oszlopátnyúlások megadása

A következő példán azt láthatjuk, hogy egy táblázat egy cellája több oszlopot is átfoghat. Íme:

```

<table>
  <tr>
    <td>első sor, első oszlop</td>
    <td>első sor, második oszlop</td>
    <td>első sor, harmadik oszlop</td>
  </tr>
  <tr>
    <td colspan="3">A második sor egyetlen cella lett, a colspan attribútum
értékét 3-ra állítottam.</td>
  </tr>
  <tr>
    <td>harmadik sor, első oszlop</td>
    <td>harmadik sor, második oszlop</td>
    <td>harmadik sor, harmadik oszlop</td>
  </tr>
</table>

```

2.3.23.2 Egy cella több sort is átfoghat:

```

<table>
  <tr>
    <td>első sor, első oszlop</td>
    <td rowspan="3">A második oszlop egyetlen cella lett, a rowspan
attribútum értékét 3-ra állítottam.</td>
    <td>első sor, harmadik oszlop</td>

```

```

</tr>
<tr>
  <td>második sor, első oszlop</td>
  <td>második sor, harmadik oszlop</td>
</tr>
<tr>
  <td>harmadik sor, első oszlop</td>
  <td>harmadik sor, harmadik oszlop</td>
</tr>
</table>

```

2.3.23.3 Táblázat fejléce és címe

<th>...</th>

Lehetőség van arra is, hogy kiemeljük a táblázat fejlécét (a táblázat legelső sorát, oszlopát, vagy mindkettőt) és címet rendeljünk hozzá. A fejléc egy celláját ne a <td>...</td> címkék közé, hanem a <th>...</th> címkék közé írjuk, ekkor a böngésző alapból megvastagítja és középre rendezi a benne lévő szöveget.

A táblázat fejléce lehet egyrészt az oszlopok felett. Ekkor mindegyik th címkének adjuk meg a scope="col" attribútumot, ami azt jelenti, hogy az adott fejléc az alatta lévő oszlophoz tartozik.

Ha egy fejléc a mellette lévő sorhoz tartozik, akkor a scope="row" attribútumot kell beállítani.

<caption>...</caption>

A táblázat címét a <caption>...</caption> címkék közé írjuk, rögtön a kezdő <table> címke után!

Megadhatunk a sorokhoz és az oszlopokhoz is fejlécet. Ekkor a sorokhoz megadott fejlécekhez adjuk meg a scope="row" attribútumot, az oszlopokhoz megadott fejlécekhez pedig a scope="col" attribútumot.

2.3.23.4 Rövid leírás megadása a táblázathoz

<table summary=" ">

A summary-vel, a table címke attribútumaként minden adattáblához adjuk meg a rövid szöveges leírását. Ez tartalmazza, hogy miről szól a táblázat, milyen adatok vannak benne és hogyan kell értelmezni. A summary értéke nem jelenik meg a grafikus böngészőkben, ez a képernyő felolvasóknak szól.

2.3.24 Méretezés

Az alábbi tulajdonságok az elemek méretét állítják be.

Tulajdonságok:

width

Egy elem szélességét adja meg.

Lehetséges értékek: auto, %, px, em.

height

Egy elem magasságát adja meg.

Lehetséges értékek: auto, %, px, em.

max-width

Egy elem maximális szélességét adja meg.

Lehetséges értékek: auto, %, px, em.

min-width

Egy elem minimális szélességét adja meg.

Lehetséges értékek: auto, %, px, em.

max-height

Egy elem maximális magasságát adja meg.

Lehetséges értékek: auto, %, px, em.

min-height

Egy elem minimális magasságát adja meg.

Lehetséges értékek: auto, %, px, em.

2.3.25 Pozícionálás

Position tulajdonsággal tudjuk megadni az elemek helyzetét.

Alapértéke: static (az elemek a normál helyzetükben jelennek meg)

Tulajdonság:

position

Egy elem helyzetét adja meg.

Lehetséges értékek: static, relative, absolute, fixed.

vertical-align

Egy elem függőleges igazítását állítja be. Csak sorbeli elemekre, vagy táblázatcellákra érvényes! A vertical-align tulajdonsággal tudunk például képet a szöveghez igazítani.

Lehetséges értékek: baseline, sub, super, top, text-top, middle, bottom, text-bottom, length, %.

2.3.26 Blokk-szintű elemek és sorbeli elemek, elemek megjelenítése

Display tulajdonsággal a blokk-szintű és a sorbeli megjelenést tudjuk szabályozni. Továbbá sokféle megjelenés elérhető lenne vele, de Internet Explorer semelyik verziója nem támogatja, ezért használata nem ajánlott.

Tulajdonság:

display

Egy elem megjelenítési módját adja meg.

Lehetséges értékek: inline, block, list-item, run-in, compact, marker, table, inline-table, table-row-group, table-header-group, table-footer-group, table-row, table-column-group, table-column, table-cell, table-caption, none.

2.3.27 Gördítősáv hozzáadása egy elemhez

Tulajdonság:

overflow

Gördítősáv megjelenését állíthatjuk be vele egy olyan elemre, amiben nem fér ki a tartalom. Alapérték a visible.

Lehetséges értékek: visible, hidden, scroll, auto.

2.3.28 A float és a clear tulajdonság

Gyakran kell használni ezt a két tulajdonságot.

Tulajdonság:

float

Egy elemet úsztat jobbra vagy balra.

Lehetséges értékek: none, left, right.

clear

Megakadályozza, hogy egy elem bal, jobb, vagy mindkét oldalon legyen egy olyan elem, amire be van állítva a float tulajdonság.

Lehetséges értékek: none, left, right, both.

2.3.29 Formok (űrlapok)

Űrlapok segítségével kérhetünk be adatokat a honlapot látogatóktól.

`<form>...</form>`

Ha formot szeretnénk beágyazni a dokumentumba, az egészet a `<form>...</form>` címkék közé kell tenni. A form tartalmát `<div>...</div>` vagy `<p>...</p>` címkék közé írjuk. Az `<input...>` pedig egy beviteli mező.

Példa:

```
<form action="parser.php" method="post">
  <div>
    <label for="lastname">Vezetéknév: </label>
    <input type="text" name="lastname" id="lastname">
    <br>
    <label for="firstname">Keresztnév: </label>
    <input type="text" name="firstname" id="firstname">
    <br>
    <label for="email">E-mail: </label>
    <input type="text" name="email" id="email">
    <br>
    <input type="submit" value="Elküld">
  </div>
</form>
```

2.3.29.1 Beviteli mező

Egysoros beviteli mezőre példa:

```
<form action="parser.php" method="post">
...
<input type="text" id="lastname" name="lastname">
...
</form>
```

Az egysoros beviteli mező fontosabb attribútumai:

`type="text"`

Ez mondja meg a böngészőnek, hogy ez egy egysoros szöveges beviteli mező.

`name="lastname"`

A `name` attribútum azért kell, mert a szerveroldali parancsfájl, aminek elküldi a böngésző az adatokat és amit az `action="parser.php"` attribútummal adtunk meg, ez alapján fogja megtalálni a megfelelő értékeket. A `parser.php` parancsfájl ugyanis `<név>=<érték>` párokként kapja meg a beírt adatokat, ahol a név a `name` attribútum értéke minden egyes formelemhez, az érték pedig a felhasználó által beírt adat. Arra figyeljünk, hogy a név csak az angol ábécé kisbetűit, és számokat tartalmazzon, és számmal NE kezdődjön a név!

`id="lastname"`

Ez a `label` elem miatt kell, ami a beviteli mező előtti feliratot definiálja:

```
<label for="lastname">Vezetéknév: </label>
<input type="text" name="lastname" id="lastname">
```

`size="4"`

Mennyi karakter szélességű legyen a mező.

maxlength="4"

Beírható karakterek számának korlátozása.

value="Kezdeti szöveg a mezőben"

Itt adható meg az alapértelmezett szöveg (ezt meg lehet változtatni).

2.3.29.2 Elküldő gomb

Az elküldő gombot ezzel tudjuk beszúrni:

```
<input type="submit" id="submitButton" value="Elküld">
```

Fontosabb attribútumai:

type="submit"

Ennek hatására lesz ez egy elküldő gomb. Amikor rányomunk, a begépelt adatokat a böngésző feltölti a szervernek, az action attribútumban megadott program felé.

value="Elküld"

A gomb feliratát állítja be.

2.3.29.3 Jelszóbevitel

Bejelentkezéskor szükségünk van jelszóbevitelre alkalmas szövegmezőre. Itt a beírt karakterek helyett csak csillagok vagy kis körök jelennek meg.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
```

```
"http://www.w3.org/TR/html4/strict.dtd">
```

```
<html lang="hu">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>HTML fájl</title>
```

```
</head>
```

```
<body>
```

```
<form action="passwordparser.php" method="post">
```

```
<div>
```

```
<label for="user">Felhasználó: </label>
```

```
<input type="text" name="user" id="user">
```

```
<br>
```

```
<label for="pass">Jelszó: </label>
```

```
<input type="password" name="pass" id="pass">
```

```
<br>
```

```
<input type="submit" value="Elküld" id="submitButton">
```

```
</div>
```

```
</form>
```

```
</body>
```

```
</html>
```

2.3.29.4 Többsoros beviteli mező

`<textarea>...</textarea>`

Többsoros beviteli mezőt létrehozására alkalmas.

`cols`

Mennyi oszlopa legyen.

`rows`

Hány sora legyen.

2.3.29.5 Rádiógombok

Ha több lehetőség közül kell egyet kiválasztani, akkor használjuk.

A `type` értékének most "radio"-t kell megadni.

Megjelenik egy új attribútum, a `value`: ez legyen ugyanaz, mint a rádiógomb előtti felirat.

Fontos: a két rádiógomb `name` attribútuma legyen azonos, mert csak így lehet pontosan egyet választani közülük. Viszont a két `id` attribútumnak különböznie kell.

2.3.29.6 Jelölőnégyzetek

Jelölőnégyzeteket akkor alkalmazunk, amikor több lehetőség közül kell egyet vagy többet kiválasztanunk.^[1]

Röviden ennyit szerettem volna leírni a CSS és HTML nyelvekről. Próbáltam minél tömörebben fogalmazni és csak a lényeget leírni. Tudom, lenne még miről beszélni még a HTML és a CSS nyelv kapcsán is, de akit bővebben érdekel a téma az úgy is sok oktatóanyagot talál ezekkel kapcsolatban az interneten.

A fejezet elején szóba került a NetObject Fusion mint egy lehetséges web design szoftver. Most következő pár oldalon ennek a programnak a rövid és tömör bemutatására vállalkozok.

2.4 NetObjects Fusion

2.4.1 Bevezetés

Ez egy professzionális HTML-szerkesztő és site-menedzser, mely beépített böngészővel, grafikus és kódszintű szerkesztővel, valamint egy holnapvarázslóval is rendelkezik.

A HTML-szerkesztő programok olyan programok, melyek HTML oldalak elkészítésére alkalmasak. Mivel a HTML egyszerű szöveg egy szöveges állományban, ezek legegyszerűbben egy sima szöveget szerkeszteni képes programmal is létrehozhatók. Alább felsorolok egy pár ilyen programot is.

A modernebb elvárásoknak azonban inkább az olyan programok felelnek meg amelyek WYSIWIG elv alapján működnek. Segítséget nyújtanak a kódok bevitelére és ellenőrzik a bevitt adatokat ezzel jelentősen lerövidítve az elkészítés időtartamát.

Mit értünk WYSIWIG elv alatt?

Angol szóösszetétel: What You See Is What You Get

Magyarul: Amit látsz, azt kapod.

Most nézzünk pár programot kódszerkesztő programokra és a grafikus alapú programokra:⁴

Szöveges alapú, kódszerkesztő programok

1st Page 2000

Alleycode HTML Editor (van beépített előnézet funkció)

Arachnophillia

BBEdit, van beépített előnézet funkció

BlueFish

CoffeeCup HTML Editor 2006 (van beépített előnézet funkció)

ConText

HTML Kit

HTMLPad

Html-Mankó (Editor2Web)

Macromedia-tól a Hometown (van beépített előnézet funkció)

Notepad++

Notepad2

Quanta Plus

SCREAM

Textile

Grafikus alapú, WYSIWYG programok

Dreamweaver

Sharepoint designer (Régi neve:Microsoft FrontPage)

WebEditor

NetObjects Fusion

Én ezek közül a NetObject Fusion-t fogom bemutatni egy alap weboldal létrehozása közben.

2.4.2 A programról:

A programot a NetObject Inc hozta létre. Alapvetően kétféle nyelven érhető el a program. Angolul és németül.

Támogatott operációs rendszerek Windows (összes verzió) és Mac OS

Honlapjuk: <http://netobjects.com/>

2.4.2.1 Verziók és a kiadás dátuma:

- NetObjects Fusion 1 (1996) (Windows/Mac)
- NetObjects Fusion 2 (1997) (Windows/Mac)
- NetObjects Fusion 3 (1998) (Windows/Mac)
- NetObjects Fusion 4 (1999) (Windows)
- NetObjects Fusion 5 (2000) (Windows)
- NetObjects Fusion MX (2001) (Windows)
- NetObjects Fusion 7 (2002) (Windows)
- NetObjects Fusion 7.5 (2003) (Windows)
- NetObjects Fusion 8 (2004) (Windows)
- NetObjects Fusion 9 (2005) (Windows)
- NetObjects Fusion 10 (2006) (Windows)
- NetObjects Fusion 11 (2008) (Windows)

2.4.2.2 Program előnyei:

- A programmal könnyedén lehet kialakítani webhelyünk oldalstruktúráját.
- A „vidd és dobd” eljárással kedvünkre változtatgathatunk az oldalunk felépítésén.
- A Fusion a bonyolultabb eljárásokat többnyire automatizálja, így jelentősen megkönnyíti a készítő dolgát.
- Könnyen létrehozhatók vele dinamikus HTML elemek is.
- Hatalmas és igényes stílustárral van felvértezve.
- A NetObjects Fusion rendkívül gazdag választékot nyújt a beépülő és kiegészítő lehetőségekre, animációk, javascript, flash, formok készítésére, illetve azok beágyazására.
- Aprólékos, szinte mindenre kiterjedő beállítások biztosítják a honlapok egyéni felépítését és azok menedzselését a szinkronizáló segítségével.
- A program erőssége a site menedzser - feltöltő, szinkronizáló, valamint a multimédiás adatok támogatása.

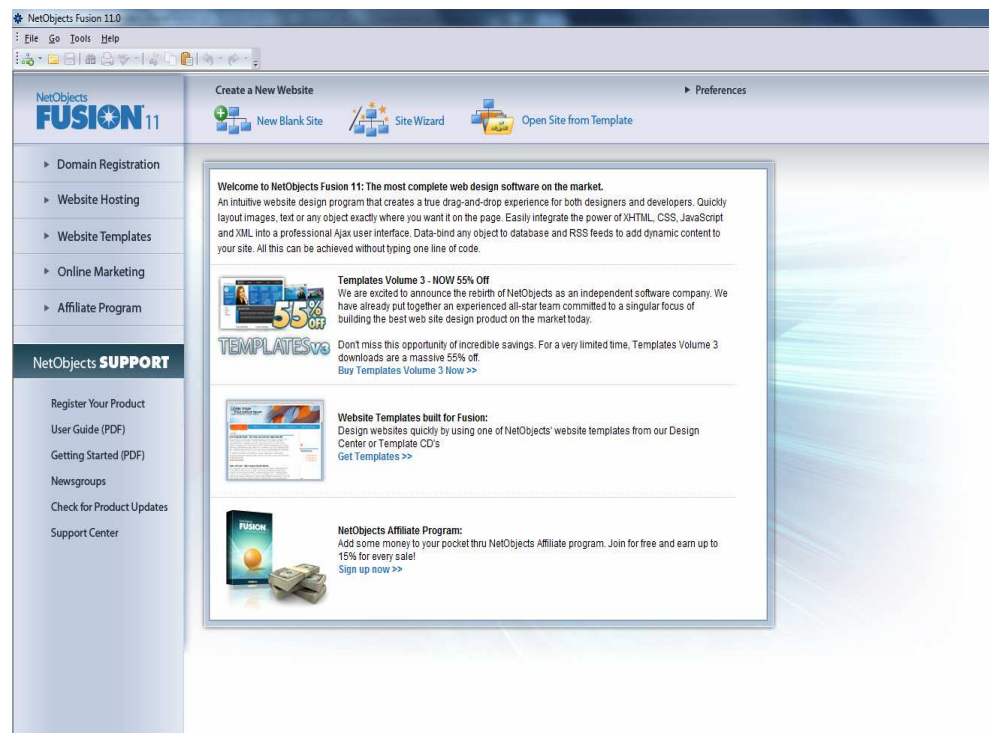
Miután sorra vettük a program milyen besorolásba tartozik, milyen verziói léteztek és mik az előnyei és a hátrányai akkor kezdjük is el.

A weboldal létrehozásához a legutolsó, azaz a 11-es shareware verziót fogom használni. A program teljes áru változata 150 dollárba kerül.

Megemlíteném, hogy a programból létezik egy 7.5-ös Essentials verzió is, ami teljesen ingyenesen használható. Valamint létezik még a 10-es verzióból is egy 30 napos trial verzió is, amit a megadott ideig szintén ingyenesen használhatunk.

2.4.3 NetObject Fusson 11 indítása, kezdeti lépések

A bejelentkező képernyő:

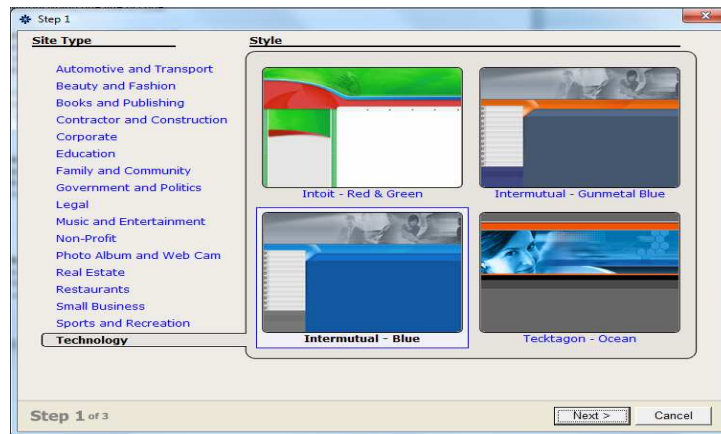


Itt kiválaszthatjuk, mit szeretnénk csinálni. Több lehetőség közül választhatunk:

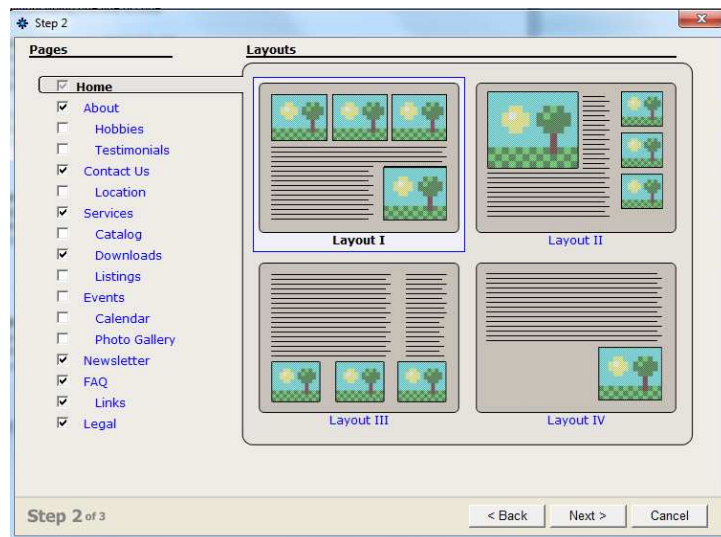
- Üres oldal létrehozása.
- Egy varázsló segítségével kiválasztom milyen oldalszerkezetet és stílust akarok.
- Megnyitok egy már meglévő templatet.

Én jelenleg egy új oldalt szeretnék létrehozni. Szeretném használni a beépített és előre formázott stílusokat is, ezért a „site wizard” menüpontot választom ki.

Első lépésben megkérdezi a program, hogy milyen beépített stílust szeretnék használni. A döntés ránk van bízva. Én most kiválasztok egy nekem tetszőt:

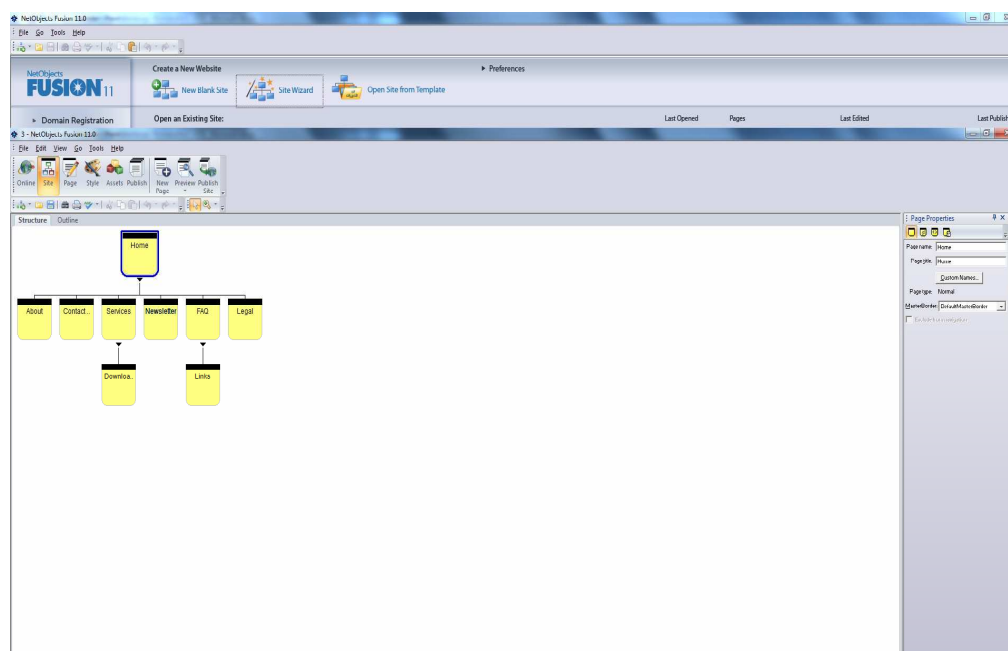


Következő lépésben el kell döntenünk milyen kezdeti weboldalszerkezetet szeretnénk. Mik szerepeljenek a weboldalon.



Válasszuk ki azokat, melyeket szeretnénk, majd lépünk tovább. Következő ablakon velünk kapcsolatos információkat kér be a program, de ezt nem muszáj kitölteni. Lépünk tovább. A website oldalunknak ezután adjunk nevet és mentjük el.

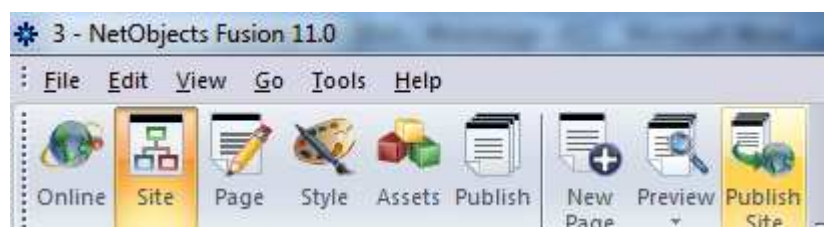
A program létrehozza a kezdeti beállításoknak megfelelő (stílus és struktúra) honlapunkat. A következő képernyő fog minket fogadni:



Láthatóvá válik ezen az oldalon grafikusán is az oldalunk felépítése.

Innentől kezdve teljesen szabadon választhatóak a lépések, amiket szeretnénk csinálni. Nem fogom lépésről lépésre bemutatni és leírni a weboldal létrehozását, mert a lépések tetszőleges sorrendben követhetik egymást és jelen szakdolgozat kereteit a lépésről – lépésre való bemutatás meghaladná. Úgy gondoltam, hogy bizonyos alapvető dolgokat fogok megmutatni mit és hol kell beállítani. A program teljes, minden menüpontra és elemre való ismertetése szintén nem célom.

A fentiek figyelembevételére után nézzük át a fontosabb menüpontokat:



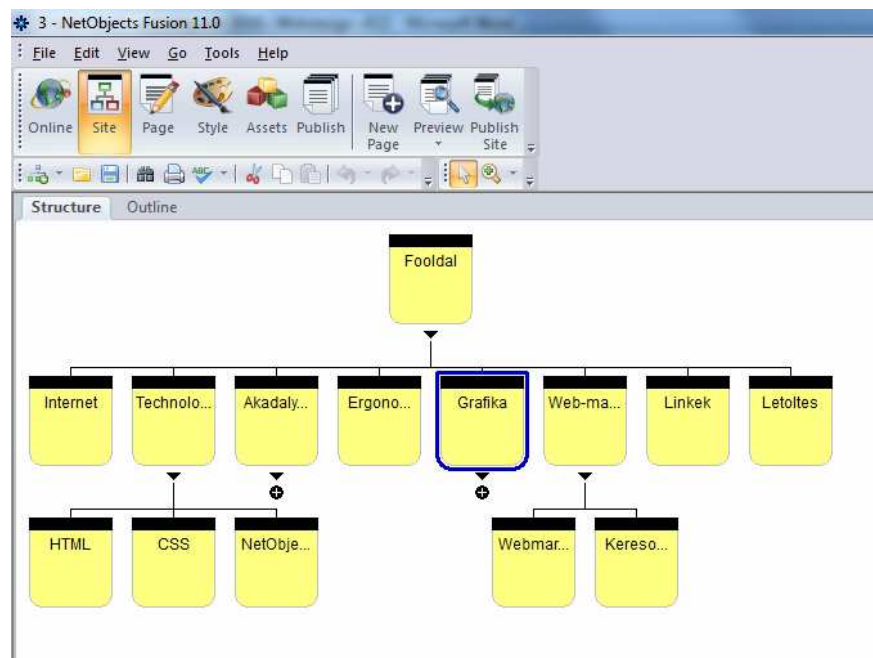
Az alábbi részben ezeket a menüpontokat fogom kicsit részletezni (teljesség igénye nélkül).

2.4.4 „Site” menüpont

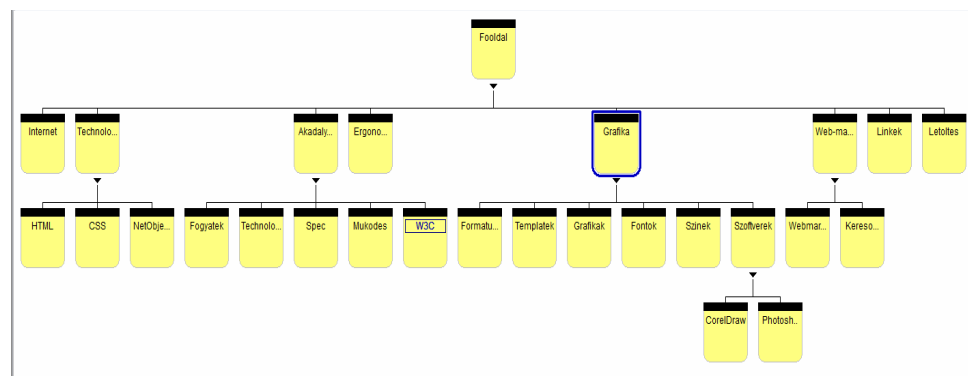
Ebben a menüben információt kaphatunk az oldalunk felépítéséről grafikusán. Szerkeszthetjük a honlapunk felépítését. Új lapokat adhatunk a honlapunkhoz. Lapokat törölhetünk. Oldalainkat nevezhetjük el illetve át.

2.4.4.1 Oldalunk felépítése:

Mint említettem a „Site” menü grafikusán megjeleníti a honlapunk felépítését. Ha nagyon bonyolult az oldalunk felépítése (sok oldalból áll), akkor nehézséget okozhat az áttekintése. Ennek megkönnyítése érdekében a programban van lehetőség az oldalaink kibontására és csoportba záráására is. Az alábbi képen erre mutatok példát. Példánkban a „Technologia” oldal teljesen ki van bontva, viszont a „Grafika” oldal alatti oldalak nem láthatóak. Ezt mutatja a + jel (kibontható).

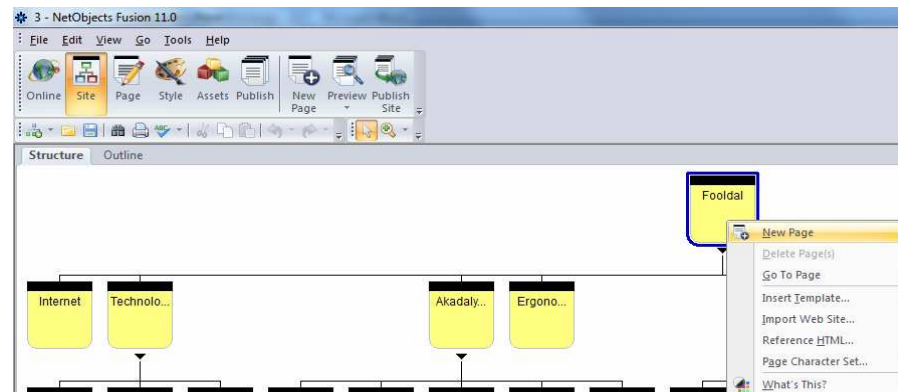


Példánkban, ha az oldalakat teljesen kibontanánk, akkor így alakulna a honlapunk felépítése:

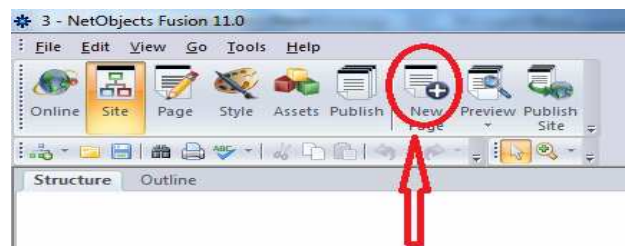


2.4.4.2 Oldalak létrehozása:

Egy adott oldalon, ha jobb egérgombbal kattintunk, akkor elérhetővé válik egy almenü (lásd az alábbi képen), amiből ha a „New Page” menüpontot választjuk, létrejön az új oldalunk, ami a kiválasztott oldal alá kerül.

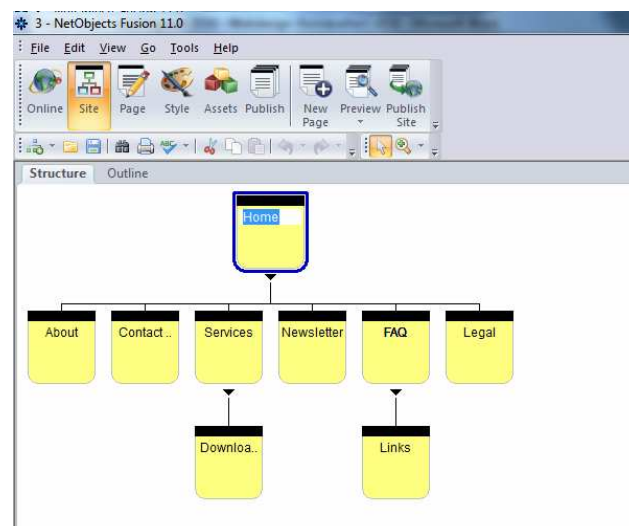


Nem csak az almenüből tudunk létrehozni új oldalt, hanem a jelzett helyről is:



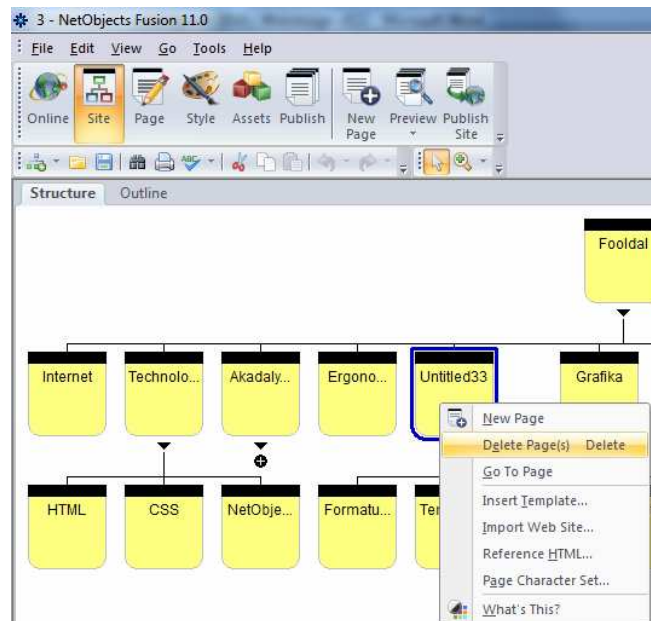
2.4.4.3 Oldalak átnevezése:

Átnevezhetjük az oldalainkat a Site menüben. Itt az oldal jelenlegi nevén egyet kattintva átírhatóvá válik az oldal neve.

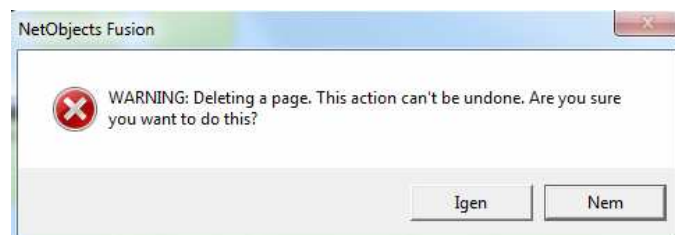


2.4.4.4 Oldalak törlése:

Az adott oldalon, ha jobb egérgombbal kattintunk, akkor elérhetővé válik a „delete page” menüpont.

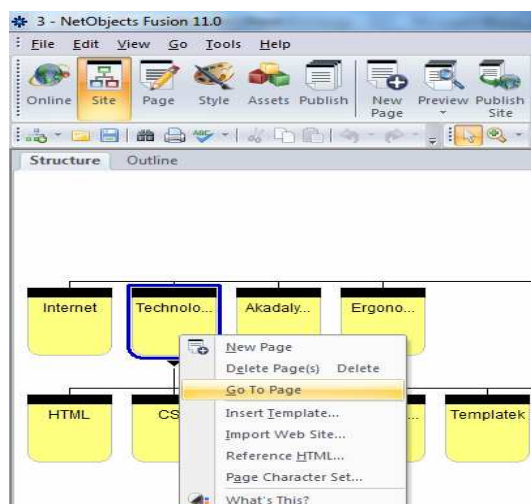


Ezt kiválasztva egy figyelmeztető ablak jelenik meg, amely közli, hogy az oldal törlésre kerül, és ezt elfogadjuk-e vagy sem.



2.4.4.5 Ugrás az oldalra:

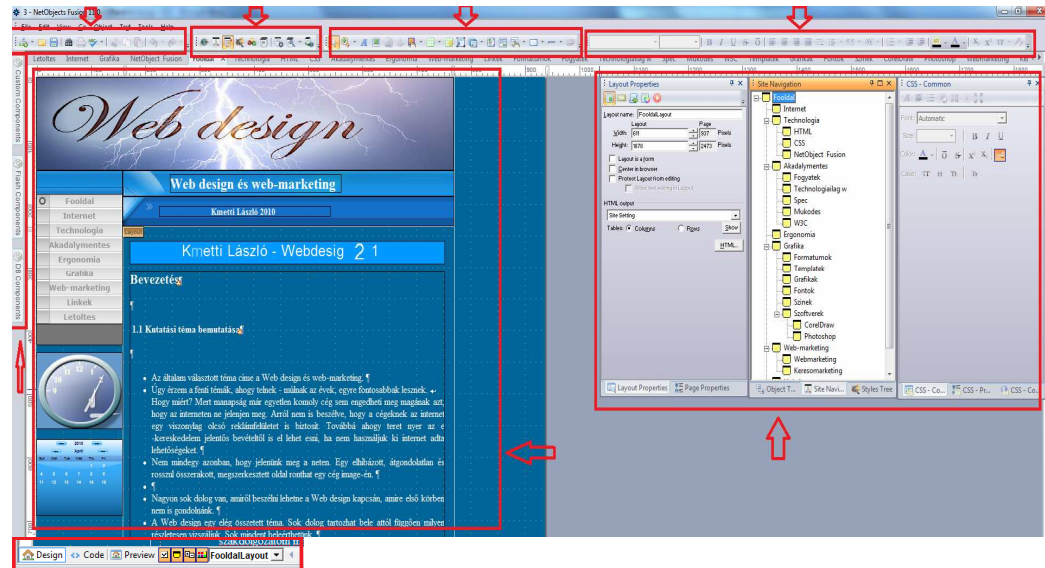
Egy adott oldalra ugorhatunk, ha rákattintunk a weboldalra és kiválasztjuk a „go to page” menüpontot.



2.4.5 „Page” menüpont

Itt lehet az oldallal kapcsolatos dolgokat beállítani.

Az alábbi képhez hasonló szerkezet fogad minket (szabadon átrendezhető).
A képen piros téglalapokba foglaltam a főbb csoportokat és piros nyílal jelöltem őket.



Képernyő tetején 4 csoport található:

Első csoportban az alábbi elemek érhetőek el:



- betöltés, - mentés, - nyomtatás, - nyelvvellenőrzés, - kivágás,
- másolás, - beillesztés, - művelet visszavonása,

Második csoportban az ismertetett főbb menüelemek érhetőek el:



- „Site”, - „Page”, - „Style”, - „Assets”, - „Perview”,
- „Publish Site”

Harmadik csoportban a „Page” főmenü almenü elemei érhetőek el:



Ezeket az elemeket pár oldallal alább részletesen ismertetem.
Lásd 2.4.5.5, 2.4.5.7, 2.4.5.8, 2.4.5.9, 2.4.5.10, 2.4.5.11 és 2.4.5.12 részekben.

Negyedik csoportban a szövegformázással kapcsolatos menüelemek érhetőek el:

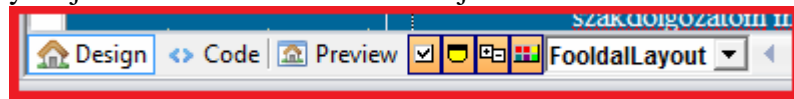


Részletesebben lásd a 2.4.5.6 részben.

Képernyő közepén 3 csoport található:

1. Baloldal legszélén, egy 3-as menücsoporthoz fogad, ami akkor jön elő, ha az egeret rávisszük. Ezek a következők:
 - „Custom components”
 - „DB components”
 - „Flash components”
2. Baloldalt található az a felület, ahol tudjuk az oldalunkat szerkeszteni.
3. Jobboldalon találhatóak pedig az aktuálisan elérhető menük, amivel finom hangolhatjuk az oldalon lévő elemeket. Ezek a menük is tetszőlegesen átrendezhetőek, elhelyezhetőek.

Képernyő alján a következő elemeket találjuk:



Itt 3 féle nézet között válthatunk:

- Tervező nézet
- Kód nézet
- Gyors oldalnézet

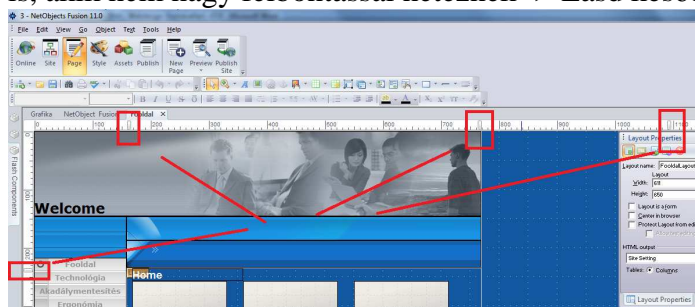
Hátrébb kapcsolókat találhatunk, amivel ki/be kapcsolhatjuk a képernyő jobb oldalán megjelenő menüket.

Részletesen az alábbiakban olvashatunk róla.

Lásd 2.4.5.13, 2.4.5.14 és 2.4.5.15-ös fejezetben.

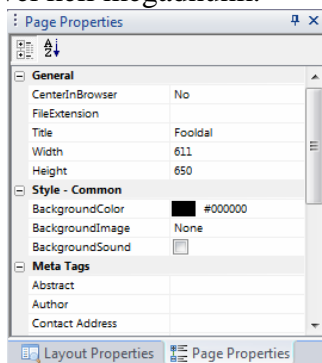
2.4.5.1 Keretek módosítása:

Célszerű ezt minél hamarabb beállítani. Itt adhatjuk meg (a jelzett területeken), milyen széles legyen a weboldalunk. A fejlécre és az oldalsávra mennyi helyet akarunk elkülöníteni. Célszerűen ne a legnagyobb értékeket vegyük figyelembe, mert gondolnunk kell azokra is, akik nem nagy felbontással neteznek -> Lásd később.



2.4.5.2 Hátter módosítása:

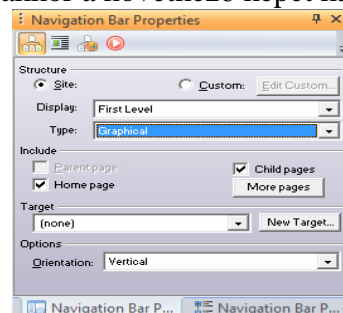
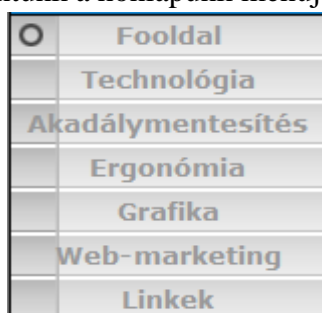
Ha rákattintunk a háttérre, majd a „Page Properties” menüben válasszuk ki a BackgroundColort. Itt tetszőleges érték felvehető, beállítható. Ezek a beállítások csak az adott oldalra szólnak. Ha azt szeretnénk, hogy az összes oldalon ilyen háttérünk legyen, akkor sajnos az összes oldalon egyesével kell megadnunk.



Mint látható háttérnek nem csak szín adható meg, hanem akár képet is berakhatunk.

2.4.5.3 A honlapunk menürendszer:

Amikor kezdetben kiválasztottunk egy stílust, akkor az adott stílushoz megfelelő menürendszert kaptunk. Ha ez nem tetszik, természetesen van lehetőség a cserére, vagy az adott menü finomhangolására is. Ha rákattintunk a honlapunk menüjére, akkor a következő képet kapjuk:



Megadható milyen szintek jelenjenek meg a menüben, vagy akár saját magunk is definiálhatjuk, mely oldalak jelenjenek meg a menüben. Megadható, hogy grafikus, szöveges vagy flash alapú legyen. Hogy helyezkedjen el (vízszintesen vagy függőlegesen). Megadható hogy a szöveget hova igazítsa.

Mostanra az oldalunkon már elnyerte az alap felépítését. Kialakult az oldal felépítése, váza. Beállítottuk a menürendszert, a háttérteret. Megadtuk, hogy milyen széles az oldalunk, mekkora a hasznos munkaterület.

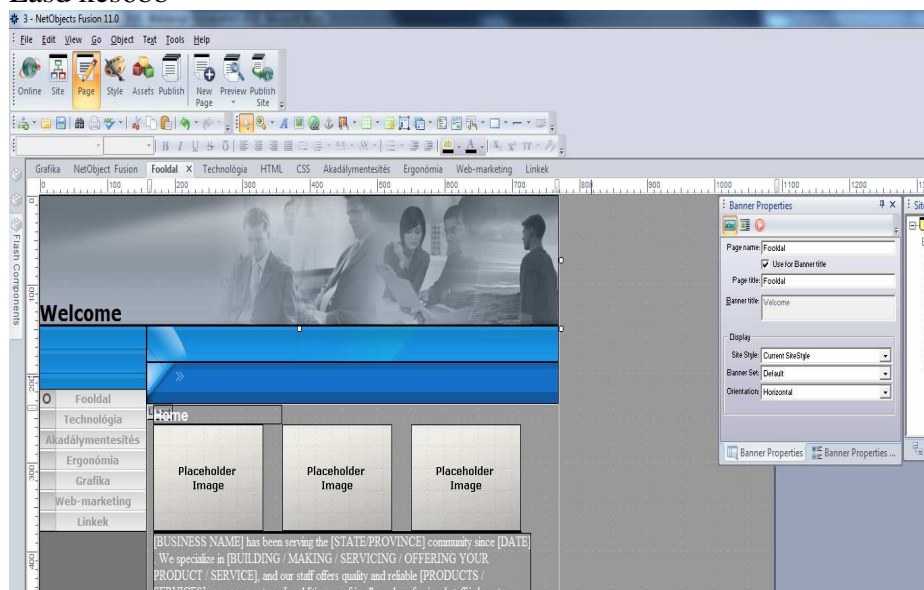
Jöhet az oldal tartalommal való feltöltése /képek és szövegek/.

2.4.5.4 Banner megadása:

Az oldalunkat egyedivé szeretnénk tenni. Amikor a stílust kiválasztottuk akkor kaptunk egy kezdeti bannert. Ez természetesen nem felel meg a jelenlegi igényeknek.

(Egyszerűen törölhető egy 'DELETE' gomb megnyomásával.)

Ha szeretnénk lecserélni sajátra, akkor létre kellene hozni saját bannerünket. Ennek létrehozására ajánlom a Photoshop programot -> Lásd később



Én a következőt terveztem most hozzá:



Megjegyzés:

A tervezés során felhasznált forrást csatolom a szakdolgozatomhoz.

A következőkben sorra veszem a fontosabb elemeket, amiket az oldalunkra beszúrhatunk. Ezen a menüsoron fogok végigmenni:



Az első két elem a kiválasztás és a kicsinyítés és nagyítás. Ezeket nem részletezném.

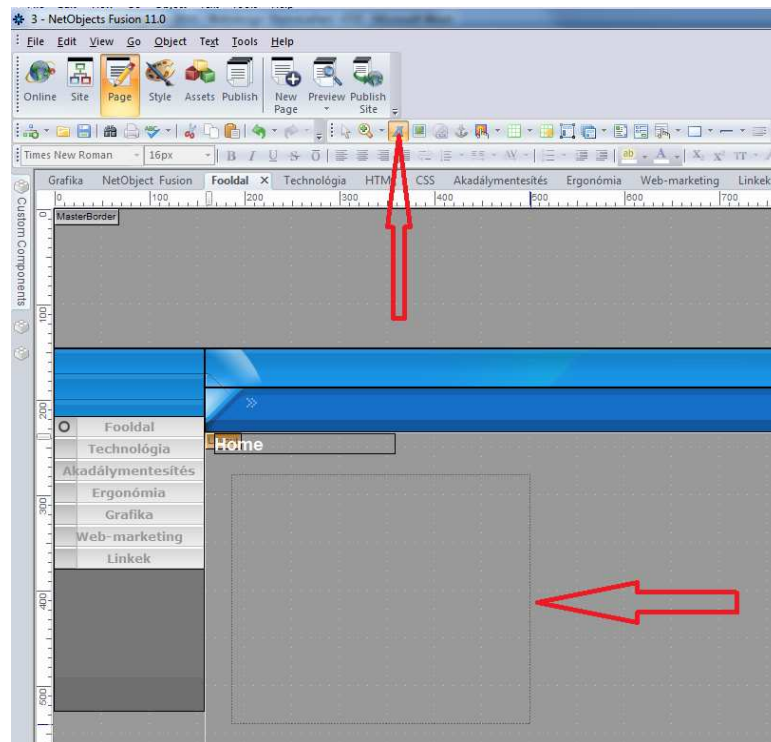
2.4.5.5 Szöveg beviteli mező:

Válasszuk a felfele mutató nyílnál elhelyezkedő „A” gombot. Erre, ha rákattintunk, akkor kijelölhetjük, hogy hova akarunk szöveges beviteli mezőt tenni (ezt jelöltem az alsó piros nyíl segítségével).

Kinagyítva az érintett terület:



^



2.4.5.6 Szöveg formázása:

Hasonlóképpen történik, mint ahogy a Word-be megszokhattuk.
Szöveg formázására szolgáló eszközsor:



Itt is megtalálhatóak a szokásos formázási lehetőségek.

Betűtípus megadása. Méret megadása. Félkövér, dőlt, aláhúzott betűk.

Balra, középre, jobbra zárás és sorkizárás. Továbbá szöveg színének és háttérszínének megadása.

2.4.5.7 Kép beszúrása:

Válasszuk az alábbi képen látható ikonok közül baloldaltól számolva a negyedik ikont.

Ezután szintén válasszuk ki hova akarjuk a képet betenni (jelöljük ki).

Következő lépésben adjuk meg a kép elérési útvonalát.

Kép a programból:



2.4.5.8 Linkek:

Többféle link megadható a programban (link type mezőben).

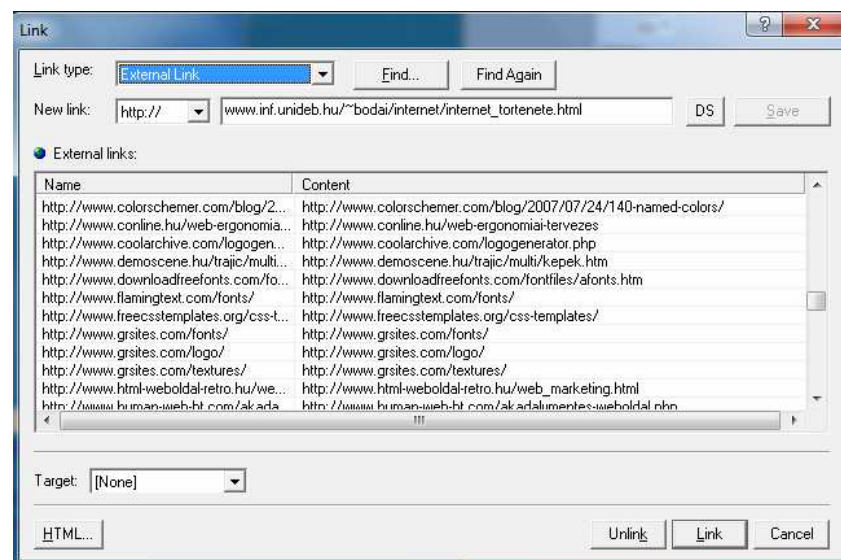
Például külső, belső és fájl link.

Az alábbi képen jelölöm melyik elemmel érhető el ez a funkció.

Kép a programból



A link panel felépítése:



2.4.5.9 Táblázat:

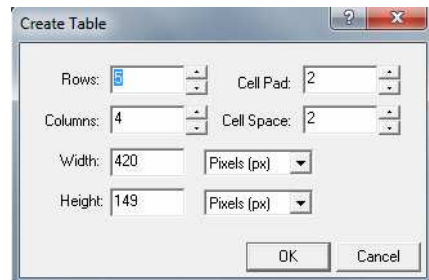
Az eszköztár alábbi elemére kattintva megadható hol legyen a táblázat. Mennyi oszlopa és mennyi sora legyen többek között.

Az alábbi képen jelölöm melyik elemmel érhető el ez a funkció.

Kép a programból



^

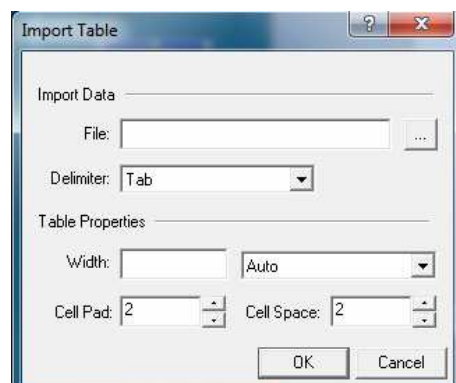


2.4.5.10 Táblázat importálása:

Nem csak új táblázatot hozhatunk létre, hanem lehetőségünk van táblázatok importálására is. Ezt a következő menüvel érhetjük el:



^



2.4.5.11 Form elem:

Megnyomása után kijelölhető hova akarjuk tenni az űrlapunkat.

Az alábbi képen jelölöm melyik elemmel érhető el ez a funkció.

Kép a programból:



^

Form kialakítására szolgáló menü:



2.4.5.12 Elválasztó elem:

Szövegrészek elválasztásánál jöhet jól.

Az alábbi képen jelölöm melyik elemmel érhető el ez a funkció.

Kép a programból:



^

Most még szót ejtenék a 3 fajta nézetről, amit az 57. oldalon már megemlítettem.

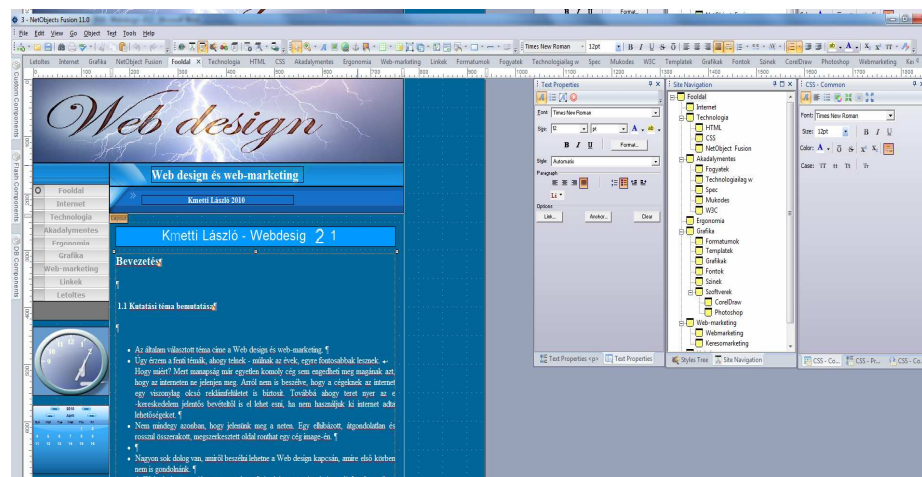
Ezek a következők voltak:

- Tervező nézet
- Kód nézet
- Gyors oldalnézet

2.4.5.13 Tervező nézet:

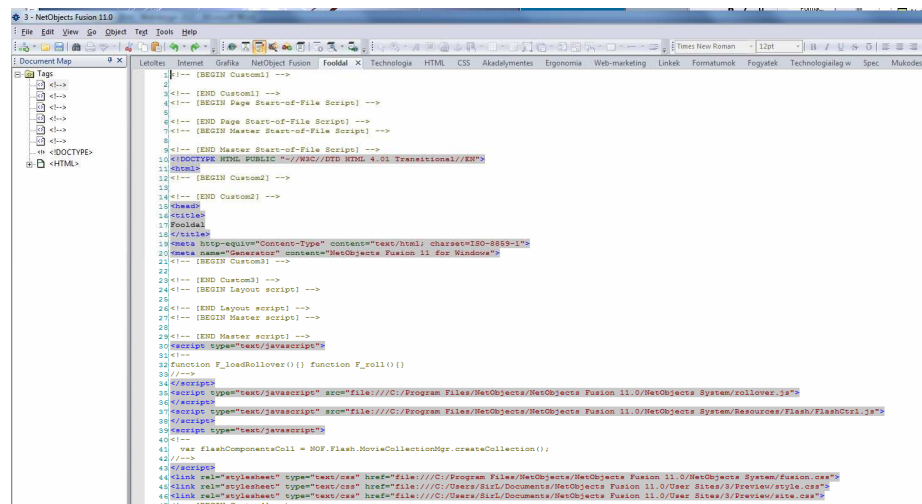
Alapértelmezés szerint ebben a nézetben tartózkodunk.

Itt érvényesülnek a WYSIWIG elvek. Megadjuk, hogy melyik elem hova kerüljön, hogy nézzen ki. Tervező nézet ablaka:



2.4.5.14 Kód nézet:

Ebben a nézetben szerkeszthetjük (finomíthatjuk) a forráskódunkat.

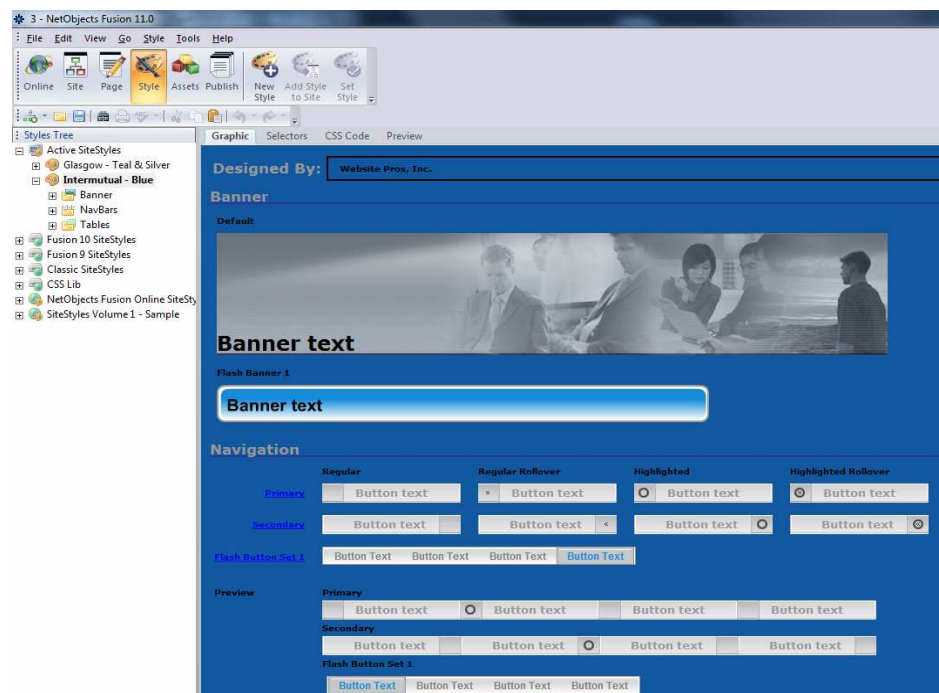


2.4.5.15 Gyors oldalnézet:

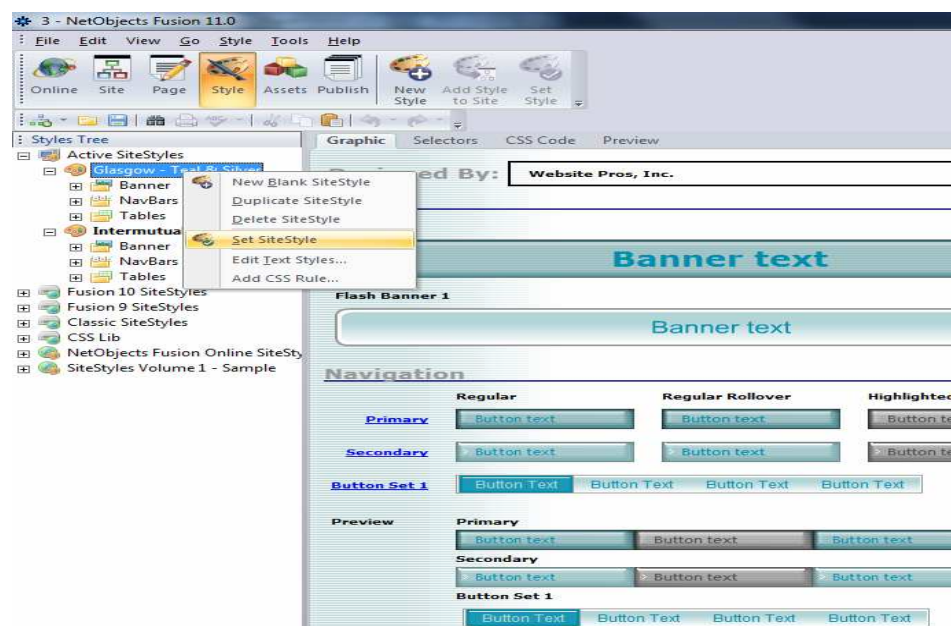
Ebben a nézetben megtekinthetjük, hogy nézne ki az oldalunk a böngészőben.

2.4.6 „Style” menüpont

Ha szeretnénk változtatni a kezdetkor megadott stíluson, akkor ebben a menüpontban van rá lehetőségünk. Itt választhatunk a program beépített stílusai közül („Style Tree”). Több beépített stílus közül választhatunk a programban (Fusion 9, Fusion 10, Classic SiteStyles). A program bal oldalán érhetőek el a beépített stílusok. A képernyő jobb oldalán láthatjuk az adott stílus kinézetét:



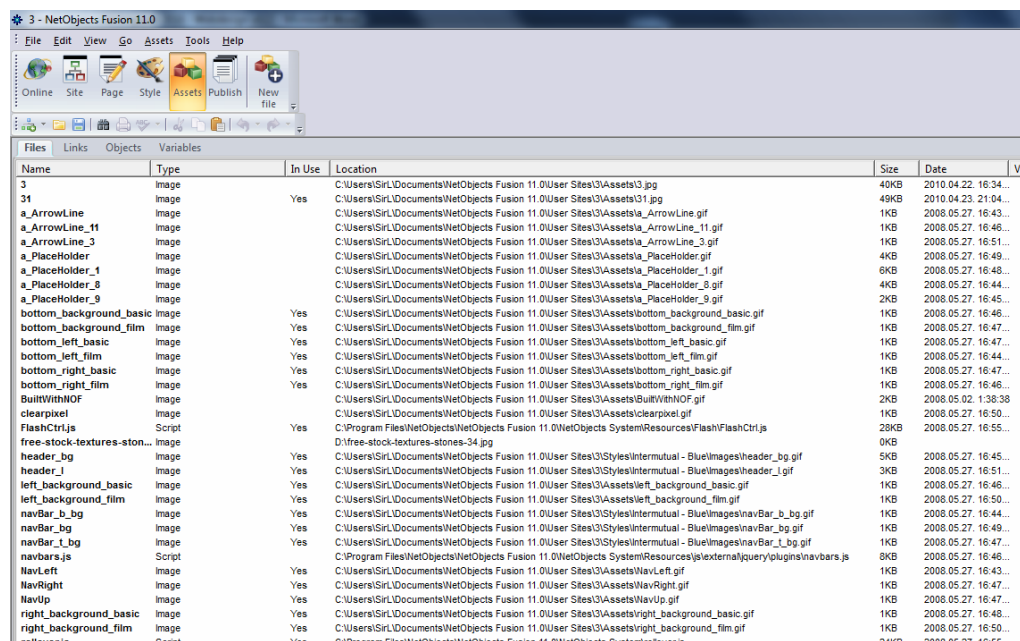
Másik stílus alapértelmezettnek állítása:



Ha a beépített stílusok közül egyik sem felel meg számunkra, akkor saját magunk is definiálhatunk saját stílust.

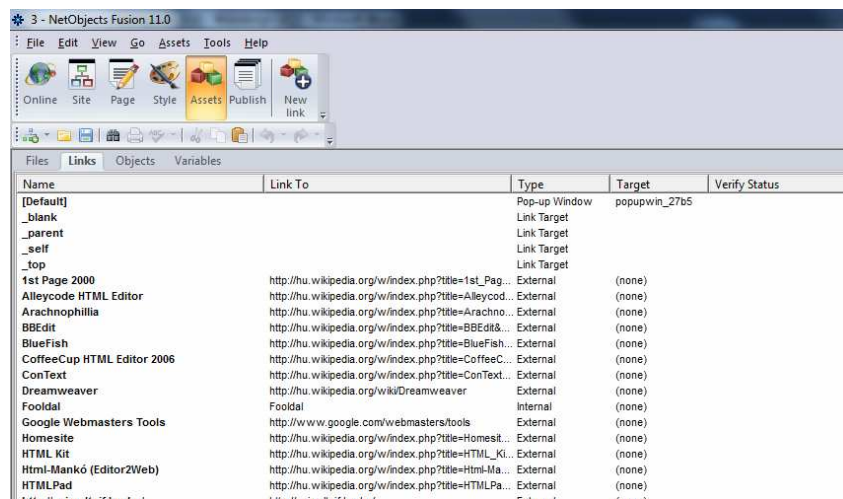
2.4.7 „Assets” menüpont

Ebben a menüben megnézhetjük, hogy a honlapunk milyen file-okból áll jelenleg. Ezek a fájlok hol találhatóak, mekkora a méretük és mikor lettek létrehozva („files” fül).



Name	Type	In Use	Location	Size	Date	Ve
3	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\3.jpg	40KB	2010.04.22. 16:34...	
31	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\31.jpg	49KB	2010.04.23. 21:04...	
a_ArrowLine	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_ArrowLine.gif	1KB	2008.05.27. 16:43...	
a_ArrowLine_11	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_ArrowLine_11.gif	1KB	2008.05.27. 16:48...	
a_ArrowLine_3	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_ArrowLine_3.gif	1KB	2008.05.27. 16:51...	
a_PlaceHolder	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_PlaceHolder.gif	4KB	2008.05.27. 16:49...	
a_PlaceHolder_1	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_PlaceHolder_1.gif	6KB	2008.05.27. 16:48...	
a_PlaceHolder_8	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_PlaceHolder_8.gif	4KB	2008.05.27. 16:44...	
a_PlaceHolder_9	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\A_PlaceHolder_9.gif	2KB	2008.05.27. 16:45...	
bottom_background_basic	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_background_basic.gif	1KB	2008.05.27. 16:46...	
bottom_background_film	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_background_film.gif	1KB	2008.05.27. 16:47...	
bottom_left_basic	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_left_basic.gif	1KB	2008.05.27. 16:47...	
bottom_left_film	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_left_film.gif	1KB	2008.05.27. 16:44...	
bottom_right_basic	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_right_basic.gif	1KB	2008.05.27. 16:47...	
bottom_right_film	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\bottom_right_film.gif	1KB	2008.05.27. 16:46...	
BuiltWithNOF	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\BuiltWithNOF.gif	2KB	2008.05.02. 1:38:38	
clearpixel	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\clearpixel.gif	1KB	2008.05.27. 16:50...	
FlashCtrl.js	Script	Yes	C:\Program Files\NetObjects\NetObjects Fusion 11.0\NetObjects\System\Resources\Fusion\FlashCtrl.js	28KB	2008.05.27. 16:55...	
free-stock-textures-ston...	Image	Yes	D:\free-stock-textures-stones-34.jpg	0KB		
header_bg	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\header_bg.gif	5KB	2008.05.27. 16:45...	
header_l	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\header_l.gif	3KB	2008.05.27. 16:51...	
left_background_basic	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\left_background_basic.gif	1KB	2008.05.27. 16:46...	
left_background_film	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\left_background_film.gif	1KB	2008.05.27. 16:50...	
navBar_b_bg	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\navBar_b_bg.gif	1KB	2008.05.27. 16:44...	
navBar_bg	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\navBar_bg.gif	1KB	2008.05.27. 16:49...	
navBar_l_bg	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\navBar_l_bg.gif	1KB	2008.05.27. 16:47...	
navbars.js	Script	Yes	C:\Program Files\NetObjects\NetObjects Fusion 11.0\NetObjects\System\Resources\external\plugins\navbars.js	8KB	2008.05.27. 16:46...	
NavLeft	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\NavLeft.gif	1KB	2008.05.27. 16:43...	
NavRight	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\NavRight.gif	1KB	2008.05.27. 16:47...	
NavUp	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\NavUp.gif	1KB	2008.05.27. 16:47...	
right_background_basic	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\right_background_basic.gif	1KB	2008.05.27. 16:48...	
right_background_film	Image	Yes	C:\Users\SirL\Documents\NetObjects Fusion 11.0\User Sites\3\Assets\right_background_film.gif	1KB	2008.05.27. 16:50...	
runtime.js	Script	Yes	C:\Program Files\NetObjects\NetObjects Fusion 11.0\NetObjects\System\Resources\external\plugins\runtime.js	24KB	2008.05.27. 16:55...	

Milyen linkeket használunk az oldalunkon. Ezek a linkek milyen oldalra mutatnak és milyen típusúak („links” fül):



Name	Link To	Type	Target	Verify Status
[Default]		Pop-up Window	popupwin_27b5	
_blank		Link Target		
_parent		Link Target		
_self		Link Target		
_top		Link Target		
1st Page 2000	http://hu.wikipedia.org/w/index.php?title=1st_Page...	External	(none)	
Alleycode HTML Editor	http://hu.wikipedia.org/w/index.php?title=Alleycode...	External	(none)	
Arachnophilia	http://hu.wikipedia.org/w/index.php?title=Arachnophi...	External	(none)	
BBEdit	http://hu.wikipedia.org/w/index.php?title=BBEdit&...	External	(none)	
BlueFish	http://hu.wikipedia.org/w/index.php?title=BlueFish...	External	(none)	
CoffeeCup HTML Editor 2006	http://hu.wikipedia.org/w/index.php?title=CoffeeC...	External	(none)	
ConText	http://hu.wikipedia.org/w/index.php?title=ConText...	External	(none)	
Dreamweaver	http://hu.wikipedia.org/wiki/Dreamweaver	External	(none)	
Fooldal	Fooldal	Internal	(none)	
Google Webmasters Tools	http://www.google.com/webmasters/tools	External	(none)	
Homesite	http://hu.wikipedia.org/w/index.php?title=Homesite...	External	(none)	
HTML Kit	http://hu.wikipedia.org/w/index.php?title=HTML_Ki...	External	(none)	
Html-Mankó (Editor2Web)	http://hu.wikipedia.org/w/index.php?title=Html-Ma...	External	(none)	
HTMLPad	http://hu.wikipedia.org/w/index.php?title=HTMLPa...	External	(none)	
Internet Explorer	http://www.microsoft.com/ie/	External	(none)	

Továbbá megnézhetjük az oldalunkon használt objektumokat („objects” fül) és a változóinkat is („variables” fül).

2.4.9 „Preview” menüpont

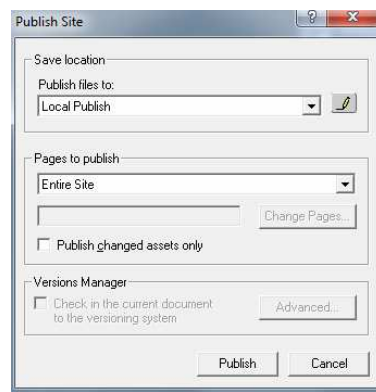
Ha le szeretnénk bármikor ellenőrizni, hogy néz ki az oldalunk, akkor használhatjuk ezt a menüpontot. Hatására a böngészőnkben megjelenik a szerkesztett oldalunk. Megnézhetjük, hogy a felhasznált elemek mind a helyén vannak-e. Ezzel az opcióval még nem jön létre a forráskód, arra a „Publish Site” menüpontot kell majd használnunk.

2.4.10 „Publish Site” menüpont

Megadhatjuk, hol szeretnénk létrehozni a honlapunkat. Publikálhatjuk helyileg is (saját merevlemez), vagy rögtön az internetre is feltölthető.

Megadható továbbá az is, hogy a teljes tartalmat, vagy csak egy részét szeretnénk publikálni. Publikálás hatására elkészül a tényleges forráskód.

Kép a programból:



Ezzel főbb menük ismertetésének a végére értem. A függelék részbe beteszek még pár fontosabb kimaradt elemről képet. Itt gondolok például a beépített flash animációkra, vendégkönyvre, fotó galériákra, valamint hang és videó beszúrására.

Most viszont folytatnám egy új témával az akadálymentesítés és ergonómiával.

3. Akadálymentesítés és Ergonómia

3.1 Bevezetés

Az előző fejezetben röviden végigvettem a HTML és a CSS alapjait. Továbbá mutattam egy programot, ahol ezeket a megszerzett ismereteket fel tudjuk használni.

Ebben a fejezetben most más oldalról közelítem meg a témát. Nem a technológiai és szoftveres oldalról, hanem a felhasználók felől.

Az akadálymentesítés részben szó lesz arról, hogy a különböző hátrányos helyzetű csoportok számára mik a kívánalmak. Milyen tanácsokat lehet adni, hogy ők is tudják az általunk létrehozott oldalt használni. Ehhez a témához fog még kapcsolódni egy ide vonatkozó szabvány a WCAG 1.0 és a frissebb 2.0-ás verzió. Ezeket is röviden ismertetem.

A fejezet hátralévő részében azt ismertetem mitől lesz egy weblap hatékony. Mitől nő az oldalon a felhasználó komfortérzete. Szó lesz továbbá arról is, hogy a felhasználók az oldal mely részét nézik meg és milyen ergonómiai hibák lehetnek. Ezt az utolsó részt szó szerint fogom majd idézni a megadott weboldaltól.

Akkor kezdjük az elején, mégpedig az akadálymentesítés témakörnél.

3.2 Akadálymentesítés

A világhálót sokféle ember böngészi, akik különböző adottságokkal rendelkeznek.^[1] Nem csak ép emberek, hanem azok is böngésznek, akik valamilyen fogyatékkal kénytelenek élni. Gondolok itt többek között a látás, hallás, testi vagy szellemi fogyatékkal élőkre. Az elmúlt évekig viszonylag keveset lehetett hallani az akadálymentesítésről, de nonprofit szervezetek és magánszemélyek erőfeszítéseinek köszönhetően azonban egyre inkább figyelmet kap, mind a magánszektorban, mind pedig a kormányzati szférában az akadálymentesítés.

Nem szoktunk azon elgondolkozni, hogy mennyire mások vagyunk. Lehet, ami az egyiknek jó, az a másiknak nem. Lehet, hogy amit én átlátok, azt a másik nem fogja. Lehet, hogy ami nekem gyorsan lejön, az a másiknak csak lassan fog, mert lassabb kapcsolattal rendelkezik, vagy éppen a számítógépe lassabb. Lehet, hogy ami egy 4:3-as hagyományos CRT monitoron 800x600-as felbontás mellett jól néz ki, az egy 16:9-es oldalarányú 1920x1080-as felbontású monitoron nem fog jól mutatni.

Általános elv, hogy a weboldalainkat minden felhasználó ugyanúgy használhassa. Egy akadálymentes website létrehozása tehát mindössze annyiból állna, hogy a hozzáférhetőséget korlátozó tényezőket megszüntetjük. Egy akadálymentes honlappal viszont nyerhetünk is.

Mik az akadálymentes website pozitív hatásai?

- Az akadálymentes weboldalak — strukturált felépítésüknek köszönhetően — a webes keresőkben jobb helyezést érnek el. A Googlebot az internet legbefolyásosabbak felhasználója.
- Pozitív PR — a website tulajdonosa hangoztathatja társadalmi felelősségvállalását és a szabványoknak való megfelelését.
- A website tulajdonosa olyan piaci csoportokat érhet el, melyek még kevésbé telítettek és így növelheti a website látogatottságát.

Mik lehetnek egy nem-akadálymentes website negatív hatásai?

- Jelenleg még Magyarországon nem fordult elő, hogy emiatt akár egy céget is bepereltek volna, ám a hatályos törvények szerint tilos bármilyen hátrányos megkülönböztetés. A magyar közigazgatási honlapok egységesítését célzó KIETB ajánlás előírja a kormányzati honlapok akadálymentessé tételét.
- Potenciális ügyfelek elvesztése. Másodrangú polgárként kezelt, panaszos ügyfelek. A felhasználók általában pozitív tapasztalataikat ritkábban osztják meg egymással, mint a negatívakat. Kiszámú elégedetlen ügyfél véleménye is negatív PR. [7]

Egy akadálymentesített honlap létrehozása esetén nem csak a fogyatékkal élőket kell figyelembe venni, hanem az egyéb, valamilyen szempontból különleges igényű csoportokat is. Nézzük ezeket a csoportokat, vegyük sorra őket.

3.2.1 Fogyatékkal élők

Akik testi vagy lelki fogyatékkal élnek nagy csoportját alkotják az akadálymentesített honlapok felhasználóinak, látogatóinak. Ők jobban igénylik, hogy otthonról intézhessék az ügyeiket, mint azok, akik ép embernek. Például egy mozgáskorlátozott embernek gondot jelenthet bemenni egy hivatalba, míg egy ép mozgású ezt jobban megteheti.

3.2.1.1 Vakok

Legtöbben azt gondolják, hogy az akadálymentesített honlapok egyetlen célcsoportja a vakokból áll. Ez az állítás így nem állja meg a helyét, de igaz, hogy a vakok és gyengénlátók igen nagy felhasználói csoport. Az akadálymentes honlap nem egyenlő a vakbarát honlappal. Ahhoz, hogy a vakok is igénybe tudják venni a web kínálta lehetőségeket képernyőolvasóra, vagy Braille kijelzőre van szükség^[5], mert a vakok alapvetően kétféle módon képesek az írott szöveg befogadására:

- hallás által
- tapintás segítségével

A vakoknak a színek és képek nem sokat jelentenek. A beszélőprogramjuk nincsen felkészítve arra, hogy ezeket az anyagokat megfelelően el tudják mondani. Ahhoz hogy a felolvasó programjuk értelmezni tudja az oldalakat, elérhetővé kell tenni az oldalakat szöveges formátumban is. Vakbarát oldalakra példa:

<http://latasserult.budapest.hu/Engine.aspx>

<http://nfgm.gov.hu/akadalymentes>

<http://www.vk-pecs.bibl.hu/vak/Index.php>

https://www.otpbank.hu/portal/A_index.jsp

<http://vakbarat.whyssoft.hu/>

<http://fusz.hu/vakbarat/>

Mitől lehet egy oldal vakbarát?

Törekedjünk kizárólag a szöveges információra, ahol csak lehet. Ne a megjelenés, a minél színesebb küllem legyen a fő szempont, hanem törekedjünk olyan formátumban közzétenni az oldalunkat, hogy a vakok is könnyen tudják használni. A honlap szerkesztésénél ügyelni kell arra, hogy a látássérült csak azt az információt kapja meg első körben, ami alapján el tudja dönteni,

hogyan az számára éppen szükséges-e, vagy nem. Kerülnünk kell a frame-ek, animációk és a flash-ek használatát. Ha több menüpontot szeretnénk elhelyezni egymás mellett, azt rakjuk olyan táblázatba, amely a karakteres böngészővel úgy jelenik meg, hogy az egymás melletti mezők egymás alá kerülnek.

3.2.1.2 Gyengénlátók

A gyengénlátók igényei nem teljesen azonosak a vakokéval, hiszen nagy, jól szerkesztett ábrákat képesek lehetnek látni, sőt segít is nekik, mint mindenki másnak is, a szöveg megértésében. Azok, akik nem használnak felolvasóprogramot, igénylik, hogy a betűtípust illetve a kontrasztot állíthassák az oldalon. A betűméret (akár drasztikus méretű) megnövelésére van a legnagyobb igény körükben. Erre a böngészőkben elhelyezett betűméret változtatásra alkalmas részt, képernyő nagyítót használnak, de előfordul, hogy speciális megjelenítő eszközöket (nagy méretű kijelzőket, kivetítőket stb.) is alkalmaznak. A számukra elérhetővé tett oldalak előnyeit a kisméretű monitort használó látogatóink is élvezik.^[6]

3.2.1.3 Színvakok és hallássérültek

Fontos alapszabály akadálymentesített honlap készítésekor, hogy szín és hang ne hordozzon fontos információt. A szín nem csak a (szín)vakokat akadályozhatja a megértésben, hanem a színtévesztőket, régi monokróm monitort, esetleg modern, de kevés színnel rendelkező mobiltelefonos böngészőt használókat is. Hang esetében a hangkártyával illetve hangszóróval fel nem szerelt, például irodai gépekre is gondolni kell, mert azok a felhasználók is funkcionális hallássérültek tekinthetők. Ha az oldalaink fekete-fehér megjelenítő eszközön jól látható és jól használható, akkor jó eséllyel a fenti csoport tagjai is elérik azok tartalmát. Kritikus pontok, tanácsok:

- A szöveg kiemelkedik-e a háttérből, jól olvasható-e?

Szöveg nélkül alkalmazzunk nagy kontrasztot: minél nagyobb az eltérés a háttér és betűszín között, annál jobb.

- A szöveg háttérül lehetőleg színt használjunk, a mintát kerüljük, mert fárasztó és zavaró lehet!

- Vannak színekombinációk, amik fárasztóak a szemnek és, arról nem is beszélve, hogy egy színvak nem fogja látni a szöveget. Egyik ilyen kombináció például a vörös + zöld, amit egy színvak nem fog tudni megkülönböztetni.⁸

- A linkek színeként, amennyiben lehetséges, ragaszkodjunk a megszokott kék-lila színekombinációhoz, s az aláhúzás formátumhoz. Más szövegrész kiemelése az aláhúzással nem ajánlott, nem csak a honlapon, minden más elektronikus és papíralapú dokumentumban is!

3.2.1.4 Mozgássérültek

Ebbe a csoportba nem csak azok a felhasználók tartoznak, akikre első körben gondolnánk (hiányzik valamilyen végtagjuk, egyáltalán nem képesek mozogni vagy csak részben), hanem azokat is ide lehet sorolni, akik valamilyen alternatív kezelőfelülettel kezelik a számítógépüket, mert valami akadályozza őket.⁹ Gondolok itt arra, hogy például nem rendelkeznek egérrel, csak billentyűzettel vagy fordítva. Nagyon nagy segítség egy mozgássérült számára, de mindenki másnak is hasznos, ha nem kell sokat gépelni. Az egyszer a regisztrációnál már megadott email-címet például a belépésnél nem kell újra kitölteni, vagy esetleg nem kell minden alkalommal begépelni a jelszavát.

A mozgássérültek sérülései igen különbözőek lehetnek, ebből következik, hogy az általuk használt segítő technikák, és módszerek is különbözőek lehetnek, az eltérő igényeknek megfelelően.

Általánosságban elmondható, hogy a derékon aluli sérülések nem befolyásolják a számítógép használatot. A főbb gondot a kézsérülések, a kézhiány, a mozgási nehézségek és a mozgásképtelenség okozza.

- Kézhiány, vagy teljesen használhatatlan kéz esetén alkalmazott módszerek

Költségtakarékos és egyszerűen használható voltak miatt a szájba vehető, vagy fejre erősíthető pálcák (amellyel a billentyűzet kezelését lehet megoldani) használata a legelterjedtebb.

Vannak olyan eszközök is, amelyek segítségével lehetővé válik az egérhasználat kéz igénybevétele nélkül, ilyen például a szem mozgását figyelő, szoftverrel összehangolt kamera, vagy a fejre erősíthető fejmozgást követő kurzorkezelő szerkezet. Van lábbal használható egér is, amelynek egyik fele a kattintást, másik a pozicionálást segíti.

- Remegés, túlmozgás esetén alkalmazott módszerek

Főleg billentyűzetre szerelt kéztámaszokat, vagy speciális billentyűzeteket alkalmazva kiküszöbölhetőek a félregépelések, azonban a gépelés még ezekkel a segédeszközökkel is lényegesen lassabban lehetséges.

- Néhány szoftveres segítő módszer

Az egér kezelése lehetséges billentyűzetről és fordítva is igaz, ha a képernyő billentyűzetekre gondolunk. Többször szükség lehet a leragadó billentyűszolgáltatásra, amely segítséget jelenthet a billentyűkombinációk leütésében, akár egy ujjat használva is.

Ezek a lehetőségek többnyire az összes asztali operációs rendszerben megvannak.

- Egyéb eszközök

Van olyan szájjal fújva és szívva kezelhető cső, amely alkalmas az egér jobb és bal gombjának megfelelő működést szimulálni. Egyes speciális esetekben az agy elektromos hullámaint érzékelő pántot helyeznek a webet komoly mozgásproblémákkal megközelítő embertársaink fejére, amelyhez egy az egyszerű eldöntendő kérdések megválaszolásán alapuló szoftver kapcsolódik.

Ezzel az eszközzel – bár meglehetősen lomha tempóban – lehetővé válik a böngészés teljesen mozgásképtelen emberek számára is.

Előfordul, hogy mozgássérült embertársunk kizárólag a fejét képes mozgatni. Ilyenkor például a szájjal kezelhető botkormány (joystick) lehet segítségére egy képernyőre kirajzolt, szoftveresen kezelhető billentyűzettel kombinálva.

Számtalan különleges egér, nyomógomb, botkormány és billentyűzet létezik, amelyek felsorolására nehéz volna vállalkozni, hiszen sokszor ezeket az eszközöket egyedileg, személyre szabottan gyártják.

3.2.1.5 Értelmileg visszamaradottak

Komoly kihívást jelent egy olyan átlátható és könnyedén használható felhasználói felület kialakítása, amelyet értelmi fogyatékosok is tudnak használni. Nekik a szövegértés okoz komoly gondot. Hogy segíthetünk nekik?

- Oldalunk legyen átlátható és egyértelmű.

Például ne írjunk hosszú egybefüggő szöveget. Alakítsuk úgy ki az oldalt, hogy lehetőség szerint ne kelljen görgetnie a felhasználónak.

Az így kialakított oldal mindenkinek előnyös, de a fogyatékosoknak elengedhetetlen. Erre mindig törekedni kell.

- Felhasználó lépésről lépésre való vezetése.

Célunk, hogy mindig értse és tudja a felhasználó, hogy pontosan hol tart az adott folyamatban. Míg ez megkönnyíti egyes emberek dolgát, másokat idegesíthet, ezért érdemes ezeket a szolgáltatásokat kikapcsolhatóvá tenni.

- Segítsük őket ábrák használatával.

- Hagyjunk időt nekik, hogy a kapott információkat feldolgozzák.

3.2.1.6 Diszlexiások (és diszgráfiások)

Számukra a hosszú egybefüggő szövegek jelentik a fő akadályt, nemegyszer használnak felolvasóprogramot, hogy ezzel is megkönnyítsék a böngészést. Számukra segítség lehet, ha rövidebb részekre tagoljuk a szövegeket például alcímek alkalmazásával.

A diszlexia enyhébb változatának tekinthetjük az olvasási nehézségekkel küzdést. Gépelési nehézséggel küszködőknek nagy segítség lehet a zsebtelefonokon már elterjedt prediktív szövegbevitel, amely a nyelvnek megfelelően segíti a felhasználót.

3.2.1.7 (Fényérzékeny) epilepsziában szenvedők

Általában gond nélkül tudják az oldalakat böngészni, ám bizonyos minták, vagy villogó elemek, képterületek görcsös rohamot válthatnak ki belőlük.

Többnyire csak az animációkat tiltják le a böngészőjükben.

Tanács: Kerüljük a gyorsan villogó elemeket az oldalunkon.

3.2.1.8 Kulturálisan elszigetelt emberek

Náluk (többnyire) nem találhatóak segítő technológiák, mivel nem is ismerik azokat. Legnagyobb problémájuk, hogy nem nagyon ismerik a weboldalak által használt alapvető navigációs eszközök használatát sem eléggé, ezért számukra különösen fontos, hogy egyértelmű legyen az oldalainkon való eligazodás.

Sokszor ebbe a csoportba tartozónak tekinthetők a kisebbségben élők, és a szegények is.

3.2.2 Technológiailag megkülönböztetettek

Itt nem a felhasználó személye a fontos, hanem az általa használt eszközök képességeiben rejlő különbségekből adódó problémákat vesszük sorra. Azok is hátrányos helyzetűnek számítanak, akik kis képernyőméretet használnak böngészésre, akik elavult hardvert vagy elavult böngészőt használnak valami oknál fogva netezésre.

3.2.2.1 Eltérő képernyőméret

Vannak kis és vannak nagy képernyőjű megjelenítők. A világ a felé halad, hogy mindkét szélsőségből egyre több lesz, hiszen sorban jelennek meg a böngészésre képes mobiltelefonok, PDA-k, valamint a 22 colos és annál nagyobb monitorok is. Éppen ezért nehéz egy optimális felbontást választani. A CNN honlapján például, egy 19 colos monitoron a teljes képernyő kevesebb, mint kétharmadát teszi ki a fix szélességű oldal. Ez még a kisebbik baj, de ráadásul a tényleges hír elfér ennek a résznek a negyedébe. A hír maga a teljes képernyő 17%-át foglalja el vízszintesen (220 pixel).

Kis képernyők esetén ugyanakkor gyakori a vízszintes görgetés. Ezt lehetőleg el kell kerülni, mivel nagyon átláthatatlanná teszi az oldalt és nagyon nehézkes lesz a kezelése is. Amennyiben PDA-ra is szeretnénk optimalizálni az oldalunkat, és nem szeretnénk az újratöltést rábízni a gép böngészőjére, ezt stíluslapokkal megoldhatjuk.

3.2.2.2 Elavult böngészők

Nem szabad azt feltételezni, hogy mindenki rendelkezik a legújabb böngészőkkel. Sokan használnak olyan régi számítógépet és operációs rendszert, amely nem is teszi lehetővé a böngésző frissítését vagy éppen nincs rá lehetőségük, hogy frissebbet tegyenek fel a számítógépre (nincs megfelelő jogosultságuk hozzá). Ezek a böngészők például nem képesek JavaScript vagy Flash animációk futtatására, esetleg még a stíluslapokat se támogatják. Az is előfordul, hogy bizonyos cégeknél biztonsági okokból vannak a JavaScripek és a Flash animációk letiltva. Ezekről sokkal szebb lesz (lehet) ugyan az oldal, esetleg plusz funkcionalitást adhat (például kliens oldalon ellenőrzi az űrlapba írt adatokat), de az oldal működőképessége nem függhet ezek meglététől.

3.2.2.3 Gyenge, lassú hardver

Ez is egy lényeges szempont, és talán Magyarországon sokkal lényegesebb, mint azt gondolnánk. Egy régi gép kevés memóriával rendelkezik, nem tud bizonyosnál nagyobb oldalakat megjeleníteni, és természetesen régi böngészővel rendelkezik, ami további korlátozás. A leggyakoribb hardveres probléma ugyanakkor a lassú Internet-kapcsolat. Amikor a fejlesztő készíti a honlapot, azt vagy a saját gépén teszi, vagy egy, a szerverrel helyi hálón összekötött gépen. Ilyenkor könnyen abba a hibába esik, hogy nem veszi észre, hogy milyen nagy adatállományokat, képeket, táblázatokat szűrt be az oldalba. Egy olyan oldalon, ahol a menü képekből van kitéve, és szépen megváltozik a színe, ha a felhasználó fölé viszi az egeret, könnyen több száz kilobájtosra is dagadhat az oldal mérete. Ez egy régi modemnél párperces töltőgetést jelent, amit csak igazán fontos és tartalmas honlap esetében vár ki a felhasználó, legtöbbször inkább odébbáll. ^[4]

3.2.3 Speciális célcsoportok

Akadálymentes honlap készítésekor érdemes arra odafigyelnünk, hogy oldalunkat fiatalok (gyerekek) és idősök is látogathatják. Ezen korcsoportoknak más és más dolgokra van szükségük. Nézzük, mik ezek.

3.2.3.1 Gyerekek

A gyerekek sokkal többre képesek, mint azt gondolnánk, három éves kortól már könnyedén tudják használni az egeret, viszont - mivel nem ismerik a betűket - a billentyűzet használata náluk a speciális gombokra korlátozódik. A figyelmük könnyen irányítható, viszont könnyen el is kalandozik, azaz az oldalon lévő legszínesebb, legérdekesebb, vagy mozgó elemre fognak figyelni, rákattintani. További fontos igény, hogy a gyerekeket vezetni kell, lépésről lépésre, ha végzett egy feladattal, oldallal jön a következő, szépen sorban.

3.2.3.2 Idősök és informatikai szempontból alacsony képzettségűek

A mai nyugdíjas generáció életében a technika hatalmas fejlődésen ment keresztül, ennek a korosztálynak nagyon lényeges a korábban már többször is hangoztatott átláthatóság és a lépésenkénti vezetés. Így körülbelül érti, hogy mi történik, de ha mégis elveszne, biztos lehet benne, hogy a program végigvezeti, és a végén kilyukad valahol. A manapság olyan közkedvelt felugró (popup) ablakok például teljesen elrontják ezt az átláthatóságot. Mindenkinek jól jön egy jól felépített súgó, de azok számára, akik nem annyira értenek az informatikához, ennek megléte és használhatósága létfontosságú lehet.

3.2.4 Működés ellenőrzése

Érdemes az elkészült honlapot több böngészőn és a böngészők különböző verzióján is ellenőrizni, hogy biztosan megjelenik-e az a tartalom, amit szeretnénk. Jelenleg a legelterjedtebb böngészők: Internet Explorer (v6, v7, v8-as verziók) [A 6-os verzió már nem ajánlott mivel nem támogatott]. Firefox (v2.x, v3.x), Opera, Chrome.

3.2.5 W3C Validator

Ahhoz hogy minden böngésző meg tudja jeleníteni a honlapot, minimum követelmény, hogy a HTML, XHTML, CSS kódok szabványosan legyenek megírva. Ezt tudjuk ellenőrizni a W3C Validátor segítségével.^[13]

Lásd: <http://validator.w3.org/>

3.2.6 W3C – WAI

3.2.6.1 W3C

A World Wide Web Consortium (W3C) foglalkozik a webes szabványosítással (ajánlásokkal). Jelszava a semlegesség és egyetértés. A szervezetet Tim Berners-Lee alapította 1994-ben MIT-n.

Céljai:

- Ajánlások és szabványok segítségével segíti összehangolni a különböző webes technológiákat.
- Előmozdítja a web fejlődését.
- Garantálja a technológiák széleskörű felhasználhatóságát.

Három alapintézménye van:

- Az amerikai MIT
- A franciaországi székhelyű ERCIM
- Japán Keio Egyetem

A konzorciumnak ma több mint 400 tagja van szerte a világon, fejlesztők, kutatólaboratóriumok, termékforgalmazók, tartalom-szolgáltatók, mindazon területek képviselői, akik elkötelezettek a web egységesítése iránt. 2002-től van Magyar Iroda is. A W3C-nek három magyar tagja van: az MTA SZTAKI, a Budapesti Műszaki és Gazdaságtudományi Egyetem, valamint a KOPINT-DATORG. A helyi közösségekkel való kapcsolattartást 17 regionális iroda segíti.

Magyar iroda tevékenysége, szolgáltatásai:

- W3C technológiák népszerűsítése.
- Konferenciák, oktatásszervezés.
- Tanácsadás.
- Akadálymentesítési és használhatósági tanulmányok készítése.
- Magyar nyelvű információk nyújtása (weboldal, hírlevél, szóróanyagok, RSS)
- W3C dokumentumok fordítása.

Magyar iroda célja a magyar webes élet fejlődésének elősegítése és a nemzetközivel kompatibilis hazai webes szabványok létrehozása.

W3C szervezet fogalmazta meg a jól ismert HTML, XML, CSS stb. ajánlásokat is, de ezen kívül még sok egyéb, kevésbé ismert szabvánnyal és ajánlással rendelkezik.

A W3C Web Accessibility Initiative (WAI) a W3C akadálymentesítési kezdeményezése. Ezen belül több csoport dolgozik, amik közül a legfontosabb a Web Content Accessibility Guidelines (WCAG). Ez a szervezet 1999-ben kiadott egy dokumentumot, amit ajánlássá nyilvánítottak (Web Content Accessibility Guidelines 1.0 [WCAG1]).^[3]

3.2.6.2 WCAG 1.0

WCAG 1.0 három szintet különböztet meg, amihez különböző feltételek tartoznak.

Melyek ezek a főbb feltételek:

(Forrás: <http://bartal.org/web-accessibility-akadalymentes-weboldalak.html>)

”

- Szöveges alternatívát szükséges nyújtani a vizuális és hallható tartalomhoz (jellemzően vakok, siketek részére).

- Ne támaszkodjunk egyedül a színekre (rossz: „Kattintson a zöld gombra a továbblépéshez”)

- Valid (X)HTML és style sheet-ek használata.

- Tartalom nyelvének megadása.

- Ha táblázatokat használunk, szerkezetük legyen olyan, hogy lineáris szöveggé transzformálva is értelmezhetőek legyenek.

- Az új technológiák (Java appletek, Flash intrók és menük, illetve más ActiveX objektumok) mellett nyújtsunk szöveges alternatívát.

- A felhasználói felület minden funkciója legyen közvetlenül elérhető (azaz ne hozzunk létre fontos navigációs linkeket pl. kliens oldali script-tel).

- Tervezzünk platform függetlenül.

- Vegyük figyelembe a W3C szabványait és ajánlásait

- Használjunk egyértelmű navigációs elemeket. A használt HTML elemek feleljenek meg a dokumentum struktúrájának. Ne használjunk tehát logikai elemeket formázási célra (h1...h6, blockquote), a h1...h6 hierarchiát tartsuk meg (pl. h1 után ne használjunk h4-et, átugorva a köztes heading szinteket).

- Érhetően és egyszerűen fogalmazzunk.

”

Csoportokról:

Priority 1 (A) - Ez az első csoport. Olyan ellenőrzőpontokat tartalmaz, amelyek betartása kötelező egy akadálymentesített honlap létrehozása esetén. Ha ezen ellenőrzőpontokat nem tartja be az oldal, akkor egy vagy több társadalmi csoport nem tud hozzáférni ehhez a webes tartalomhoz. Ha betartja, akkor ezekbe a csoportokba tartozó felhasználók használhatják ezeket a webes tartalmakat. Ha egy honlap betartja az összes priority 1-es feltételt, akkor kiteheti a W3C WAI-A logót.



Priority 2 (AA) – Ez a második csoport. Ezen ellenőrzőpontokat a webes tartalmat fejlesztőnek ajánlott betartania. Ha nem tartja be az itt felsorolt követelményeket akkor egy vagy több társadalmi csoport nehezen tudja használni ezen webes tartalmakat. Ezen ellenőrzőpontokba tartozó feltételek teljesülésével jelentős akadályok szüntethetők meg a hozzáféréssel kapcsolatban. Ha egy honlap betartja az összes priority 1-es és priority 2-es feltételt, akkor kiteheti a W3C WAI-AA logót.



Priority 3 (AAA) – Ez a harmadik csoport. Ezen ellenőrzőpontok betartása a webes tartalomfejlesztőnek csak lehetőség. Ha ezeket betartja a webes tartalomfejlesztő, akkor egy vagy több társadalmi csoport könnyebben hozzáférhet a webes tartalomhoz, azaz javul a tartalom elérhetősége. Ha egy honlap betartja az összes priority 1, priority 2 és priority 3-as feltételt akkor kiteheti a W3C WAI-AAA logót.



Részletes útmutató itt tekinthető meg:

<http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/>^[11]

Részletes checklist itt tekinthető meg:

<http://www.w3.org/TR/WCAG10/full-checklist.html>^[10]

3.2.6.3 WCAG 2.0

Munkaterv fázis indul (2005.nov.23.).

Jelenlegi ajánlás (2008.december.11.)

WCAG 2.0 sokkal általánosabban fogalmazza meg a célokat. Míg a WCAG 1.0-ban erős volt az eszközfüggőség addig a WCAG 2.0 inkább elveket kíván megfogalmazni.

Bizonyos esetekben szigorúbb feltételeket határoz meg, mint az 1.0-ás kiadás.

A 2.0-ás kiadás alkotói igyekeztek úgy megfogalmazni az ajánlást, hogy azon oldalak tulajdonosai, akik teljesítették WCAG 1.0-át azoknak alig vagy semmit se jelentsen az átállás.

4 alapelvet tartalmaz. Az alapelveken belül további irányelveket fogalmaztak meg. Most csak a 4 alapelvet sorolom fel és megemlítem, hogy melyik csoport mennyi fő irányelvet tartalmaz. Ha részletesebben érdekel valakit a téma, akkor az oldal alján lévő linken részletesen lehet olvasni az ajánlásról.

Lássuk akkor a 4 alapelvet:

- Észlelhetőség (érzékelhetőség) - Az információt és a felhasználói felület elemeit olyan módon kell megjeleníteni a felhasználók számára, hogy azokat érzékelni tudják.

Jelenleg ez a csoport 4 fő irányelvet tartalmaz.

- Működtethetőség (működőképesség) - A felhasználói felület részei és a navigáció működőképesek legyenek. Jelenleg ez a csoport 4 fő irányelvet tartalmaz.

- Érthetőség – Az információnak és a felhasználói felület kezelési módjának érthetőnek kell lennie. Jelenleg ez a csoport 3 fő irányelvet tartalmaz.

- Robusztusság (erőteljesség) - A tartalomnak elég robusztusnak kell lennie ahhoz, hogy a különböző alkalmazások által, beleértve a kisegítő technológiákat is, megbízhatóan értelmezhető legyen. Jelenleg ez a csoport 1 fő irányelvet tartalmaz.^[2]

Részletesen itt olvashatunk róla magyarul:

<http://www.w3c.hu/forditasok/WCAG20/>^[12]

<http://www.w3c.hu/forditasok/UNDERSTANDING-WCAG20/index.html>

Eredeti angol oldal elérhetősége:

<http://www.w3.org/TR/WCAG20/>

3.3 Ergonómia /felhasználóbarát oldalak/ (usability)

A web-ergonómia arról szól, hogyan készíts használható weboldalt (felhasználóbarát weboldalt).^[18,22] Felmerül a kérdés, hogy mitől is felhasználóbarát egy weblap? Akadálymentes, ergonomikus, szabványos, figyelemfelkeltő. Továbbá gyorsan^[23] töltődik le és viszonylag hosszú ideig a weblapon tudja tartani az internet felhasználót.

További kérdések használhatósággal (ergonómiával) kapcsolatban:

- Mi a jó az ember számára?
- Mitől elégedett egy felhasználó?
- Mitől lehet növelni a komfortérzetét?

Elsősorban ezzel a kérdésekkel foglalkozik ergonómia. Ha úgy építünk fel egy weboldalt, ami jó az emberek számára, akkor jobb látogatottságot érhetünk el, nőhet az elégedett látogatók száma és nőhet az oldalunkra történő külső hivatkozások száma is.

Miért lesz jobb a látogatottsága az oldalnak?

- A kereső találati listán is jobb helyezése lesz a szabványos oldalaknak.
- Fogyatékkal élők - például színtévesztők, látási problémákkal küzdők - is használni tudják a weboldalt. (Részletesebben az előző fejezetben lehetett róla olvasni)

Miért lesznek elégedettebbek a látogatók?

- Azért, mert a weboldal gyorsan töltődik le (rövid a letöltési idő).
- Jobban használható, mert ergonomikus kialakítású.

Nézzünk egy pár tippet, hogy hogyan lehet hatékony az oldalunk.

3.3.1 Típek egy hatékony weblap kialakításához:

(Forrás: <http://www.sitemarketing.hu/Keresooptimalizalas/Website-ergonomia-hatekony-honlap.html>)

”

1- Először is weboldalunk tartalma legyen olvasható!

Ez egy nagy kontrasztarányú háttér és betűszín választást jelent.

Lehetőleg fehér háttérrel fekete, vagy sötétszürke betűkkel.

2- A betűk mérete minimum 10px. méretű legyen

3- A sorok nagyságrendileg 500-600 pixel szélességűek legyenek.

4- A szöveget tagolva helyezzük el, hogy az olvasó azt érezze, hogy na még ezt a kis részt elolvasom.

5- Használjunk alcímeket, amelyek 2-4 pixel-el nagyobbak legyenek, mint a szöveg betűmérete.

Ha megírtuk az oldal tartalmát, olvassuk el csak az alcímeket, mert a látogatók nagy része ezt fogja tenni.

Az alcímeknek fel kell tudni kelteni az olvasó érdeklődését a tartalom iránt.

6- Ne használjunk 2-nél több betűtípust egy oldalon, ne döntsük a betűket, legjobb, ha a normál Arial fontot használjuk.

7- Ne használjunk 3-nál több színt az oldalon, mert zavaró.

8- A látogató fejében a link kék, és alá van húzva, ezért használjuk mi is ezt a megoldást.

9- Az oldal kezelhetőségében, menü elhelyezésben ne találjunk ki túl sok újat, ne próbáljuk gyökeresen átnevelni a látogatót olyan módszerekre, amellyel csak a mi weboldalunkon boldogul.

10- Törekedjünk az egyszerűsége, a design kialakításánál.

Ne használjunk olyan képeket, animációkat, flasheket, amelyek nem az oldal célját szolgálják, mert elvonják a látogató figyelmét attól a feladattól, amelyet mi szeretnénk, hogy végrehajtson. (vásárlás, letöltés, stb...)

11- A navigálhatóságot, adott feladat végrehajtását ne magunk, vagy kollégánk tesztelje, mert bármennyire is hihetetlen, mi nem olyanok vagyunk, mint a célcsoportunk!

12- A lapozás közben legyen egyértelmű a látogatónak, hogy még mindig ugyan azon a weblapon van.

13- Fogjuk meg a látogató kezét, és vezessük végig a feladaton.

Tájékoztassuk előre arról, hogy mi lesz a következő lépés.

Minden lehetséges költséget, pl. postaköltség a webshopokban, közöljük előre a látogatóval.

14- Ha azt szeretnénk, hogy rákattintson egy linkre, akkor az legyen nagy és feltűnő, lehetőleg gomb.

15- Űrlap kitöltésénél, ha valamit elront, vagy kihagy a látogató, akkor ne egy üres weblapot adjunk neki vissza, hogy újra előről keljen kezdenie a feladatot.

A rendelés leadásánál csak a kitöltendő adatlap szerepeljen az oldalon!

Ne legyen semmi, ami felkeltheti a látogató érdeklődését, és elviheti Őt a rendelés véglegesítéséről!

16- Ha egy kérdésünkre több, de előre megfogalmazható választ várunk a látogatótól, akkor használjunk rádiógombot, sok lehetőségénél pedig legördülő választási lehetőséget.

17- A kitöltendő rubrikák mellé tegyünk magyarázó szövegeket, pl. telefonszám formátum.

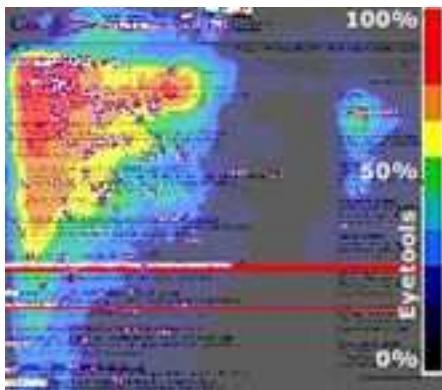
18- A bizalomépítést vegyük komolyan, jól látható helyen legyen elérhető a kapcsolat és a GYIK.

19- Használjunk belső keresőt, amely lehetőleg a jobb felső sarokban legyen elhelyezve.

[14]
„

20- A látogató az aranyháromszögnek nevezett részt vizsgálja át először. Ez a rész legyen a leginformatívabb! Melyik is ez a rész? A képernyő bal felső része.

Az alábbi képen látszik hődiagrammal nézve melyik is a legtöbbet nézett terület egy weboldalon:



Kép forrása: http://eyetools.com/research_google_eyetracking_heatmap.html

3.3.2 Mitől nőhet egy felhasználó komfortérzete?

Legfontosabbak:

- Honlap letöltési időt próbáljuk alacsonyan tartani
- Web design
- Használhatóság
- Tartalom
- Kellemes, harmonikus színek

3.3.2.1 Honlap letöltési idő:

A weblap letöltési idejét általában a képek növelik meg. Ha nagy képeket rakunk fel nő az oldal leöltési ideje. Ha sokáig kell várni a felhasználónak az oldal behozatalára, akkor esetleg továbbáll, és nem várja meg az oldal betöltését. Éppen ezért nem célszerű a főlapra nagyméretű képeket rakni. Optimalizáljuk az oldalunkon a képek méretét.

3.3.2.2 Web design:

Próbáljunk a felhasználók igényeit minél jobban kielégítő oldalt létrehozni. Ehhez hozzá tartozik egy jó menürendszer is. Lényeges az oldalon a betűk mérete és típusa is. Háttérszínekhez megfelelő betűszíneket válasszunk. A felhasználóbarát kialakítás a józanészre támaszkodik. (A betűk méretéről, típusáról és a színekről a következő fejezetben olvashatsz részletesebben.)

3.3.2.3 Használhatóság:

Használhatóság egyszerű szabályai:

- Tedd akadálymentessé a honlapot!
- Kerüld azt, ami zavarja (zavarhatja) a felhasználót!
- Alkalmazz egyszerű, következetes navigációt!
- Spórolj a felhasználó idejével!
- A menüpontok száma lehetőleg 5-7 között legyen!
- Jól válaszd meg háttérszínt és szöveg színeket!
- Jól látható helyre tedd az impresszum, kapcsolat linket!
- Segítsd HTML oldaltérképpel az eligazodást! ^[24]

3.3.2.4 Tartalom:

Olyan tartalmat tegyünk fel ami a érdekes és hasznos az oldal látogatói számára. A tartalom jó elhelyezéséhez fontos tudnunk hogyan olvasnak az emberek a weben. Egy vizsgálat keretében (<http://www.useit.com/alertbox/9710a.html>) azt találták, hogy az emberek nem olvasnak szóról-szóra, hanem pásztáznak. Átfésülik az oldalakat kulcsszavak, kulcsmondatok után. Legalább is az emberek 79%-a ezt teszi. Éppen ezért pásztázható oldalak kellenek. Ezt hogy érhetjük el?

- Kiemelt kulcsszavak kellenek.
- Fejezetcímek kellenek.
- Egy bekezdésben csak egy gondolat szerepeljen.
- Felsorolásos vagy számozott listák kellenek.
- Fordított piramis módszerrel írjunk (konklúziót előre, aztán részletezzünk).^[20]
- Tördeljük a szöveget kisebb egységekre.
- A lényeges információkat az oldalak felső részére tegyük.^[19]

Továbbá írjunk hatékonyan. Ezt hogyan tehetjük meg? Legyen az írásunk tömör, tárgyilagos (csak tényeket tartalmaz) és pásztázható (sok kiemelést és számozott, illetve felsorolásos listát tartalmazó szöveg).^[21]

3.3.2.5 Kellemes, harmonikus színek használata:

A színekről, színkódokról, színek hatásáról a következő fejezetben olvashatsz részletesebben.

Lásd a 4.6-os fejezetet a 89. oldalon.

3.3.3 Ergonómiai hibák

(Forrás: http://ergomania.blog.hu/2009/02/20/ergonomiai_tipushibak_a_magyar_weben_2009_ben)

3.3.3.1 „I. Ergonómiai hibák okai

1. weblapfejlesztések esetlegesek
 - nincsenek tervek, hanem rosszul artikulált vágyak
 - célok homályosak
 - szerepek nem tisztázottak
2. szekvenciális fejlesztés
megrendelő » designer » fejlesztő » optimalizálás
gondok a szekvenciális kivitelezéssel:
 - nincs visszajelzés
 - nincs megfelelően előkészítve a következő lépés
 - hibák öröklődnek és halmozódnak
 - későbbi szakértők tudásával a korábbi szakaszok is sikeresek lehetnének
 - jelentősen drágább
3. szakértők hiánya
 - fejlesztő
 - designer
 - seo szakértő
 - projektvezető
 - marketinges
 - tesztelő
 - ergonómus
 - stratégiai tervező
4. nem látják az ergonómia fontosságát
 - nem hallottak róla
 - ez is csak pénzlenyúlás
 - az ergonómiához mindenki ért
 - a mi embereink értenek hozzá
 - általános igazságok puffogatása
 - túl bonyolult, túl sok idő
 - nincs rá pénz
5. ergonómia alárendelt szerepben
 - ergonómia nem tervezésnél,
 - hanem az áttervezésnél kerül fókuszba
 - előnye:
 - már jól látjuk a tartalmakat
 - vannak ismereteink a felhasználói szokásokról
 - hátránya:
 - radikális változással szembeni ellenállás
 - pluszfejlesztésre nincs keret
 - programozói ellenállás
 - szerkezetet a design alakítja ki vagy nincs szerkezet
 - továbbra sem gondolkozunk az emberek fejével
6. elmaradás 6-8 év

3.3.3.2 II. létező problémák

1. átgondolatlan navigáció
 - redundancia
 - nem sztenderdizált elnevezések
 - értelmezhetetlen elnevezések
 - semmitmondó elnevezések
 - régimódi menüstruktúra alkalmazása
 - sok menüpont
2. keresési alkalmazások primitív színvonala
 - sok helyütt hiányzik
 - ha van teljesen megbízhatatlan
 - részlegesen keres csak
 - keresési találatokban nem mindig lehet látni, hogy miben is volt a találat tulajdonképpen
3. információ strukturálatlan és elavult
 - regisztráció nehézkes volta
 - felesleges adatok bekérése
 - rossz strukturáltság
 - felhasználó számára kényelmetlen adatbekérdezés
 - nem lehet tudni mire való
4. nem megfelelő technológiák
 - frame
 - SQL lekérdezések url-ben
 - animált gifek
 - flash túlhajszolás
5. marketing ízű kommunikáció
 - 90-es évek unalmas marketing szövegei
 - felhasználó félrevezetése
 - sok szó, kevés tartalom
6. fókuszvesztés
 - nem lehet tudni mi a weblap fő célja
 - maguk az üzemeltetők sem mindig tudják
7. sablonos gondolkozás
 - termékek megjelenítése
 - termékek előnytelen bemutatása
 - termékek nincsenek jól összekapcsolva
 - termékekről az információ áttekinthetetlen
 - termékeket nem lehet összehasonlítani
 - elégtelen vagy nem megfelelő információ a döntéshez
8. színes katalógus szindróma
 - nyomtatott anyagokban bevett megoldások
 - beszkenelt anyagok, pdf használata
 - vonzódás a színes információhiányos felületekhez „^[16]

4. Grafika, megjelenés

4.1 Bevezetés:

Most, hogy az előző fejezetekben megismertük már az alap technológiákat. Szó esett a különböző csoportoknak milyen igényeik vannak egy weboldallal kapcsolatban, és velük kapcsolatban mire kell figyeljünk.

Ebben a fejezetben szót ejtenék a fontosabb és elterjedtebb grafikai formátumokról, amiket az oldalunkon fel tudunk használni.

Egy pár oldalt ajánlanék, ahonnan grafikai elemeket lehet leszedni vagy csinálhatunk éppen (lásd: logók). Textúrákról, fontokról melyekkel az oldalunkat tovább csinosíthatjuk.

Ebben a fejezetben szó lesz a színekről, színkódokról is, Milyen színek vannak és azoknak milyen hatásuk van. Szó lesz még arról is, hogyan keletkeznek, mik is azok az RGB és CMYK színek?

A fejezet végén pár szóban szó lesz az egyik leghíresebb grafikai szoftverről az Adobe cég által készített Photoshopról programról is.

Akkor kezdjük mindjárt a weboldalunkon használható grafikai formátumok ismertetésével.

4.2 Fontosabb formátumok:

A következő részben a mostanában használt és elterjedt képformátumokról olvashatsz. Részletesebben ezek a következők: JPG, GIF, PCX, PNG, BMP, SVG.

Felsorolom ezeknek a formátumoknak az előnyeit és hátrányait is.^[14,15,16,17]

4.2.1 JPEG (Joint Photographic Experts Group)

Egyik legelterjedtebb képek tárolására alkalmas formátum.

A JPEG formátum a CMYK, az RGB és a szürkeárnyaltos színmodot támogatja. A GIF formátumtól eltérően a JPEG az összes színadatot megőrzi az RGB módú képekben. A képeket egy információvesztéssel járó algoritmussal tömöríti (veszteséges tömörítés) és így a kép mérete 10-100x kisebb helyet foglalhat el, mint tömörítés nélkül. A veszteséges tömörítés mértéke állítható. Nagyobb méretű tömörítés kisebb fájl méretet, de rosszabb képminőséget eredményez. Kisebb méretű tömörítés nagyobb fájl méretet és jobb képminőséget eredményez. Ha maximális minőségre állítjuk a tömörítést, akkor a kép tömörítése legtöbb esetben megkülönböztethetetlen lesz az eredetitől. Elsősorban állóképek és rajzok tárolására ajánlott. 16.7 millió színt támogat. Gyakran alkalmazzák weblapok esetén is. Grafikonok vagy más egyéb hirtelen színátmenetű ábrák tárolására nem ajánlott.

Előnye:

- Kódoláskor választható a tömörítés mértéke.
- Elterjedt, támogatott.
- Webes felületre ajánlható, mint fájlformátum.

Hátránya:

- Információvesztéssel járó tömörítés.

Használt kiterjesztései: .jpeg .jpg .jpe^[6,7]

4.2.2 GIF (Graphics Interchange Format)

1987-ben jelent meg az első verzió (GIF87a). Az akkoriban használt RLE formátumot akarták ezzel a formátummal helyettesíteni (RLE csak fekete-fehér képeket volt képes kezelni). 1989-ben a CompuServe kiadta a továbbfejlesztett változatot (GIF89a). Ez a változat már engedte az animáció létrehozását azáltal, hogy több képet tárolt el egy fájlban. A GIF fájl verzióját a fájl első 6 bájtyának segítségével úgy lehet megállapítani, hogy azokat ASCII jelekként értelmezzük és megvizsgáljuk, hogy a GIF87a illetve GIF89a sztringet tartalmazza-e. Ez a formátum 8 bites színmélységű (256 színű) képeket tud kezelni, ezért elsősorban rajzok tárolására, grafikonok és egyéb hirtelen színátmenetű ábrák tárolására való.

Tömörítésre a LZW algoritmust használja. Ezt az algoritmust szabadalom védi. Éppen emiatt hozták létre a PNG formátumot.

Előnye:

- Maga idejében gyors volt az algoritmus.
- Kis fájl méret érhető el vele.
- Animációk is tárolhatók vele.
- Webes felületre ajánlható mint fájlformátum.
- Átlátszó réteg definiálásának lehetősége.

Hátránya:

- LZW eljárást szabadalom védte. Szabadalom lejárt:
 - 2003 (Nagy-Britanniában, Franciaországban, Németországban és Olaszországban)
 - 2004 (Japán, Kanada)
- Csak 256 szín tárolására alkalmas.

Használt kiterjesztései: .gif^[8,9]

4.2.3 PNG (Portable Network Graphic)

Veszteségmentes tömörítésre alkalmas fiatal formátum. A GIF utódként tekintenek rá. Tömörítésre nem az LZW algoritmust (jogi problémák miatt) használja, hanem a deflation algoritmust használja.

PNG előnye a GIF-hez képest:

- Fokozatosan átlátszó képeket tesz lehetővé alfa csatornákkal.
- Képek fényességének függetlenítése a megjelenítéstől gamma korrekcióval.
- Hasonló vagy jobb tömörítés.
- True color módban lévő képeket is képes tárolni.
- Veszteségmentes tömörítés

Hátránya:

- A JPG tömörítéshez képest a tömörítési arány nem olyan kedvező (viszont veszteségmentes tömörítés).

Használt kiterjesztései: .png^[12,13]

4.2.4 BMP (Bitmap)

A Microsoft által létrehozott bináris fájlformátum. Az angol szó magyarul bittérképet jelent. A BMP formátum négy részből áll:

- Fájlfejléc

A fájlra vonatkozó alapvető adatokat tárolja.

- Információs fejléc

Az eltárolt kép jellemzőit írja le (felbontás, színmélység, stb.)

- Paletta (ha van)

Az eltárolt kép által használt színek RGB kódjait sorolja fel.

- Bittérkép

A kép tényleges tárolási helye, ahol képpontról képpontra jegyzik fel azok színeit. A bittérkép a kép képpontjait sorfolytonosan tárolja, ami alapvetően kétféleképpen történhet.

Ha a színmélység 8, vagy nála kisebb, akkor színindexeket sorol fel, amelyek a paletta színeire mutatnak (a paletta első színe kapja a 0-s indexet). Minden színindex egy képpontot ír le.

Ha a színmélység 24 bites, akkor RGB színkódokat sorol fel, és ilyenkor nincsen a fájlban paletta. Minden (3×8 bites) RGB színkód egy-egy képpontot ír le. Bár a színindexes bittérkép csak kevesebb színt különböztethet meg, mérete jóval kisebb az RGB színkódosénál, hiszen egy képpont nem 24, csak 8, 4, 2 vagy 1 bitet foglal el (természetesen, egy bittérképen belül csak egyféle hosszúságú színindexeket használnak).

A bittérkép további jellemzői:

- Az RGB számhármassokban B; G; R a sorrend.

- A kép soronként lentől felfelé haladva tárolódik (vagyis a kép legalsó sora kerül a bittérkép legfelső sorába).

- A bittérképen belül a képnek megfelelő sorokat szükség szerint 0-s bitekkel egészítik ki úgy, hogy minden sorhoz négygyel osztható számú bájt tartozzék.

Előnye:

- Tömörítést nem használ (veszteségmentes)

- Parancsra gyorsan megjelenik, mivel nem kell használat előtt a képet kitömöríteni, mint a .jpeg esetén.

- Elterjedt, támogatott.

Hátránya:

- Nagy fájl méret

- Nem ajánlott a webes felületen való használata.

Használt kiterjesztései: .bmp^[11]

4.2.5 SVG (Scalable Vector Graphics)

Az SVG egy XML alapú leíró nyelv. W3C által definiált nyílt szabvány. W3C keretében SVG Working Group fejlesztette ki. 1998-ban kezdődött a fejlesztés. SVG 1.0 ajánlás 2001-ben. SVG 1.1 ajánlás 2003-ban. Jelenleg dolgoznak a SVG 1.2-esen.

A böngészők jelentős része natívan támogatja. A natív támogatás azt jelenti, hogy beépülők vagy kiegészítők nélkül képes az SVG-t kezelni.

Három fajta grafikus elemet támogat az SVG:

- vektorgrafikus alakzatok^[*]
- rastergrafikus és digitális képeket^[**]
- Szövegeket

Előnye:

- Ingyenes, nyílt szabvány.
- Böngészők részéről natív támogatás.

Használt kiterjesztései: .svg^[4]

4.3 Templatek

4.3.1 Mik is a templatek? Mire jók?

A web template-k (honlap sablonok profi designerek által előre elkészített weblap designok, melyek üzleti és privát célokra egyaránt alkalmazhatók. Ezek a honlap designok tetszőlegesen alakíthatóak. Csak annyit kell tenni velük, hogy kicseréljük a szöveget, beillesztjük a saját képeinket az oldalba, és már el is készült a honlapunk.

4.3.2 Legnagyobb előnyük:

A sablonok legnagyobb előnye az idő- és költségtakarékosság. Minimális összegért juthatunk egy tökéletes internetes megjelenéshez a lehető legrövidebb időn belül.^[18]

Weboldalak ebben a témában:

<http://www.freecsstemplates.org/css-templates/>

<http://opentemplates.org/>^[3]

<http://www.oswd.org/>^[5]

*, ** megjegyzések: A vektor és rastergrafika közötti különbségről később olvashatsz.

4.4 Grafikák

4.4.1 Animált gif-ek

Weboldalainkat szebbé tehetjük például animált gif-ek használatával. Ezeket vagy magunk rajzoljuk meg (például Photoshoppal), vagy a netről is van jó pár ami ingyenesen leszedhető.

Ajánlott weboldalak ebben a témában:

<http://animaltgif.lap.hu/>

<http://www.animationfactory.com/en/animations/>

<http://webgrafika.lap.hu/>

4.4.2 Logók

Némelyik weboldalon gyorsan és egyszerűen létrehozhatunk magunknak saját logókat, amivel szintén feldobhatjuk az oldalunkat:

Weboldalak ebben a témában:

<http://www.coolarchive.com/logogenerator.php>

<http://cooltext.com/>

<http://www.grsites.com/logo/>

<http://www.3dtextmaker.com/>

4.4.3 Texturák

Weboldalunk háttéréül, vagy képek készítésekor jól jöhetnek olyan textúrák amik ingyenesen letölthetők.

Weboldalak ebben a témában:

<http://www.grsites.com/textures/>

<http://www.mayang.com/textures/>

4.4.4 Fotók

Milyen fotó kerüljön a weboldalra?

Legyen kis méretű! Elektronikus értelemben a fotó nagyságát a kilobyte méret határozza meg. Honlapra 100 kB alatti elegendő, sőt a galériaoldalak esetében minimális méretű indexképeket jelenítsünk meg, melyekre rákattinva nyílik meg a nagyméretű kép.

- A képfájl típusa JPG, GIF, esetleg PNG legyen, bár ez utóbbi némelyik böngészőn nem megfelelően jelenik meg.
- Legyen figyelemfelkeltő.
- Összhangban legyen a többi fotóval vagy a szöveggel.^[1]

Szót ejtenék itt még arról, hogy a neten található képek közül nem mind ingyenes és szabadon felhasználható. Ha el akarjuk kerülni azt, hogy olyan képeket használjunk fel az oldalunkon, ami nem ingyenes, akkor célszerű olyan oldalon nézelődni, ahol ingyenesen felhasználható képeket helyeznek el. Ilyen oldalakra mutatnék példát:

Ingyenesen használható képek letölthetők:

<http://www.sxc.hu/>

<http://www.bigfoto.com/>

<http://download-free-pictures.com/>

<http://www.freepixels.com/>

4.5 Fontok

Weboldalainkat nem csak grafikai elemekkel, hanem egyedi fontok használatával is csinosíthatjuk. Figyelembe kell venni viszont, hogy ha a felhasználó gépén nincsen feltelepítve az adott font, akkor nem úgy fogja látni a weboldalunkat, mint ahogyan mi szeretnénk.

Ajánlott weboldalak ebben a témában:

<http://www.grsites.com/fonts/>

<http://www.1001freefonts.com/fonts/afonts.htm>

<http://www.downloadfreefonts.com/fontfiles/afonts.htm>

<http://www.flamingtext.com/fonts/>

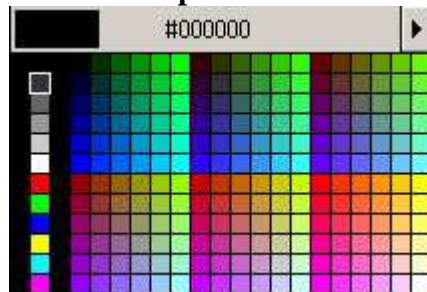
<http://www.acidfonts.com/>

http://fonts.goldenweb.it/index_file/l/en/d2/True_Type_Fonts

4.6 Színek, színekódok

4.6.1 Színekről

4.6.1.1 A színek csoportosítása



A színek többféle módon csoportosíthatóak.

Két nagy felosztásuk létezik:

- Fizikai felosztás.
- Színek hangulati hatása szerint.

Színek között 6 kitüntetett szín található (3 alap és 3 mellékszín):

Alapszínek (RGB):

- Vörös (alapszín) (Red)
- Zöld (alapszín) (Green)
- Kék (alapszín) (Blue)

Mellékszínek:

- Sárga (mellékszín)
- Narancs (mellékszín)
- Ibolya (mellékszín)

A feketét a színek hiánya, a fehéret a színek együttese hozza létre.

Az alapszíneket kikeverni nem lehet, de ezek keveréséből jön létre a teljes spektrum, azaz a mellékszínek.

A főbb színek adják a csoportosítás alapját, mivel az összes színt – mennyiségük miatt – képtelenség számba venni.

Fizikai felosztás

Fizikai megjelenés szerint a színek lehetnek:

- egyirányúak (fehér és fekete)
- többirányúak (tarka színek)

Megkülönböztetjük még ezen kívül:

- Telített színek

Egy szín telítettsége a fehértartalmától függ. A telített színek nem tartalmaznak fehéret, se feketét. Ezek az alapszínek (vörös, zöld és kék).

- Derített színek

A derített színek telítettek fehérrel való világosításával, míg nem a tompítottak ezek feketével történő sötétítésével jönnek létre.

A fizikai beosztás következő csoportja:

- Tisztaszínek

Spektrum színeiből állnak a vöröstől az ibolyáig. Elvileg nincsen szürketartalmuk. Egyre több fehéret hozzáadva az adott szín mind törtebbé (fehérebbé), feketével mind tompábbá (feketébb) válik.

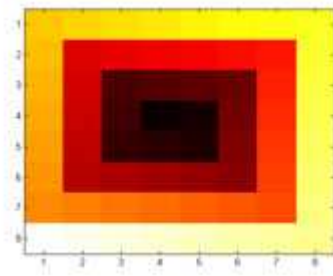
- Másodlagos színek

Ezek két tiszta szín keveredéséből létrejövő színek (narancs, zöld, viola).

A színek hangulati hatása

Hangulati hatás szerint megkülönböztetünk:

- hideg
- meleg
- semleges színek



Meleg színek:

A meleg szín kiemel, felhívja magára a figyelmet, fellendíti a hangulatot és a védettség érzését kelti.

Ilyen színek a vörös, a sárga és a narancs.

Hideg színek:

A hideg szín háttérbe tol, eltaszít, vagy elrejt, mindemellett nyugtat, pihentet.

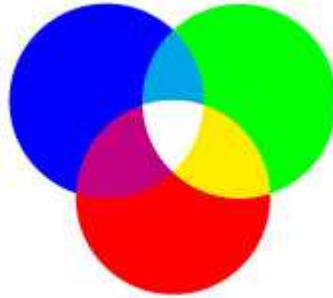
Ilyen színek a zöld színek egy része, az eget szimbolizáló kék, valamint a lila.

Semleges színek:

A semleges szín a meleg és a hideg határán helyezkedik el.

Leginkább szürke és árnyalatai, melyek folytonosan a fehértől a feketéig terjednek számtalan sok szürkeárnyalattal.

Ilyen színek a fehér, szürke, fekete és a barna.

Additív színek

Ezek a színek a három alapszínből (vörös, zöld, kék) állnak.

Ezen három színből mintegy 16,5 millió szín és árnyalat állítható elő a számítógépes képfeldolgozás RGB monitorain.

Az emberi szem ennek csak szűk tartományát képes megkülönböztetni. 2 additív szín keveredésével megkapjuk a szekunder komplementer színeket.

3 additív szín keveredésével a fehér fényt kapjuk.

Szubsztraktív színek

Ezek a színek a fehér fény spektrumában nem szerepelnek, így ezen színek valamely szín kivonásával állíthatók elő. Ezek a színek a ciánkék, bíborvörös és sárga. Angol szavak rövidítéséből CMYK (cyan, magenta, yellow, black) színeknek nevezik.

A CMYK színek a fehér fényből a színeket kivonják.

2 CMYK szín keveredésével primer komplementer színt kapunk.

3 CMYK szín keveredésével a fekete színhez jutunk.^[19]

4.6.2 Színek nevei, hexa és RGB kódjaik:

Honlapunkon többféle színt használhatunk. Vannak kitüntetett színek amiknek létezik nevük. Az alábbi weboldalon összefoglalták ezeket a neveket (140 szín) és a könnyebb felhasználás érdekében megadták a hexa és RGB kódjukat is.^[2]

<http://www.colorschemer.com/blog/2007/07/24/140-named-colors/>

4.7 Grafikai szoftverek:

4.7.1 Bevezetés

A grafikus programok a képkezelés és tárolás szempontjából két csoportra oszthatók.

1. Vektorgrafikus szoftverek

Az ábrázolás során a képet alkotó alakzatokat matematikai egyenletekkel írják le.

Előnyük:

- Az ilyen képek korlátlan mértékben nagyíthatók
- Általában kisebb helyet foglalnak el a képek.

Ilyen szoftver például a CorelDraw.

2. Rasztergrafikus szoftverek

Ábrázolás során a képet pixelekből (képpontokból) építik fel.

Az állományok a kép minden egyes képpontjának színét eltárolják. Azt a tulajdonságát használja ki a rasztergrafika az emberi szemnek, hogy bizonyos méret alatt az ember nem látja a pontokat, hanem folytonos színes felületeket lát.

Előnyük:

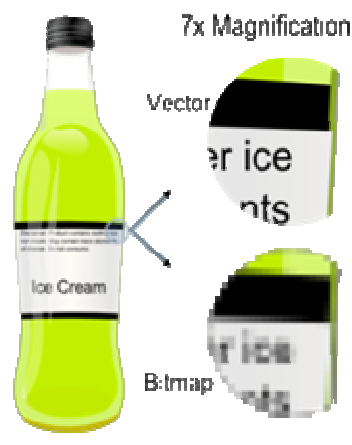
- Alkalmasak a valósághű képek ábrázolására.
- Minden egyes képpont külön szerkeszthető.
- Fényképek feldolgozására, retusálására kiválóan használhatóak.

Hátrányuk:

- Képek nagy lemezterület igénye.
- Nagyobb memóriaigény
- Korlátozott a képek nagyítási lehetősége (minőségromlás nélkül).

Ilyen szoftver például az Adobe Photoshop.

Példa a vektoros és raszteres grafika közötti különbség bemutatására:



4.7.2 CorelDraw

A CorelDraw egy vektorgrafikus alkalmazás.

Egy olyan általános célú grafikai alkalmazás, amivel hatékonyan lehet posztereket, logókat tervezni. Használják weboldalak, web grafikák, többoldalas szórólapok, prospektusok létrehozásához is.

Ez egy kereskedelmi szoftver (vagy szoftvercsomag része).

Kapható például a CorelDRAW Graphics Suite X4 csomagban is.

Tartalmaz szinte minden olyan funkciót ez a csomag, ami a professzionális grafikák elkészítéséhez szükséges.

Például 10 000 clipart, 1000 nagyfelbontású fotó, 1000 betűtípus és 80 mintasablon segíti munkánkat.

4.7.3 Photoshop

Az Adobe Photoshop egy rasztergrafikus képszerkesztő és fényképfeldolgozó program, melynek a fejlesztője az Adobe Systems.

Első verziója, a 0.63-as 1988-ban jelent meg Machintosra.

A Photoshop egyfajta etalon a képfeldolgozással és grafikával foglalkozók körében. Ezzel a szoftverrel szinte mindenféle képfeldolgozással kapcsolatos probléma megoldható. A programban korábban többféle újítás jelent meg, amelyet előbb vagy utóbb a konkurens programok fejlesztői is átvettek.

Például:

- Rétegek kezelése.
- Beépülő segédprogramok, azaz a „plug-in”-ek használata.

A képek rétegenkénti kezelése előnyei:

- A rétegekre helyezett objektumok, a kép más rétegeinek módosítása nélkül változtathatók meg.
- A rétegek tartama önállóan mozgatható.
- A rétegek átlátszóvá tehetők.
- Rétegek között különféle csoportosítási és egyéb műveletek végezhetők.
- Sorrend variálható.
- Rétegenként külön-külön grafikus szűrő alkalmazható.

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

A CS3 verziótól kezdve két változatban kerül forgalomba Standard és Extended. Utóbbi a Standard verzió minden szolgáltatásán kívül fejlett eszközöket tartalmaz a 3D modellezés és mozgó képek szerkesztéséhez.

A Standard verziót a fotósok, webes szakemberek és az amatőr felhasználóknak, míg az Extended bővített kiadást a filmesek, mérnökök és egészségügyi dolgozók számára ajánlja a gyártó.

A Photoshop verziói

1990 - Photoshop 1.0 (kizárólag Mac-re)
1991 - Photoshop 2.0 (kizárólag Mac-re)
1992 - Photoshop 2.5 (Mac, Windows)
1993 - Photoshop 3.0 (Mac, Windows)
1996 - Photoshop 4.0 (Mac, Windows)
1998 - Photoshop 5.0 (Mac, Windows)
1999 - Photoshop 5.5 (Mac, Windows)
2000 - Photoshop 6.0 (Mac, Windows)
2001 - Photoshop 7.0 (Mac, Windows)
2003 - Photoshop CS (8.0) (Mac, Windows)
2005 - Photoshop CS2 (9.0) (Mac, Windows)
2007 - Photoshop CS3 (10.0) (Mac, Windows)
2008 - Photoshop CS4 (11.0) (Mac, Windows)
2010 - Photoshop CS5 (12.0) (Mac, Windows)

Megjegyzés:

A 4.0-s változattól kezdve a Photoshop magyar nyelvű kezelőfelülettel is elérhető.^[20]

5. Webes marketing, keresőmarketing

5.1 Bevezetés

Mostanra már elkészülhetett a weboldalunk. Volt egy kis technológiai ismertetés. Volt mostanra szó már ergonómiai és akadálymentesítési szempontokról is. Csinosítottuk az oldalunkat különböző grafikus elemekkel is.

A most következő fejezet arról szól, hogy ne csak fent legyünk a neten, hanem a lehetőségekhez képest minél sikeresebb (minél több látogató látogassa az oldalunkat) is legyen az oldalunk. Hogyan tudjuk ezt elérni? Miket kell tennünk ennek érdekében?

5.2 Web-marketing

Három dologból áll, három dologra szokták mondani, hogy web-marketing.

Az első jelentésén érthetik a weben történő reklámkampányt.

Második jelentése lehet weboldal „eladásának” technikája.

Harmadik felhasználási terület ahol ezt szoktak használni az a keresőmarketing. Ugyanakkor értheti web-marketing alatt mindhármat együttesen is.

Web-marketingnél nem felétlenül az a cél, hogy minél több látogatót vonzunk a weboldalunkra, hanem az, hogy olyan látogató jöjjön a weboldalunkra, aki potenciális vevő is lehet.

Keresőmarketingnél az a cél, hogy minél több látogató látogassa az oldalunkat és ezt úgy érjük el, hogy a keresőben a weboldalunk jó pozíciót ér el. Ez még nem jelenti azt, hogy a weboldalunkat felkeresik, és pláne nem jelenti azt, hogy ha árulunk a weboldalon valamit, akkor vesznek is onnan valamit. Ez csak egy lehetőség a látogatottság növelésére.

Mivel a web-marketing és a keresőmarketing eszközei hasonlóak, és nem szeretném kétszer ugyanazt leírni, ezért ezeket a technikákat a keresőmarketingnél részletezném.

Térjünk is át a keresőmarketing és az eszközeire és azok részletezésére.^[1]

5.3 Keresőmarketing

A keresőmarketing egy gyűjtőfogalom. Az online marketing egyik fajtája. Tartalmazza mindazon technikákat, lépéseket, mellyel több látogatót tudunk hozni a weboldalunkra a keresőkből, azaz arról szól, hogy tudjuk növelni a látogatottságot a keresőkből (Magyarországon a Google elérése 95% felett van). Cél, hogy előkelő helyen szerepeljünk a keresőkben egy adott szóra. Az internetezők több mint egy harmada gondolja az, hogy aki első a keresők találati listáján, az egyben a legjobb is. Fontos azt tudnunk, hogy a keresőmarketing nem egy egyszeri munka, hanem egy folyamat. A kezdeti lépés a legnagyobb, de utána folyamatos követést igényel. Ahhoz, hogy ezt megértsük és elfogadjuk, ismernünk kell a keresőmotorok, PPC rendszerek működési elveit, illetve azt, hogy ez gyakorlatilag folyamatosan változik.

Keresőmarketing ágai.

1 - Keresőoptimalizálás (search engine optimization – rövidítve: SEO) vagy (search engine PR – SEPR)

Ez a folyamat a keresők találati listájában, azaz az organikus találati listán való előrébb jutást segíti. Ehhez többféle technikát vet be.

2 - Keresőhirdetés (pay per click – rövidítve: PPC)

Keresőkben megjelenő kattintás alapú fizetett hirdetéseket jelenti (szponzorált hirdetések).

3 – Fizetett listázás (paid inclusion)

Felmerül a kérdés, hogy melyik keresőmarketing megoldást válasszuk? A válasz nem egyszerű, mindig az adott szituációtól, weboldaltól függ.^[3,4,5,6]

Nézzük részletesen akkor őket:

5.3.1 Keresőoptimalizálás (SEO - search engine optimization)

A keresőoptimalizálás a Google előretörésével egyidős. Az amerikai cégek nagyon hamar ráeszméltek arra, hogy közvetlen hasznuk származhat abból, ha a találati lista elején szerepelnek.

Cél: Keresők találati listáiról minél több látogató érkezzon a honlapunkra, azaz a honlapunk látogatottságának a növelése.^[11]

Eszköze: A weboldalunk minél előrébb jelenjenek meg a keresési találatokban.

A keresőoptimalizálás többnyire weboldal optimalizálást jelent.

Két nagy ága van:

- on page lépések (weboldalon történő optimalizálás)
- off page lépések (weboldalon kívüli optimalizálás - például külső linképítés)

Keresőoptimalizálás lépései új weboldal illetve már létező weboldal esetén:

- Új weboldal esetén:
- Kulcsszóelemzés
- Domain névválasztás
- Tartalomírás
- Weboldal struktúra elkészítése, belső linkelés
- Az weboldal elkészítése, leprogramozása
- Title és Meta Tag –ek megadása
- Regisztrálás keresőkbe
- Külső linkkapcsolatok kiépítése
- Riportolás

Már meglévő weboldal esetén kicsi eltérést tapasztalunk:

- Benchmark, felmérés a jelenlegi helyzetről
- Kulcsszóelemzés
- Tartalom átírása
- A weboldal átstrukturálása, belső linkelés
- Technikai elemzés és átalakítás
- Title és Meta Tag –ek megadása
- Regisztrálás keresőkbe
- Külső linkkapcsolatok kiépítése
- Riportolás

A keresőoptimalizálás egy hosszú, időigényes folyamat. Azonnal nem látszik a hatása. El kell telni egy kis időnek, amíg a keresőmotorok robotjai, észlelik a változásokat. Ha észlelték a változásokat, akkor új helyre rangsorolják majd a weboldalunkat. Ha megtörténik az előresorolás, attól még ne várjunk rögtön drasztikus látogatottság növekedést.

Lépései részletesen:

5.3.1.1 Benchmark, felmérés a jelenlegi helyzetről (meglévő weboldal esetén):

Ha már van egy kész weboldalunk, akkor az első lépés, egy benchmark felállítása. Ennek a benchmarknak az eredményéhez fogjuk tudni viszonyítani a keresőoptimalizálás által kifejtett eredményeket.

Átvizsgáljuk a weboldalt keresőoptimalizálás és egyéb szempontokból, és készítünk egy rövid tanulmányt arról, hogy a weboldal jelenleg hogy áll, milyen eredményeket hoz.

Megvizsgáljuk, hogy a látogatók hogy érkeznek a weboldalra a keresőkből. Mikre keresnek rá? Milyen kulcsszavakra?

Megvizsgáljuk a weboldal tartalmának kulcsszó sűrűségét, azaz, hogy a gyakran használt szavakra hányadik helyen jelenik meg az oldal a keresőkben.

Megvizsgáljuk, hogy az adott weboldal domain-jének milyen a kora, ugyanis a keresők a rangsorolásnál ezt nagyon komolyan figyelembe veszik. Minél öregebb egy domain, amin működött weboldal, annál kedvezőbb lesz az elbírálás.

5.3.1.2 Kulcsszóelemzés

Talán a legfontosabb lépése a keresőoptimalizálásnak. Ahhoz hogy minél több látogatónk legyen meg kell állapítani, hogy mely kulcsszavakra, kell, érdemes, lehet előrébb hozni az oldalt. Össze kell írunk azokat a szavakat, amelyek szerintünk fontosak lehetnek az adott weboldal szempontjából.

Következő lépésként a Google AdWords kulcsszó eszközével, megnézzük, hogy ezeknek a szavaknak mekkora a keresettsége. A Google AdWords kulcsszó eszköz ajánlani fog még olyan kulcsszavakat, amelyeket azok keresik, akik az általunk javasolt kulcsszavakat is keresik.

Az ebből létrejött kulcsszólista szavaira rákeresünk egyenként a Google-ben, hogy mennyi oldal jelenik meg, magyarul mennyi a konkurens.

A két számból tehát, hogy mennyien keresnek az adott kulcsszóra, és mennyi a konkurens, kiszámoljuk azt, hogy mely kulcsszóra könnyű előrébb jutni, mely kulcsszavakra fontos, hogy elől legyünk.

2 számolási módszer használatos.

Az egyik a KEI, a másik a KER.

KEI (Keyword Effectiveness Index)

Kiszámolása: Vesszük a kulcsszó napi rákeresésének négyzetét és elosztjuk az kulcsszóra kapott oldaltalálatok számával.

Minél közelebb van a KEI az 1-hez, annál nagyobb eséllyel lehet optimalizálni az adott kulcsszóra a weboldalt, magyarul elég népszerű és nincs sok versenytárs.

KER (Keyword Effectiveness Rank)

Kiszámolása: A képletét nehéz lenne egyszerűen elmagyarázni. Nagyjából arról van szó, hogy itt már belép az is, hogy a listában szereplő kulcsszavakra átlag mennyien keresnek, és átlag mennyi a rákeresés:

P= rákeresések száma

C= konkurens oldalak száma

Ap= A listában szereplő kulcsszavak átlag rákeresései

Ac= A listában szereplő kulcsszavak átlag konkurens oldalai

Az eredmény egy százalék, amely azt mutatja meg, hogy az adott kulcsszólistából milyen százalékban lenne eredményes az adott kulcsszó. Minél magasabb ez a szám, annál könnyebb, annál érdekesebb kereső-optimalizálni rá.

5.3.1.3 Domain választás

Domain választás komolyan befolyásolja az adott weboldal sikerességét a keresőkben. Kulcsszóelemzés után tisztában vagyunk azzal, hogy melyik a legfontosabb kulcsszó. Ezután válasszunk olyan domain-t, amiben ez benne van, vagy ha van rá lehetőség, csak az a kulcsszó legyen a domain, a megfelelő végződéssel. Sokszor azonban ez nem lehetséges, mivel foglalt a domain. Mit tehetünk ilyenkor?

Vagy válasszunk más végződésű domain-t, vagy tegyük bele más kulcsszót.

A más végződésű domain-nel az a probléma, hogy a keresőknek számít, hogy ha én magyar nyelvű weboldalt csinálok, annak magyar, .hu végződése legyen. Ha más szót is beleteszünk a domain-be, gyengíteni fogja az a másik szó a legfontosabbat. Ilyenkor ajánlható: Tegyük bele a második legfontosabb kulcsszavunkat is a domain-be. Kötéjjellel, vagy alsóvonással elválasztva (soha ne egybe).

Másik megoldás lehet egy már meglévő domain megvásárlása. A keresők a rangsorolásnál a domain korát nagyon komolyan figyelembe veszik. Minél öregebb egy domain, amin működött weboldal, annál kedvezőbb lesz az elbírálás.

Viszont azt tudnunk kell domain vásárlása előtt, hogy sokat előre lefoglalnak, hogy aztán később irreális árat kérve érte később értékesíthessék azt (domain nepperkedés).

Amire még figyelniünk kell, hogy az oldalunkat ne lehessen elérni www-vel és anélkül is, irányítsuk át egyiket a másikra.

5.3.1.4 Tartalom

Tartalom megírásának is vannak szabályai keresőmarketing szempontból.

Legfontosabb, hogy az elsődleges kulcsszavainkat minél többször írjuk bele, ha lehet ragozatlan formában.

Szinonima szavaknál konzekvensen használjuk mindig ugyanazt.

Minden weboldalon egyedülálló tartalomnak kell lennie.

Ne duplikáljunk weboldalakat, vagy azok tartalmát.

Saját weboldalak duplikálása pedig tilos keresőmarketing szempontból.

Minimum 200, de 1000-nél ne több szót használjunk a weboldalon. A 200-nál kevesebb szót tartalmazó oldalakat, nem nagyon veszik figyelembe a keresők, viszont 1000 szó után már nem nagyon „olvassák” végig a tartalmat.

5.3.1.5 Weboldal struktúra, belső linkelés

Keresőoptimalizálás szempontból nem mindegy, hogy hierarchikusan, mélyen tagolt weboldalt készítünk, vagy úgynevezett lapos struktúrát alkalmazunk.

Keresőoptimalizálás és használhatóság szempontjából is fontos az úgynevezett „kattintási távolság” azaz, hogy az adott oldal mennyi kattintásnyira van a főoldaltól.

Fontos keresőmarketing szempont, hogy a weboldal jól átjárható legyen a keresők számára, magyarul eljusson mindegyik oldalra.

Ha már egy meglévő weboldalt szeretnénk átstrukturálni keresőoptimalizálás szempontjából, akkor nem szabad elfeledkezni az átirányításokról. Ha oldalunk elköltözik más URL címre, akkor a régi oldalt lehetőleg ne szüntessük meg, hanem irányítsuk át az új címre. Így a régi oldal keresőoptimalizálás szempontú eredményei átöröklődnek az új oldalra. Ilyen esetben használjunk 301-es átirányítást.

Keresőoptimalizálás szempontból fontos egy adott oldal tartalmának a helyes strukturálása is. Tagoljuk a szöveget! Használjunk címsorokat (H1, H2,..., H6). Használjunk kulcsszavakat! A szövegtörzsben a fontos kulcsszavainkat emeljük ki félkövérrel.

Ne felejtünk el címet adni a képeknek. A cím írja le lehetőleg a kép tartalmát. Lehetőség szerint ne használjunk a navigációban képi elemeket. Ha mégis használnánk, töltsük ki helyesen a kép alt tag-jét, és a képet beszédesen nevezzük el!

Készítsük sitemap-et, amit később adjunk meg a Google-nek. Ezt megtehetjük a Google Webmasters Tools -ban, amiben amellet, hogy megadhatjuk a sitemap elérését, számos nagyon hasznos funkciót kínál.

5.3.1.6 Technikai elemzés, weboldal leprogramozása

Fontos, hogy a weboldalunkat úgy alakítsuk ki, hogy a keresők jól tudják olvasni, és jól értékeljék. Vannak technikai megoldások, amik ajánlottak, és vannak amik tilosak keresőoptimalizálás szempontjából.

Hiába jó a tartalmunk, vagy mutat ránk több ezer link, ha az oldalunk keresőoptimalizálás szempontból helytelenül lett leprogramozva, egyszerűen nem fogják tudni elolvasni a weboldalunk tartalmát a keresők, és így értékelni, értelmezni sem fogják tudni azt.

5.3.1.7 Title és Meta Tag-ek megadása

Manapság a meta tag-ek jelentősége csökken, viszont a title jelentősége nő.

A title megadása tehát egy nagyon fontos lépés, és természetesen erre is vonatkoznak szabályok.

Lássunk néhány tanácsot:

- Írjunk bele kulcsszavakat.
- A title lehetőleg ne legyen több 7 szónál.
- Lehetőleg legyen minél rövidebb.
- Számít a szavak sorrendje. A legelső kapja a legnagyobb jelentőséget, és ez a vége felé csökken.

A Meta Tag-ek jelentősége gyengül, mégis hibás alkalmazásuk, vagy elhagyásuk ronthat a weboldal keresőoptimalizálás szempontú megítélésén. A meta tag-ek, mint nevük is mutatják, arra szolgálnak, hogy a látogató számára többnyire láthatatlanul tulajdonsággal lássák el weboldalunkat.

Fontos:

- A meta tag-ek legyenek összhangban az oldal tartalmával.
- A nagyobb látogatás reményében ne akarjuk megadni olyan gyakran keresett kulcsszavakat a meta tag-ekben, amikről nem is szól az oldalunk.

5.3.1.8 Regisztrálás keresőkbe

Itt arról van szó, hogy ha a kereső még nem indexálta az oldalunkat, akkor még nem ismeri. Itt megadhatjuk neki, hogy létezik az oldalunk, és vegye fel a tárolt, ismert weboldalak közé. Igaz ezt a tevékenységet előbb-utóbb automatikusan megteszik a keresők (a ránk mutató linkeken segítségével találják meg az oldalt), ha mi nem tesszük meg, a regisztrálást a keresőkbe. Lehet, hogy kicsit gyorsabb, ha mi magunk regisztráljuk a keresőkbe a weboldalunkat, de nem lényegesen nagyobb ez az idő. A regisztrálást mindenki megteheti és általában ez ingyenes maximum egy felhasználói regisztráció szükséges hozzá. Magyarországon a Google keresőmotor elérése 90-95% feletti, ezért nem érdemes más keresőbe beregisztálni a weboldalunkat.

5.3.1.9 Linképítés

Általánosan elmondható, hogy minél több külső vagy belső link mutat weboldalunkra, annál jobb.

A linkeknek 3 típusát különböztetjük meg:

- Belső link.
- Kifelé mutató link.
- Ránk mutató, külső link.

Tanácsok, észrevételek:

- Ha szöveges a linkünk, akkor használunk a hivatkozott oldalnak megfelelő szöveget, esetleg kulcsszót.
- Ha a linkünk egy kép, ne felejtsük el ALT tag-et adni a képnek, illetve magát a képfájlt is beszédesen nevezzük el, ez főleg igaz a grafikus menükre!
- Ne használjunk linkeket lenyíló menükben!
- Flash-ben készült lenyíló menü linkjei a keresőrobotoknak láthatatlanok.

Belső linkek:

Elengedhetetlen eszköz arra, hogy megmutassuk a keresőknek, az oldalainkat. A keresőrobotok linkeken keresztül haladnak, tehát a főoldalunkra érve, az ott található linkek alapján találják meg aloldalainkat, feltéve, ha ezt nem tiltottuk meg a keresőnek.

Kifelé mutató linkek

Sokat segít keresőoptimalizálás szempontjából, ha tartalmunknak megfelelő jó keresőoptimalizálással rendelkező weboldalakra hivatkozunk, nőni fog oldalunk értéke.

A mértéket itt is érdemes megtartani, 5-10 külső linknél ne legyen több egy oldalon!

Ránk mutató linkek, külső linkek

Külső linkekkel kapcsolatos észrevételek:

- Fontos, hogy a ránk mutató weboldal témája kapcsolódjon a miénkhez, magyarul a link olyan környezetbe legyen, ami a mi oldalunk tartalmához hasonló.
- Fontos hogy az a weboldal, ami ránk hivatkozik, keresőoptimalizálás szempontból értékes legyen.
- Fontos, hogy a ránk mutató link milyen szöveget használ. Ha a link szövege például a nagyon sokszor használt „tovább”, vagy „bővebben”, akkor az a „tovább” vagy „bővebben” szavakra fogja erősíteni a weboldalunkat. Használjunk kulcsszavakat a linkek szövegében, így azokat fogja erősíteni!

Page Rank

A Google a ránk mutató linkek számára és minőségére létrehozott egy mutatószámot, amelynek a Page Rank nevet adták. Ezt egy nagyon bonyolult képlet alapján számítják. A Page Rank egy kerekített szám. Ez a szám egy 10 jegyű skálán változhat. Azt mutatja, hogy mennyi és milyen minőségű oldal hivatkozik ránk, és azokra az oldalakra mennyi és milyen minőségű oldal mutat, és így tovább. A Page Rank figyelembe veszi azt is, hogy az adott oldalról, aki hivatkozik ránk, még hány link mutat.

5.3.1.10 Riport

Keresőrobotok lassan dolgoznak sajnos. A keresőoptimalizálás során elért eredményről lassan kapunk visszajelzést. Ez új oldal esetén akár több napos, hetes, sőt több hónapos folyamat is lehet. Miután nagyjából elkezdődnek a pozitív változások, a keresőoptimalizálás elején elkészített benchmarknak megfelelően riport készítünk az elért eredményekről.

Ebből a riportból világosan látszik, hogy melyek azok a keresőoptimalizálás során elér eredmények, amelyekkel elégedettek lehetünk, és melyek azok, amelyekre még rá kell dolgozni.

A keresőoptimalizálás tapasztalatok alapján működik. Lépünk egyet, és megnézzük, hogy milyen eredményeket értünk el az adott lépéssel, és az elért eredmények függvényében lépünk tovább.^[2]

5.3.2 Keresőhirdetés (PPC - Pay Per Click)

A keresőhirdetés a keresőkben megjelenő fizetett, általában klikkelés után fizetett hirdetéseket jelenti. Innen az angol mozaik szó is a PPC, mely a „Pay Per Click” kifejezésből ered, ami magyarul annyit jelent, mint „kattintás alapú fizetés”.

Keresőhirdetés alatt azokat a szöveges, képi, hirdetéseket értjük, amelyek többnyire a keresőkben jelennek meg. Többnyire kattintás után kell fizetni, de lehet megjelenés után is fizetni (régábban eladás után is lehetett fizetni).

Több keresőhirdetés is létezik ma Magyarországon.

Két legnagyobb a Google Adwords, a másik az ETARGET(olyan nagy tartalomhálózati oldalakon jelennek meg, mint például az origo.hu, index.hu, iwiw.hu, lap.hu.). Mindkét keresőhirdetés rendszer licit alapú, tehát licitálnunk kell az egyes kattintásokért.

Sokan egyszerűnek gondolják a keresőhirdetések futtatását, de az esetek többségében azonban érdemes szakemberhez fordulni.

5.3.3 Fizetett listázás (paid inclusion)

Olyan keresőmarketing eszköz, amelynél a keresőmotor díjazás ellenében szerepelteti az adott honlapot a találati listában. Sok kereső lehetőséget ad arra, hogy a cégek megvásárolják a találati listában való szereplésüket, a Google ezt azonban elutasítja, hogy minél relevánsabb eredményeket tudjon nyújtani. A fizetett listázás esetén tipikus fizetési mód az éves díj, de előfordul a kattintás utáni fizetés is. Leggyakoribb azonban a vegyes elszámolás alkalmazása.^[12]

6. Összefoglalás

6.1 Mikről esett szó az előző fejezetekben?

Az előző fejezetekben többféle szempont szerint végigjártam a web design-el kapcsolatos dolgokat.

Gondolok itt arra, hogy milyen technológiák vannak. Ezekből kettő nyelvet, ha röviden is bemutatam (HTML, CSS).

Milyen szoftvert lehet (szoftvereket) lehet használni honlaptervezésre. Ezekből a szoftverekből röviden egyet bemutatam (NetObject Fusion).

Milyen szempontok jöhetnek szóba akadálymentesítés és ergonómiai szempontból. Ezeket a szempontokat sorra vettem.

Szóba kerültek a honlapon használható grafikai elemek és színek. Milyen különbségek vannak a raszteres és a vektorgrafika között. Mind a kettőre említettem egy-egy szoftvert (CorelDraw és Photoshop) is, ami ezeket használja.

Végül, de nem utolsó sorban foglalkoztam a webes marketinggel is.

6.2 Mi volt a célom?

Célom volt, hogy a fenti dolgokba egy kis betekintést nyújtsak, és pár dologra felhívjam a figyelmet. A kitűzött céljaimat szerintem elértem. Bevezettem az olvasót sok web design-hoz köthető témákba. Látható vált milyen összetett témáról is van szó és mennyi minden tartozik ebbe a témakörbe.

Jelenlegi szakdolgozati kööttségek miatt most ennyi fért bele.

Később tervezem, hogy bővítem a technológiai részt a JavaScriptekkel, PHP-val és egy részletesebb Photoshop bemutatással is.

Mint minden, ami a számítástechnikával kapcsolatos, ez a terület is gyorsan változik.

Új technológiák és eszközök jelennek meg és terjednek el, vagy a régebbieknek egy frissebb változata, de az már majd egy másik szakdolgozat témája lehet.

Bővebben vizsgáltam a szakdolgozatom keretei között egy web design szoftvert is.

Tapasztalataim alapján most ennek a szoftvernek térnék ki a fejlesztési lehetőségeire. Ezeket a program hibáin keresztül mutatnám be.

6.3 Fejlesztési lehetőségek a vizsgált szoftverben (NetObject Fusion)

Itt felsorolnám milyen problémákkal talákoztam honlap fejlesztése közben NetObject Fusion környezetben. Egyúttal ezeknek a területeknek a fejlesztését javasolnám a program fejlesztőinek:

- Képek átméretezése

Voltak és vannak ezzel a témával gondjai a programnak.

A 9-es verzióban nem volt erre beépített lehetőség. A fejlesztők ezt láthatták, mert ezt a hiányosságot már sikerült nagyrészt kiküszöbölni a 11-es verzióban, de még nem tökéletesen^[*].

* Megjegyezném, hogy ha túl nagy felbontású képeket teszünk be az oldalunkra, akkor a program bizonyos esetekben lefagyhat.

- Globális beállítások hiánya

Nem lehet globális beállításokat megadni a programban

Ez is elég zavaró. Például jó lenne, ha az oldalak háttérét nem egyesével, hanem globálisan is be lehetne állítani.

Ezt a hiányosságot orvosolhatjuk CSS használatával.

- A beépített stíluslapokat csak elég korlátozott méretben lehet módosítani a programban.

Legalább is a programba beépített keretek között.

Természetesen a CSS kódba manuálisan bele nyúlhatunk, de ez időigényes.

- Néha nem teljesen azt adja kimenetként, mint amit a fejlesztőkörnyezetben lehet látni. /Újabb technológiák támogatásának hiánya/

Gondolok itt arra, hogy bizonyos grafikai elemek nem ott és úgy jelennek meg, ahogy azt látni lehet a holnaptervező részben.

Ez adódhat persze a sokfajta böngésző különböző képességéből és adódhat továbbá abból is, hogy nincs felkészítve a program a legújabb technikákra, mivel a legutolsó verzió még 2008-ban jelent meg.

- Fix méretű oldalak fejleszthetők vele.

Köszönetnyilvánítás:

Ezúton szeretnék köszönetet mondani témavezetőmnek, Vágvölgyi Csabának szakdolgozatom elkészítésében nyújtott segítségéért, útmutató rendkívül értékes szakmai tanácsaiért és türelméért.

Köszönettel tartozom belső témavezetőmnek Dr. Zichar Mariann-nak ötleteiért és segítségéért.

Ugyanígy köszönettel gondolok munkahelyemre — a National Instruments Európa Kft-re —, mert biztosította a tanulmányaim folytatásához szükséges pénzügyi alapot, és azt a lehetőséget, hogy tanulhassak.

Irodalomjegyzék:

Felhasznált könyvek

1. Virginia DeBolt: HTML és CSS webszerkesztés stílusosan.
2005 Kiskapu Kft.

Internet, hivatkozások

Internet háttéréhez felhasznált irodalom:

1. http://www.inf.unideb.hu/~bodai/internet/internet_tortenete.html
2. <http://jakab.fellner.sulinet.hu/jaki/oktatas/internet.htm>
3. <http://hu.wikipedia.org/wiki/Internet>
4. <http://hu.wikipedia.org/wiki/Vil%C3%A1gh%C3%A1ll%C3%B3>

Technológia részhez felhasznált irodalom:

1. <http://www.standardsmode.hu/>
2. <http://hu.wikipedia.org/wiki/HTML>
3. <http://mek.niif.hu/01300/01306/html>
4. http://hu.wikipedia.org/wiki/HTML-szerkeszt%C5%91_programok

Akadálymentesítés, ergonómia részhez felhasznált irodalom:

1. <http://weblabor.hu/cikkek/akadalymentesweb1>
2. <http://weblabor.hu/cikkek/akadalymentesweb2>
3. http://dsd.sztaki.hu/people/mate_pataki/200803_Networkshop_Akadalymentesites.pdf
4. <http://www.w3c.hu/szolgaltatasok/miertkellakadalymentesiteni.html>
5. <http://mek.niif.hu/01200/01224/html/index.htm>
6. http://www.w3c.hu/szereplesek/2006/Pataki_Mate-Weblapok_akadalymentesitese.html
7. <http://bartal.org/web-accessibility-akadalymentes-weboldalak.html>
8. <http://gevapc.hu/szintevestok>
9. <http://www.marketing4u.hu/akadalymentes-honlapkeszites/weblodal-vakok-gyengenlatok-szamara.html>
10. <http://www.w3.org/TR/WCAG10/full-checklist.html>
/angol/
11. <http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/>
/angol/
12. <http://www.w3c.hu/forditasok/WCAG20/>
13. <http://validator.w3.org/>
14. <http://www.sitemarketing.hu/Keresooptimalizalas/Website-ergonomia-hatekony-honlap.html>
15. <http://www.online.hu/web-ergonomiai-tervezes>
16. http://ergomania.blog.hu/2009/02/20/ergonomiai_tipushibak_a_magyar_weben_2009_ben
17. <http://blog.hdesign.cc/2009/06/ergonomia-a-weben-usability/>

18. <http://www.senpai.hu/website-ergonomia-szakcikkek>
19. <http://www.pszichologia.hu/cikk/cikk.phtml?id=39>
20. <http://www.senpai.hu/toltelek-szovegek-a-weben>
21. http://www.carnation.hu/hirl_cikk.php?id=47&cid=1
22. http://webni.innen.hu/J_c3_b3lMeg_c3_adrtWeboldal
23. <http://webakademia.hu/2007/10/a-gyorsabb-weblapert/>
24. <http://www.human-web-bt.com/akadalymentes-weboldal.php>
25. <http://w3c.hu/>

Grafika, megjelenés részhez felhasznált irodalom:

1. http://www.sikermarketing.hu/szinek_fotok.html
2. <http://www.colorschemer.com/blog/2007/07/24/140-named-colors/>
3. <http://opencsourcedtemplates.org/>
4. <http://hu.wikipedia.org/wiki/SVG>
5. <http://www.oswd.org/>
6. <http://hu.wikipedia.org/wiki/JPEG>
7. <http://webbuilder.hu/index.php/JPEG>
8. <http://webbuilder.hu/index.php/GIF>
9. <http://hu.wikipedia.org/wiki/GIF>
10. http://help.adobe.com/hu_HU/InDesign/5.0/help.html?content=WSa285fff53dea4f8617383751001ea8cb3f-6bce.html
11. <http://hu.wikipedia.org/wiki/Bmp>
12. <http://www.demoscene.hu/trajic/multi/kepek.htm>
13. <http://www.tferi.hu/konyv5/fej23a.html>
14. http://logout.hu/cikk/kepformatumok_mindenkinek/teljes.html
15. <http://office.microsoft.com/hu-hu/help/HA010563051038.aspx>
16. http://www.manta.hu/szotar/internet_ert.htm
17. http://ecdweb.hu/index.php?title=Grafikus_f%C3%A1jlform%C3%A1tumok
18. <http://www.webtemplatemall.com/hu/index.html>
19. http://www.prosign.hu/lap/cikk/szin_2_csoportositas.htm
20. <http://hu.wikipedia.org/wiki/Photoshop>

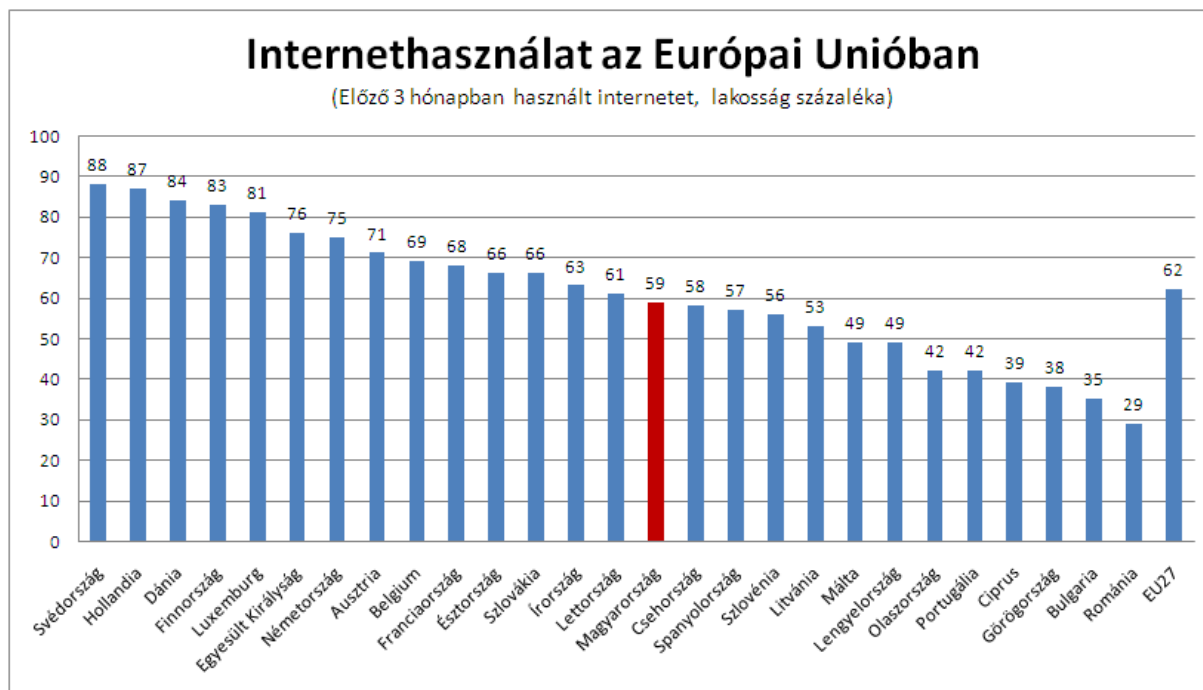
Marketing részhez felhasznált irodalom:

1. http://www.html-weboldal-retro.hu/web_marketing.html
2. <http://www.keresomarketing.eu/>
3. <http://www.qualitycert.hu/web/>
4. http://www.webma.hu/Keresomarketing_tudasbazis.html
5. <http://www.seotools.hu/>
6. http://webni.innen.hu/Keres_c5_91optimaliz_c3_a1l_c3_a1s
7. <http://www.keresooptimalizálás.net/>
8. <http://www.net-position.hu/megoldasaink/latogatottsag-noveles/>
9. <http://www.net-position.hu/megoldasaink/keresooptimalizalas/>
10. <http://weblabor.hu/cikkek/keresooptimalizalas>
11. <http://hu.wikipedia.org/wiki/Keres%C5%91optimaliz%C3%A1l%C3%A1s>
12. <http://hu.wikipedia.org/wiki/Keres%C5%91marketing>

Függelék:

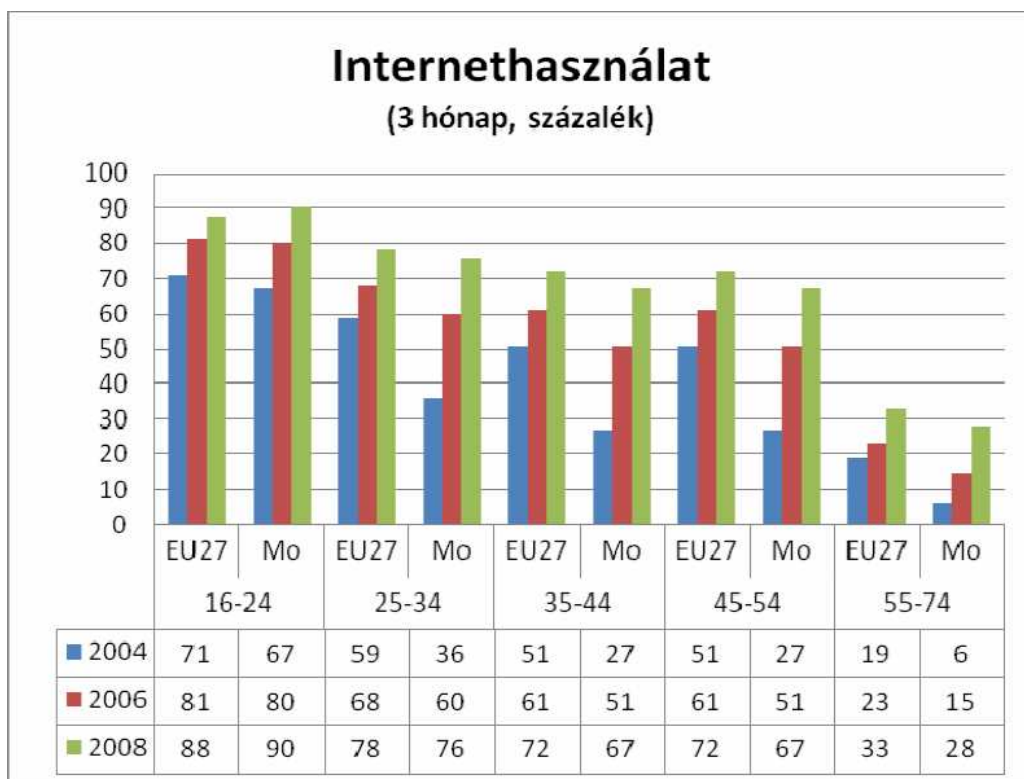
Internetes bevezetőhöz kapcsolódó adalék anyagok:

1. ábra: Az internet-használat a 16-74 éves korcsoportban (2008)



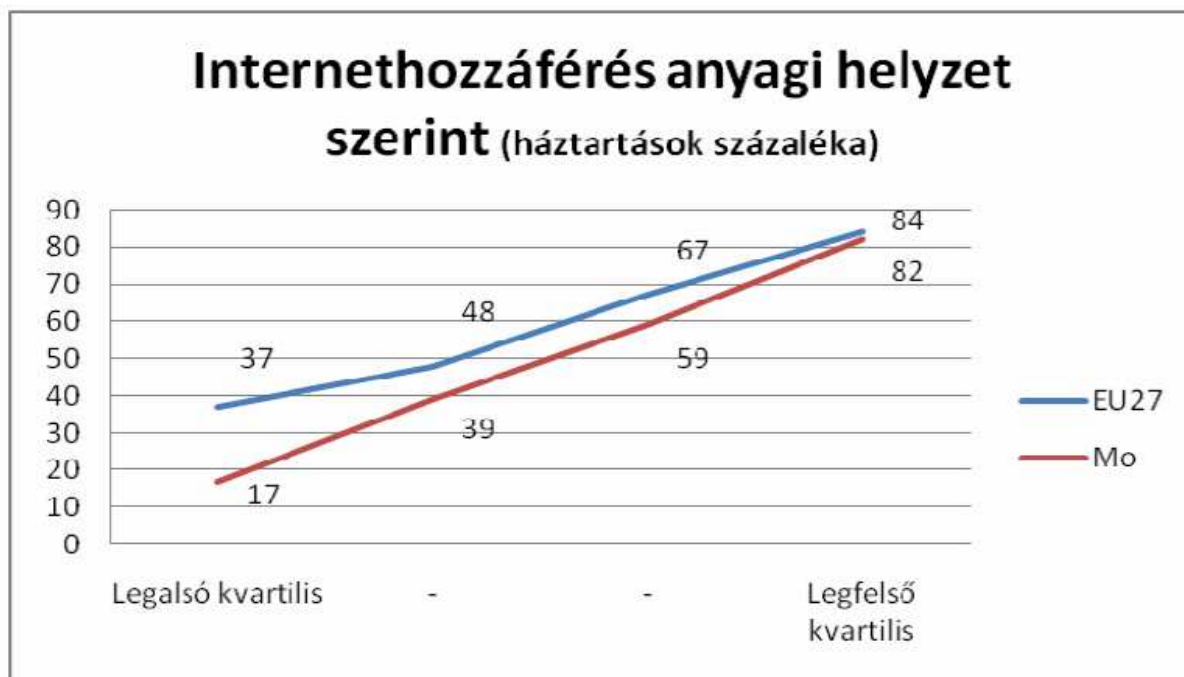
Forrás: Eurostat, 2008

2. ábra: Internethasználat az EU 27 és Magyarország viszonylatában



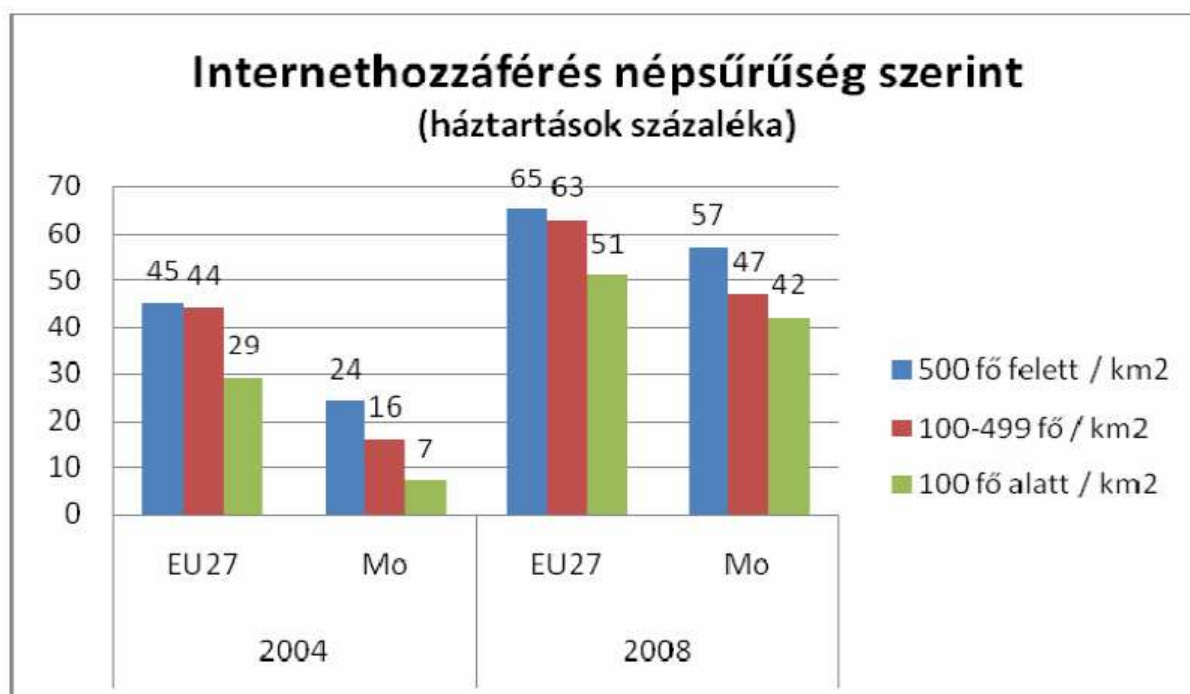
Forrás: Eurostat, 2009

3. ábra: A háztartások internetellátottsága jövedelmi csoportok szerint (2008)



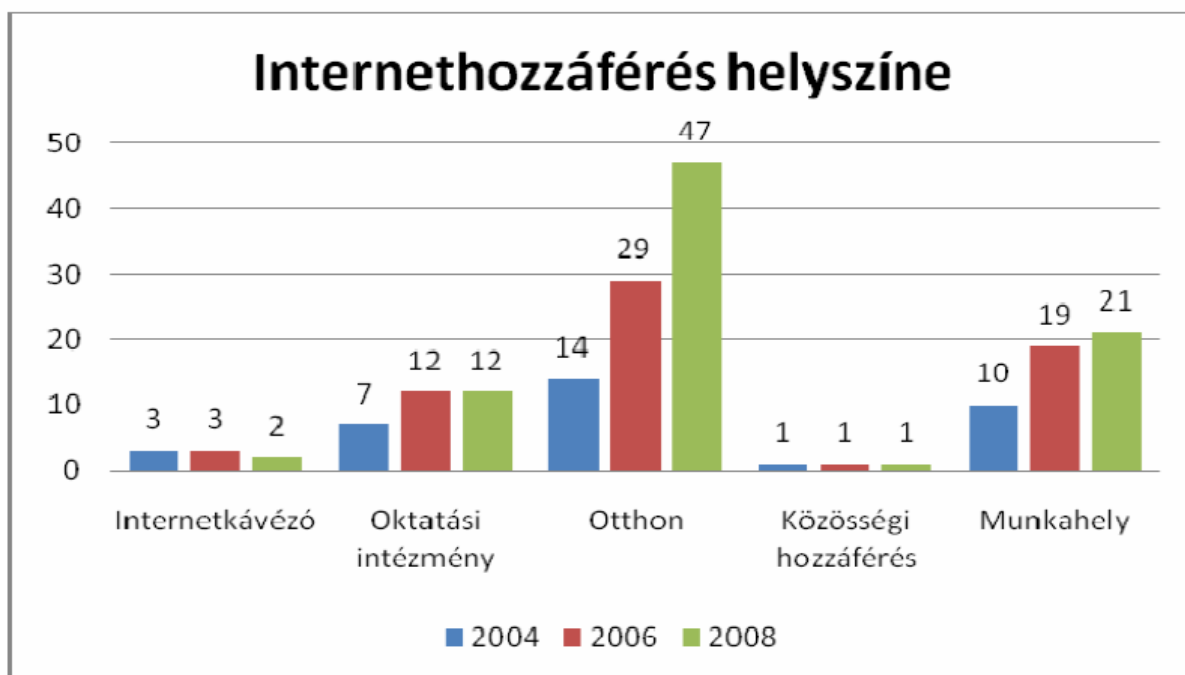
Forrás: Eurostat, 2009

4. ábra: A háztartások internetellátottsága népsűrűség szerint (2008)



Forrás: Eurostat, 2009

5. ábra: Az internet-hozzáférés leggyakoribb formái és a közösségi elérés (16 évesnél idősebb teljes lakosság százaléka)



Forrás: Eurostat, 2009

HTML utasítások rövid összefoglalója:

<!--: Megjegyzés (Description)
<!DOCTYPE: HTML-dokumentum típusának leírása (DOCument TYPE)
A: Kapcsolat - élő link (Active link)
ACRONYM: Acronim beszúrása (ACRONYM)
ADDRESS: Postacím - kiírásmód (postal ADDRESS)
APPLET: Futtatható elem helye (APPLET)
AREA: Kliens-oldali térkép helyének meghatározása (AREA)
B: Félkövér betű (Bold)
BANNER: Címszalag - nem scrollozható (Banner)
BASE: A dokumentum báziscímének meghatározása (BASE)
BASEFONT: Alapértelmezett betűkészlet (BASic FONT)
BGSOUND: Háttérhang (BackGround SOUND)
BIG: Nagyobb betűméret (BIGger font size)
BLINK: Villogó betűk (BLINK)
BLOCKQUOTE: Szövegblokk (BLOCK QUOTE)
BODY: Az egész dokumentum törzsrésze - NEM táblázaté (BODY)
BR: Sortörés (BReak)
BUTTON: Nyomógomb (BUTTON)
CAPTION: Táblázat címe (CAPTION)
CENTER: Középre helyezés (CENTER)
CITE: Idézet kiírása (CITE)
CODE: Programkód kiírásmód (CODE)
COL: Oszlop egy táblázatban (COLoumn)
COLGROUP: Oszlopcsoport (COLoumn's GROUP)
COMMENT: Megjegyzés (COMMENT)
DD: Definíció leírása (Definiton Description)
DEL: Törölt szöveg (DELETED text)
DFN: Definiált kiírásmód (DeFiNition)
DIR: Felsorolás-szerű lista készítése (DIRectory)
DIV: Szöveg tagolása (DIV)
DL: Definíciós lista (DL)
DT: Definíciós leírás (Definition Term)
EM: Kiemelt írásmód (EMphasize Text)
EMBED: Dokumentum beszúrása egy másikba (EMBED)
FIELDSET: Mezőelem beszúrása (FIELD SET)
FONT: Betű tulajdonságainak szabályozása (FONT)
FORM: Kérdőívek létrehozása (FORM)
FRAME: Keret beállítása (FRAME)
FRAMESET: Keretezés indítása (FRAMESET)
H1: Fejezetcím - a legnagyobb értékű (Heading 1)
H2: Fejezetcím - 2. legnagyobb (Heading 2)
H3: Fejezetcím - 3. legnagyobb (Heading 3)
H4: Fejezetcím - 4. legnagyobb (Heading 4)
H5: Fejezetcím - 5. legnagyobb (Heading 5)
H6: Fejezetcím - legkisebb (Heading 6)
HEAD: Az egész dokumentum fejléce - NEM táblázaté (HEADer)
HR: Vízszintes vonal (Horizontal Rule)
HTML: Az egész dokumentum kerete (HyperText Markup Language)
I: Döntött betű (Italic)
IFRAME: Lebegő keretek definiálása (In-line floating FRAMEs)
IMG: Kép vagy videó beszúrása (IMaGe)
INPUT: Beviteli mező űrlapokon (INPUT field)
INS: Hozzáadott szöveg beszúrása (INSert text)

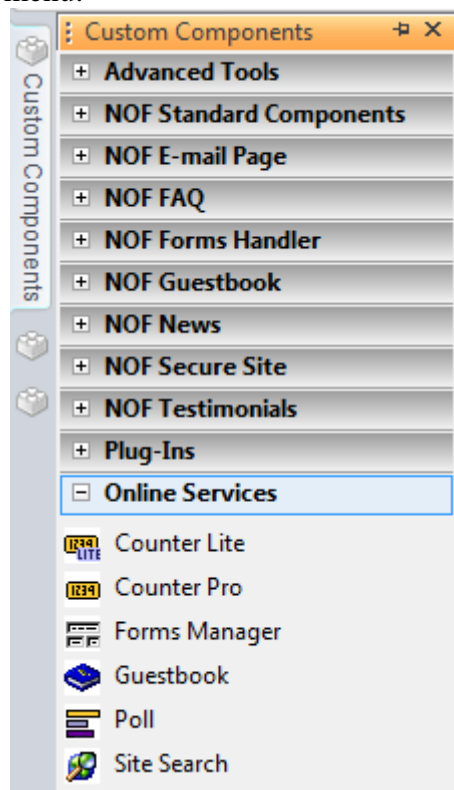
ISINDEX: Jelzi a keresőknek, hogy a lap indexelhető (ISINDEX)
KBD: Billentyűzetről bevitt adat (KeyBoarD)
LABEL: Címke definiálása (LABEL)
LEGEND: Megadja a FIELDSET elemét (LEGEND)
LI: Lista elem (List Item)
LINK: Kapcsolat más dokumentummal (LINK)
LISTING: Előre listázott szöveg (LISTING text)
MAP: Térkép grafikus elemekhez (MAP)
MARQUEE: Szöveg mozgatása (scrolling MARQUEE)
MENU: Listák menü-szerű felsorolása (MENU)
META: Nem megjelenő, de igen fontos elem (META tag)
NOBR: Sortörés letiltása (NO BReaking)
NOFRAMES: Keretek tiltásakor megjelenő adat (Non-FRAMES version)
NOSCRIPT: Scriptek megjelenítésére alkalmatlan böngészők számára írt rész (Non-SCRIPTing version)
OBJECT: Objektum beszúrása (OBJECT insert)
OL: Rendezett/sorszámozott lista (Ordered List)
OPTION: Kiválasztásos lista eleme kérdőíveknél (OPTION)
P: Paragrafus/bekezdés (Paragraph)
PARAM: Változók/paraméterek meghatározása más elemekhez (PARAMeters)
PLAINTEXT: Formázatlan szöveg (PLAIN TEXT)
PRE: Előre formázott szöveg (PREformatted)
Q: Kiemelt szöveg (Quality text)
S: Keresztülhúzott szöveg (Strikethrough)
SAMP: Irodalmi szöveg (SAMPle text)
SCRIPT: Más nyelven írt kódrészlet (script) beszúrása (SCRIPT)
SELECT: Kérdésre való kiválasztási lehetőség kérdőíveknél (SELECT)
SMALL: Kisebbségi betűméret (SMALLer font size)
SPAN: Érvényességi kör (SPAN)
STRIKE: Keresztülhúzott szöveg (STRIKEthrough text)
STRONG: Vastagított szöveg (STRONGer text)
STYLE: Stílus definiálása egy lapon belül (STYLE) - Csak a folytatásban szerepel!
SUB: Alsó index kiírás (SUBscript)
SUP: Felső index kiírás (SUPerscript)
TABLE: Táblázat (TABLE)
TBODY: Táblázat törzse (Table BODY)
TD: Táblázat adata (Table Data)
TEXTAREA: Hosszabb szöveg beviteli mező kérdőívekben (TEXTAREA)
TFooter: Táblázat lábléce (Table FOOT)
TH: Táblázat fejléce (Table Head)
THEAD: Táblázat fejléce (Table HEAD)
TITLE: Dokumentum címe - NEM táblázaté (TITLE)
TR: Táblázat sora (Table Row)
TT: Írógépszerű kiírás (TeleType)
U: Aláhúzott betű (Underlined)
UL: Nem rendezett lista (Unordered List)
VAR: Változónév (VARiable)
WBR: Sortörés engedélyezése letiltott területen. (WBR)
XMP: Mintaszöveg (eXaMPle)

NetObject Fusion ismertetéséhez kapcsolódó adalék anyagok:

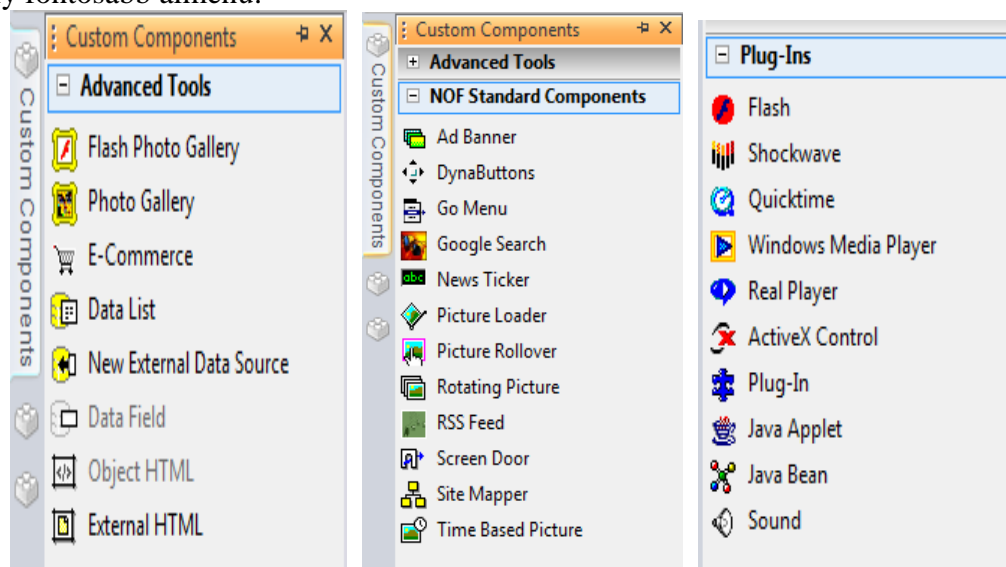
További fontosabb elemek a programból, amire a keretek szűkössége miatt nem tértem ki:

- „Custom Components” elemei

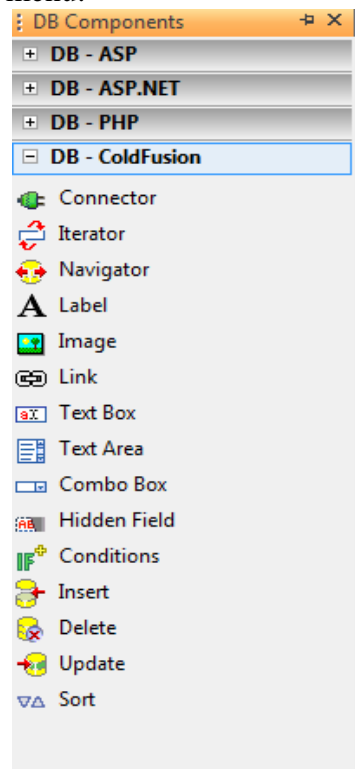
Teljes menü:



Néhány fontosabb almenü:



- „DB Components” elemei
Teljes menü:



- „Flash Components” elemei
Teljes menü:

