

# A MATARKA akadálymentesítése

„Mert közös a világunk: nincs külön világ az épek és egy másik a fogyatékossgal élők számára, egyetlen világ van, s valamennyiünknek abban kell élnünk, emberséges békeességben”. [1]

**A könyvtári szolgáltatások, azon belül is az interneten keresztül elérhető szolgáltatások akadálymentesítését nem írják elő kötelező jelleggel a könyvtárakra vonatkozó törvények, rendeletek. Megemlítendő azonban, hogy a Könyvtárellátási Szolgáltató Rendszer tagjai számára a 39/2013. (V. 31.) EMMI rendelet a könyvtári szolgáltatásokról szóló információk biztosítását írja elő a kistélepülési önkormányzatok honlapján a fogyatékkal élők tájékoztatására. Az adatbázisok, honlapok akadálymentesítése viszont jogszabályi szinten csak az elektronikus közszolgáltatások számára kötelező. De a könyvtárak mégis mindent megtesznek annak érdekében, hogy honlapjaikat, webes adatbázisaikat minél több felhasználó elérje, köztük a vakok, gyengénlátók, mozgásukban korlátozottak is. A MATARKA Egyesület is úgy látta jónak, hogy támogatásból megvalósítja a MATARKA szolgáltatás akadálymentesítését. A 2015 évi Networkshopon Sárospatakon elhangzott, erről szóló előadás cikk formába öntött, kibővített és átszerkesztett változatát tartja most kezében az olvasó.**

### Általános bevezető

A MATARKA ([www.matarka.hu](http://www.matarka.hu)) 2002 óta létező, a Miskolci Egyetem, Könyvtár, Levéltár, Múzeum által fejlesztett és működtetett, túlnyomórészt Magyarországon megjelenő folyóiratok tartalomjegyzékeinek kereshetőségét és böngészését lehetővé tevő, bárki számára térítésmentesen elérhető

szolgáltatás (1. ábra). Alapfunkcióin túl elvezet a feldolgozott folyóiratok honlapjára, hozzáférést ad ugrópontok segítségével a cikkek teljes szövegéhez, ha azok szabadon elérhetők az interneten, illetve ha nem, akkor közvetítésével a cikkek másolata megrendelhető az Országos Széchényi Könyvtártól.

The screenshot shows the MATARKA website interface. At the top, there is a navigation bar with 'Keresés', 'Index', 'Folyóiratok', 'Kosár', and 'Súgó' buttons. Below this is a search bar with a 'Keresés' button. The main content area is titled 'MATARKA' and 'Röviden a szolgáltatásról'. It contains a detailed description of the service, stating that it is a free service for searching and browsing the content of journals published in Hungary, developed and operated by the Miskolc University Library, Archives, and Museum. The description mentions that the service is available to anyone, and that the database contains over 35 libraries, 30 journals, and 8 magazines. It also states that the service is available on the internet, and that the service is free of charge. At the bottom of the page, there are logos for 'ISZT', 'SZÉCHÉNYI TERTY', and 'nka'.

1. ábra A MATARKA szolgáltatás kezdő oldala

A MATARKA ma már évente majd másfél milliónyi használatot regisztrál. Az adatbázist használják az MTMT-be szerzői bibliográfiák átemeléséhez, adatait átveszi az EHM (EPA-MATARKA-HUMANUS) közös kereső és az ODR (Országos Dokumentum Ellátási Rendszer), az ELDORADO rendszer, illetve a most alakuló Pécsi Egyetem által fejlesztett hivatkozásokat feldolgozó adatbázis is.

Mivel országos szolgáltatásról van szó, néhány éve felmerült a honlap akadálymentesítésének gondolata. 2014 tavaszán a 2006 óta létező MATARKA Egyesület sikeresen pályázott az *Internet Szolgáltatók Tanácsánál* és a támogatás egy részét szoftverfejlesztésre fordította, mely egyrészt az akadálymentes lekérdező felület létrehozását célozta meg, másrészt a közösségi tárgyszavazás kialakítását, rövidebben a címkézést. Az alábbiakban az akadálymentesítés tárgyalása következik, a címkézés szoftveres megoldása megszületett, de a tényleges munka várát még magára (kidolgozás, meghirdetés), ezért tapasztalatok híján a címkézés témaköre egy következő cikk tartalma lehet.

### Akadálymentesítés a könyvtárakban

Könyvtárak esetében az akadálymentesítés egyrészt az épületek, másrészt a szolgáltatások akadálymentesítését jelenti a fogyatékkal élő emberek számára. Bővebben kifejtve ide tartozik a fizikai hozzáférés (parkolók, rámpák, liftek, mosdók, polcok, feliratok stb.), az intellektuális hozzáférés (Braille-írás, öregbetűs könyvek, képernyő-felolvasó szoftverek, hangoskönyvek stb.), a virtuális hozzáférés (honlapok akadálymentesítése) és nem utolsósorban a könyvtári dolgozók felkészítése a fogyatékkal élők kiszolgálására.

### Honlapok elméleti akadálymentesítése

Cél az, hogy a weboldalakat minden felhasználó ugyanúgy használhassa. Amikor az akadálymentesség (angol nevén: accessibility) szóba kerül, legtöbbször a testi fogyatékosokra gondolnak az emberek, de az akadálymentes oldalak készítésekor több felhasználó igényeit kell figyelembe venni.

Érintett csoportok:

- a vakok, gyengénlátók, színvakok és szintévesztők,
- testi fogyatékkal élők – gondot okozhat nekik az egér vagy a billentyűzet használata,
- mobil eszközök felhasználói – nagyon kicsi a megjelenítője,

- régi rendszert használók – például anyagi okokból nem térhetek át korszerű szoftverekre, új számítógépekre,
- a fiatal (5–10 éves) internethasználók – esetükben a szem-kéz koordináció és az olvasás okozhat problémát,
- az idősek – a gyengébb látás, korlátozott mozgás és gyengébb szem-kéz koordináció miatt lehetnek érintettek,
- azok a felhasználók, akiknek az írott szöveg értelmezése nehézséget okoz, illetve nem beszélik folyékonyan azt a nyelvet, amelyen egy adott weboldal íródott.

Ezek a csoportok bár kicsik egyenként, és külön-külön sem generálnak komoly látogatottságot, de együtt már számuk jelentős, és egy közepes forgalmú weboldalon már számottevő lehet azoknak a száma, akik azért választják a konkurenciát (vagy tesznek panaszt egy állami szerv weboldala esetében), mert az oldal tartalmához nem férnek hozzá.

Egy akadálymentes weboldal pozitív hatásai:

- Az akadálymentes weboldalak – strukturált felépítésüknek köszönhetően – a webes keresőkben jobb helyezést érnek el.
- Pozitív PR. A weboldal tulajdonosa hangozthatja társadalmi felelősségvállalását és a szabványoknak való megfelelését.
- A weboldal tulajdonosa olyan piaci csoportokat érhet el, melyek még kevésbé telítettek – a látogatók száma növekszik.

Egy nem akadálymentes weboldal negatív hatásai:

- Bár a jogi érvelés önmagában (ma és itt) hatástalan – Magyarországon még nem fordult elő, hogy emiatt akár egy céget is bepereltek volna –, a hatályos törvények szerint tilos bárminemű hátrányos megkülönböztetés. A magyar közigazgatási honlapok egységesítését célzó *Kormányzati Informatikai Egyeztető Tárcaközi Bizottság (KIETB)* ajánlása előírta 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.

A jövőben – várhatóan – egyre több szervezet igényli majd, hogy honlapja akadálymentes legyen. A kormányzati, egészségügyi és oktatási portáloktól elvárható, hogy megfeleljenek az akadálymen-

tes oldalakra vonatkozó ajánlásoknak (jellemzően WAI-AA). Alapelvként megfogalmazható: törekedni kell arra, hogy az átlagos felhasználók számára készült oldalakat a gyengénlátók, illetve egyéb fogyatékosokkal bírók is ugyanúgy használhassák, azaz kerülendő az „ugrás az akadálymentes változatra” szemlélet. Ha ez nem lehetséges (komplex tartalom miatt, vagy a weboldal tulajdonosának kifejezett kérésére), olyan alternatív megjelenés készíthető, amely az eredeti tartalmat hiány nélkül visszaadja. Sokszor éppen az ilyen „vakbarát” változatok lecsúszított volta miatt nem használják őket a vakok és gyengénlátók. [2]

A W3C nemzetközi szervezet szabványosította a honlapok akadálymentességére vonatkozó kritériumokat. Nemzetközi szinten is, és Magyarországon is a legtöbb pályázatban, állami megrendelésben a W3C által létrehozott *Web Akadálymentesítési Útmutató (WCAG)* [3] (angolul: *Web Content Accessibility Guidelines (WCAG) 2.0*) szabványra hivatkoznak, és annak betartását követelik meg.

„A szabvány az akadálymentességnek három szintjét különbözteti meg. A legalacsonyabb az A szint, ezt mindenképpen be kell tartania egy akadálymentes honlapnak. Magasabb szintű hozzáférhetőséget, akadálymentességet határoz meg az AA (két A) szint. Az AA szint elérése legtöbbször ajánlott, főleg közérdekű, állami, önkormányzati honlapok esetében. A legszigorúbb az AAA szint, ennek az elérése már komoly követelményeket támaszt a honlap készítőivel és üzemeltetőivel szemben.” [4]

A WCAG útmutatónak 12 irányelve van, melyek 4 nagy alapelv köré csoportosíthatók. A weboldalak szerkesztőinek az egyes irányelvekhez tartozó teljesítési feltételeket, az ajánlott technikákat és módszereket kell tanulmányozniuk, illetve alkalmazniuk. A 4 nagy alapelv:

- Észlelhetőség. Ez az irányelv azzal foglalkozik, hogy hogyan legyen a honlap tartalma mindenki számára egyaránt észlelhető.
- Működtethetőség. A második irányelv azt jelenti, hogy hogyan legyen egy honlap mindenki számára jól használható.
- Érthetőség. Ez az irányelv azt írja le, hogy hogyan lehet a tartalmat mindenki számára érthetővé tenni.
- Robusztusság. Ez az irányelv pedig arról szól, hogy hogyan lehet egy honlapot más programokkal kompatibilissé tenni.

## Honlapok gyakorlati akadálymentesítése

A gyakorlati akadálymentesítés és a szabványok (W3C WCAG 2.0) szerinti akadálymentesítés nem esik egybe. Ugyanis a szabvány nem veszi figyelembe azt, hogy a technikai eszközök hogyan követik a változásokat. Ha a szabvány szerinti akadálymentesítést választja a fejlesztő, akkor a célzott felhasználók kb. fele számára még mindig elérhetetlen lenne a MATARKA, hiszen idő kell, míg a képernyőolvasóknak az a generációja, amely már teljes körűen képes a szabványok szerint akadálymentesített weblapok „feldolgozására”, általánosan elterjedtté válik. Ugyanis a képernyőolvasók a fajlagosan legdrágább technikai eszközök. Így a célcsoport tagjai sokáig használnak egy régi verziót.

## A képernyőolvasókról látóknak

A képernyőolvasók azok a programok, amelyeknek segítségével a látásukban, vagy az olvasott információ megértésében korlátozott emberek (továbbiakban, az egyszerűség és a rövidség érdekében „vakok”) a számítógépet kezelhetik. Ezek a programok a képernyő vizuális világát hallható információkká alakítják át és a program használói a hallott információk alapján a billentyűzet segítségével képesek a számítógéppel való interakcióra.

Az utahi egyetem akadálymentességgel foglalkozó intézete évente készít felmérést vakok, gyengénlátók körében arról [5], hogy milyen segédeszközöket használnak és milyenek a netezési szokásaik, illetve arról, hogy szerintük mely akadálymentességi dolgok a leghasznosabbak. E szerint a felmérés szerint a JAWS (*Job Access With Speech*) nevű képernyőolvasó még mindig jelentősen vezet a listán. A felhasználók közel 70%-ának ez az elsődleges ernyőolvasója. Ez egy fizetős program, melynek ára 1 500 USD körül van! Míg egy jogtiszt Windows már 50 USD-ért is megvásárolható. Ezt követi az NVDA (*Non Visual Desktop Access*) nevű ingyenes, nyílt forráskódú program, amely tudásában messze elmarad a JAWS-tól. Ennek részaránya kb. 10%. A többi program ezeket követi, és összesített arányuk a 20%-ot sem éri el. Az idén elkészült ennek a felmérésnek a magyar változata is és abból kiderül, hogy a magyar felhasználóknál is jellemző az, hogy akár 5–6 évig is használnak egy-egy verziót, holott a *Freedom Scientific* – a JAWS program fejlesztője – évente ad ki friss verziókat. Mivel a

Freedom Scientific cégnek a magyar piac jelentéken, így központilag nem jelenik meg magyar verzió. Ezt az *Informatika a Látássérültekért Alapítvány* készíti, de a szűkös pénzügyi lehetőségek, illetve az emberi erőforrások hiánya miatt, ezt nem tudja megtenni évente, így – általában – 3–4 verzióként jelenik meg magyar JAWS. Eddig a 4-es, a 6-os, a 11-es és a 14-es JAWS lett magyarítva. Ebből egyedül a JAWS 14.0 képes a HTML 5 szabvány szerint akadálymentesített weboldalak teljes körű feldolgozására. Ezt a verziót ma még csupán a magyar felhasználók 1/3-a használja!

A képernyőolvasó programok – programozás-technikailag – nem tekinthetők másnak, mint információgyűjtő, -átalakító és beavatkozó rendszereknek. Ennek megfelelően moduláris felépítésűek. Az egyes modulok az operációs rendszer és a hardver közötti réteggként képzelhetőek el. Általában a következő modulokból állnak:

– *Információgyűjtő, -továbbító modul*

Ez felel azért, hogy a számítógépen a látók által érzékelhető információkról adatokat gyűjtsön be. Ez a réteg az operációs rendszer mélyén, grafikus rendszerek esetében, annak ún. üzenetkezelő alrendszerében működik, abba épül be, azzal folytat folyamatos kommunikációt. Ugyanis, mikor a felhasználó valamiféle beavatkozást végez, például rákattint egy gombra, amely előzőleg nem volt fókuszban, ott több, egymást követő rendszeresemény játszódik le, melyről az operációs rendszer üzenetkezelő alrendszere kap értesítést és ez informál mindent a bekövetkezett eseményekről, hogy azokra válaszreakciók születhessenek. A fenti esetben, ha rákattintunk egy gombra, akkor körülötte – a Windows esetében – megjelenik egy szaggatott vonalból rajzolt keret, az ún. fókuszkere, illetve lefut az a program, amely az adott gombhoz hozzá lett rendelve. Ezt egy látó a látásával érzékeli, de egy vak erre nem képes. Ő csak a hallott információkat tudja feldolgozni. Tehát ezekről az eseményekről valamiféle hallható információt kellene generálni. Ezt ez a rendszer teszi meg. Jelen esetben a következő szöveges „üzenet” kerül előállításra: „Gomb felirata” gomb. (Ha a gomb felirata az, hogy „Alkalmaz”, akkor az „alkalmaz gomb” üzenet.) Ez a memóriában legenerált szöveges üzenet a következő alrendszer nélkül semmit sem érne, hiszen a vak számára még mindig nem fogható fel, mert nem hangzó formában van. Ebből a szövegből egy hullámformátumú hangot kell generálni, olyat, amelyet a gép hang-

kártyája képes „megjeleníteni”, vagyis kimondani. Ezt teszi a következő modul.

– *TTS (Text To Speech) modul, azaz beszédszintetizátor*

Ez a modul önálló program. Egy olyan kisalkalmazás, szolgáltatás, amely önálló működésre nem képes. Nem rendelkezik saját grafikus felhasználói felülettel. Csupán más programok képesek „megkérni”, hogy az általuk bemenetként küldött szöveges információt hullámformátumú hanggá átalakítva küldje ki a hangkártya hangkimenetére, hogy az ott „megjelenhessen”, vagyis a szöveges bemenetből készítsen egy hangzó kimenetet. Tehát ez a modul felel azért, hogy az előző modul által generált szöveges üzenet a vak számára is érzékelhető legyen.

– *Beavatkozó, vagy szkriptíró modul*

Ez a modul nem minden képernyőolvasó programban érhető el, ugyanis erre nem minden esetben van szükség. Ez a modul nem más, mint egy, a képernyőolvasó program működését befolyásoló makrókészítő modul. Vagyis egy olyan alprogram, amelynek segítségével olyan szkripteket, makrókat lehet készíteni, amelyekkel módosítani lehet a képernyőolvasó működését.

Minden asztali operációs rendszer lehetővé teszi a felhasználó beavatkozását mind egérrel, mind pedig billentyűzettel. Az operációs rendszerek „evolúciója” során a billentyűzet kezdetől fogva a rendszer része volt, az egér jóval később jelent meg. Csupán a grafikus felhasználói felület megjelenésével lett általános. Ugyanakkor mind a mai napig a billentyűzet a legfontosabb beviteli eszköz. Ezen található olyan billentyűk is, amelyek az egérmutatót képesek mozgatni abban az esetben, ha a rendszerhez nincs egér csatlakoztatva. A képernyőolvasók legfontosabb navigációs billentyűi ezek az ún. „kurzormozgató billentyűk”, vagyis a fel és a le, valamint a jobbra és a balra nyilak, továbbá a TAB billentyű, mely a fókuszmozgató billentyű. A TAB lenyomásával egy adatbeviteli űrlapon egyik mezőről a következő mezőre ugrasztja a fókuszt. (Az egér tulajdonképpen ennek a TAB billentyűnek a kiváltására született, mert TAB billentyűvel csakis vezérlőelemről vezérlőelemre tudunk ugrani úgy, ahogy azt a programozó által beállított TABINDEX tulajdonságok adta TAB-sorrend lehetővé teszi.) Vagyis ha van egy űrlap, amelyen van 6 beviteli mező és két gomb, akkor

ezen TAB-sorrendje – minden valószínűség szerint – így fog kinézni:

1. mező 1
2. mező 2
- ...
6. mező 6
7. gomb 1
8. gomb 2

Ha csak az első mezőt kell kitöltenünk, a többi nem, és ezt a második gombbal tudjuk jóváhagyni, akkor a TAB billentyűvel csakis úgy érjük el a második gombot, ha végigugrálunk a kettő közötti összes elemen. Az egérrel viszont, ahogy beírtuk a kívánt adatot az első mezőbe, aktiválhatjuk rögtön a második gombot, hiszen könnyedén rá tudjuk pozicionálni az egérmutatót és az egér gombjával tudjuk aktiválni a „gomb 2”-t. Hogy ezt a billentyűzetről való, időt rabló, hosszú kurzorpozicionálási feladatot le lehessen rövidíteni, az operációs rendszerek is tartalmazznak beépített lehetőségeket. Ilyen például a Windowsban az ún. „forróbillentyű”. Ez leginkább a menükben vehető észre. Ha megnézzük egy program menüsorát, akkor láthatjuk, hogy egy-egy karakter alá van húzva. Ezek azok a bizonyos forróbillentyűk. Vagyis ha ezeket megfelelő módon aktiváljuk, akkor ugyanazt a hatást érjük el, mintha az egérrel rákattintottunk volna. Például a „Fájl” menűnél az „F” betű van aláhúzva. Ha a fókus a menüsoron van, és lenyomjuk az „F” billentyűt, akkor lenyílik a „Fájl” menü. A forróbillentyűk ún. „segédbillentyűje” Windows-ban mindig az ALT. Tehát ha az ALT+F billentyűkombinációt lenyomjuk, akkor le tudjuk nyitni a fájl menüt, ahol a fel/le nyílbillentyűk segítségével navigálhatunk, s ha kiválasztottuk a megfelelő menüelemet, azt az ENTER megnyomásával aktiválhatjuk. Vagyis azt szoktuk mondani, hogy az ALT+ forróbillentyű megfelel annak, mintha az egeret ráhúztuk volna és a bal egérgombbal kattintottunk volna.

Viszont sok esetben hiányoznak ezek a forróbillentyűk. Ekkor a vakok számára nem marad más, ha el szeretnék érné az adott űrlap második gombját, mint hogy a TAB folytonos nyomogatásával ugrálva ráugorjanak. Ez nemcsak időt rabló, de nagyon bosszantó is, főleg folyamatos munkánál. Ilyenkor a beavatkozó, vagy szkript modul ad segítséget, hogy egy, a képernyőolvasóba beépített programozási nyelv segítségével mindenki saját szája íze szerint szabja testre a dolgokat. Vagyis ez ad lehetőséget arra, hogy virtuálisan forróbillentyűt rendelhessünk bármihez egy adott programban.

## Böngészés

Ha a képernyőolvasó böngészőben megjelenített tartalommal találkozunk, úgy azt át kell alakítania a saját szája íze szerint”, hiszen a weboldalak tartalma egy HTML-kódba beágyazott tartalom, mely kód nagyon sok rejtett információt is képes tárolni és ezeket is meg kell tudnia „jelenítenie” a felhasználói számára. Ebben az esetben a képernyőolvasó a böngésző DOM (dokumentum object modell) objektummodelljén keresztül képes kinyerni a szükséges információkat. Ebből a DOM-ból létrehozza a weboldal ún. virtuális dokumentumát. Ami nem jelent mást, mint azt, hogy az egyes HTML-elemeket értelmezi, és azokból elkészíti azokat az információs blokkokat, melyeket aztán – a kívánt pillanatban – a beszédszintetizátornak elküld „megjelenítésre”. Egy ilyen renderelt oldal a következőképpen néz ki.

A MATARKA kezdőlapjának renderelt változata. Ezt hallja az ember, ha egy képernyőolvasóval felolvastatja a MATARKA kezdőlapját a Firefox és Explorer böngészők esetében:

### Szalag szakaszjelző

Első szintű címsor	MATARKA
Harmadik szintű címsor	Magyar folyóiratok tartalomjegyzékeinek kereshető adatbázisa
Hivatkozás	English version Alt+Shift+V

### Navigáció szakaszjelző

Keresés	Alt+Shift+1 gomb
Index	Alt+Shift+2 gomb
Folyóiratok	Alt+Shift+3 gomb
Kosár	Alt+Shift+4 gomb
Súgó	Alt+Shift+H gomb
EHM	kereső gomb
RSS	Alt+Shift+R gomb

### Keresés szakaszjelző

Keresés	szerkesztőmező
Keresés	gomb

### Navigáció szakaszjelző

Részletes kereső megnyitása	Alt+Shift+R hivatkozás
Súgó megnyitása	Alt+Shift+U hivatkozás

### Kiegészítés szakaszjelző

Összes találat utólagos szűkítése	Alt+Shift+S hivatkozás
-----------------------------------	---------------------------

2 oszlopból és 7 sorból álló táblázat  
Mettől (év) kombinált listamező 1800

Meddig év	kombinált listamező 2015
Szakterület	kombinált listamező
Válasszon szakterületet	kiterjesztett listamező
Folyóirat	kiterjesztett listamező
Jelölőnégyzet nincs bejelölve	Teljes szöveggel elérhető cikkek
Jelölőnégyzet nincs bejelölve	Az EPA-ban archivált cikkek
Szűkítés	gomb
Táblázat vége	

Első szintű címsor	Matarka
Harmadik szintű címsor	Röviden a szolgáltatásról .....

#### *Navigáció szakaszjelző*

Profilom	Alt+Shift+7 gomb
Kapcsolat	Alt+Shift+8 gomb
Magunkról	Alt+Shift+9 gomb
Belépés	gomb

#### *Kiegészítés szakaszjelző*

Támogatóink	
Hivatkozás	Internetszolgáltatók Tanácsa
Hivatkozás	Nemzeti Kulturális Alap

A képernyőolvasó program ezt a virtuális dokumentumot kezeli és a programot használó személy ezen keresztül képes a webdokumentummal „kapcsolatba” lépni, vagyis azon interakciókat végezni. A látók a böngésző tartalmához kizárólag csak az egér használatával férnek hozzá. A tartalomban a gördítősávval, vagy a lapozó billentyűkkel (PgUp, PgDown) tudnak navigálni. A sorról sorra való olvasást a szemükkel végzik. A vak erre képtelen. Ezért rendereli az ernaőolvasó program a fenti „egy sor-egy elem” módon a weboldalak tartalmát és hozza létre a virtuális dokumentumot. Ugyanis mikor a vak személy a fel-le billentyűkkel lépeget az oldal tartalmában, akkor valójában ebben a virtuális dokumentumban lépeget. A képernyőolvasó program gondoskodik arról, hogy a virtuális dokumentumban való navigálás és a böngészőben megjelenő tartalom szinkronban legyen, vagyis a fizikai fókus, már amennyiben ez értelmezhető, a virtuális dokumentumban kijelölt helyen jelenjen meg a böngésző tartalmában is.

A képernyőolvasó programok – programtól függően – kínálnak ún. webes navigációs segítséget. Ezek általában ún. navigációs billentyűk. Lévén a webdokumentumok nem szerkeszthető szöveges fájlok, ezért sok elemük nem fókuszképes, tehát TAB-bal nem lehet rá navigálni. A navigációs billen-

tyűk ezen a problémán segítenek. Például egy weboldalt – általában – illik címsorokkal tagolni. A „H” billentyű nyomogatásával a program felhasználója a weboldal címsorain (<h1>..<h6>) képes végigugrálni. Vagyis mintegy „feltérképezni” az oldalt. De ugyanígy vannak billentyűk az egyéb HTML-elemekhez is, mint pl. a „P” a <p> elemhez (bekezdés), vagy a „L” a <ul><ol> (lista) elemeken való gyors navigáláshoz.

#### **A MATARKA akadálymentesítése**

Az egyik dolog az, hogy mit mond a szabvány és az útmutató, és a másik dolog az, hogy hogyan használják a vakok a gyakorlatban a dolgokat. Az útmutató ernaőolvasóra vonatkozó javaslataiban van néhol ellentmondás. De az élet kitermelte már a jó, vakok által elfogadott, bevett gyakorlatot, mely az ernaőolvasók képességeinek kihasználásán alapul, illetve vannak olyan megoldások is, amelyek az útmutatóban nem igazán szerepelnek.

Amikor belefoglunk a projektbe, végiggondoltuk azt, hogy a MATARKA célközönsége mire használja a szolgáltatásunkat. Továbbá felmértük, hogy a célközönség feltehetően milyen informatikai, gépkezelési ismeretekkel rendelkezik, mert ezek a legfontosabb információk ahhoz, hogy a szoftver felhasználói felülete mindenki számára felhasználóbarát legyen. A felhasználói visszajelzések a legfontosabbak és nem a saját „dizájn elképzelések”. Egy elsődlegesen munkára, gyors információszerezésre használt szoftvernek elsősorban átláthatónak, könnyen érthetőnek, informatívnak kell lennie. A „dizájn” nem mehet a kezelhetőség rovására. Átgondoltuk azt is, hogy a megcélzott felhasználói réteg milyen hardvereszközökön keresztül fogja használni az adott szoftvert. Itt is, mint az előbbiekben a „minimumot” kell figyelembe venni és ebből kell kiindulni a felület tervezésekor.

A MATARKA új felhasználói felületének megtervezésekor az alábbiakból indultunk ki:

- A megcélzott felhasználók átlagos, vagy átlag alatti informatikai, gépkezelési ismeretekkel rendelkezők, akiknek a könnyű információhoz jutás a lényeg. Könnyű kezelés, egyszerűség és gyorsaság számukra a legfontosabb.
- A hardver, amelyen futtatni fogják az alkalmazást: leginkább asztali gépek, laptopok; kisebb részben táblagépek. Sok esetben régi Windows és böngésző verziók.
- A felületet akadálymentessé szeretnénk tenni a gyengénlátók, vakok számára.

Ezért a következő felülettervezési döntésekre jutotunk:

- Az átlagos, vagy átlag alatti informatikai ismeretekkel rendelkezők miatt a felületnek leginkább egy megszokott, asztali alkalmazás felületét kell imitálnia, vagyis menüsor és oldalsáv használatával kell megjeleníteni az információcsoportokat és azok alcsoportjait. (Lévén a legtöbb asztali programban ezt szokták meg a felhasználók, és ennek kezelése egyértelmű számukra.) A menüsoron a fő kategóriáknak kell elérhetőnek lenniük. A képernyő bal oldalán pedig oldalsávba kell rendezni az adott főkategória (menü) elemeit úgy, hogy a kiválasztott menü a menüsoron, és a menüelem az oldalsávon mindig egyértelműen jelezze a felhasználónak, hogy éppen hol van.
- A kétoszlopos vertikális elrendezés egyértelműbb a felhasználónak, mint a horizontális térszervezés. A keresésnél az első hasádba vertikálisan elrendezve lesznek a tényleges keresési feltételek, a második hasádba (tulajdonképpen ez a jobboldali oldalsáv) kerülnek a találatok. A kéthasábos elrendezéstől csak a folyóiratok évfolyamainak kibontásánál és a tartalomjegyzékek megjelenítésénél tértünk el.
- A laptopok és táblagépek felbontása miatt, továbbá az akadálymentes nagyíthatóság érdekében a webhely szélességét korlátozni kellett. A midi táblagépek kb. 800 képpont szélességben tudnak információt megjeleníteni. Ezért döntöttünk a bevált 770 pixeles szélesség mellett. Ugyanis az asztali gépeken, vagy laptopokon a Ctrl+ "+" -szal egyszerűen fel lehet nagyítani az oldalt és akkor minden arányosan nagyítódik. Illetve készíthető egy „teljes képernyőszélesség” CSS fájl, amit JavaScriptből dinamikusan lehet alkalmazni. Vagyis akkor az éppen aktuális képernyőszélességre lehet „kihúzni” az oldalt. Ez a „liquid” dizájn, ami mindig a rendelkezésre álló területhez igazodik. A szeparált CSS alkalmazása miatt ezt bármikor, utólag is meg lehet tenni és a kb. 2 soros JavaScript kód is beilleszthető utólag a sablonba, s így azt minden oldal „örökli”.
- A tesztelők véleményét figyelembe véve alakítottuk ki az oldal HTML-kódját, ami egy köztes kód lett. Ugyanis a tesztelők felhívták a figyelmünket arra, hogy sok helyen még a potenciális felhasználói körben WindowsXP-t, vagy Windows Vistát használnak régebbi verziójú böngészővel, melyek nem támogatják az összes HTML5 újítást. Ez a köztes kód azt jelenti, hogy a kód gyakorlatilag HTML5-ként kerül deklarálásra, mert a `<!DOCTYPE html>` deklaráció

HTML5-re deklarálja az oldalt. Viszont – mivel a leginkább elterjedt böngészők (IE, Firefox) régebbi verziói még egyáltalán nem tudják kezelni a HTML5 új szemantikai elemeit (`<header>`, `<footer>` stb.), illetve a régebbi képernyőolvasó programok sem ismerik ezeket a szemantikai elemeket, így ezek helyett a HTML4-ben használt régebbi megfelelőjük mellett döntöttünk. Vagyis az oldal szemantikai leírásában (`<body>` elemen belül) az új elemeket nem használtuk, helyettük a `<div>` elemeket alkalmaztuk. Megjelenés szempontjából semmiféle különbség nincs. A képernyőolvasó programok is a ROLE tulajdonság alapján jelentenek be elemeket, nem pedig a szemantika alapján. Szemantika szempontjából csupán a HTML1-től ismert alapelemeknek van jelentősége. Vagyis a címsorok, bekezdések stb. elemek a fontosak. A HTML5 által bevezetett elemek a jelenlegi segítő technológiák szempontjából lényegtelenek. Ezzel a megoldással a régebbi böngészők és a képernyőolvasók is gond nélkül tudják kezelni az oldalt. Azt azonban tudni kell, hogy a HTML5 összefoglaló név alatt 3 új technológia jelent meg. A HTML5 egyrészt tartalmazza az új HTML-szemantikai elemeket, továbbá vele együtt jelent meg az új JavaScript verzió, melyben új lehetőségekkel bővült a DOM programozhatóság, illetve ekkor jelent meg az új CSS3 szabvány is. A JavaScript új verziója és a CSS3 támogatás már mindegyik böngészőben megvan, ezért csupán a HTML-szemantikában kell kompromisszumokat kötni. A `<!DOCTYPE html>` deklaráció ezeket a technológiákat is engedélyezi, tehát mind a JavaScriptben, mind pedig a CSS-ben lehet ezáltal használni az újításokat. A CSS3-nál kellett ügyelni arra, hogy ennek adaptációja böngészőnként változik, tehát meg kellett nézni, hogy egy-egy adott CSS3 lehetőséget hogyan lehet egy adott böngészőre „testre szabni”.

- A teljes akadálymentesség miatt sem lehetett tiszta HTML5 szemantikát alkalmazni. Viszont a „köztes” kódban minden olyan dolog megvan, amely az új dizájnt a segítő technológiák számára „átláthatóvá” teszi és egy vak is könnyedén fogja tudni kezelni, mert a gyorsbillentyűk, a korrekt tartalmi elembejelentések (ROLE tulajdonság alapján), illetve az egyéb tulajdonságdeklarációk mind a legújabb, mind pedig a régi ernyőolvasó programok számára transzparenssé, kezelhetővé teszik a MATARKA-t.
- A többféle megjeleníthetőség miatt arra törekedtünk, hogy szeparált CSS fájlokban legyenek az oldalak megjelenítését megadó kódok. S

a CSS technológián belül is a dinamikus CSS csatolást láttuk ésszerűnek. Ez azt jelenti, hogy az oldalhoz tartozik egy, vagy több alap CSS fájl, mely minden esetben be kell töltsdjön (ez tartalmazza az alapvető formázást, mely mindig azonos kell legyen) és tartoznak úgynevezett „alternatív” CSS fájlok, melyek az oldal elemeinek vizuális megjelenítéséért felelnek. Ez a technológia teszi lehetővé azt, hogy eltérő megjelenítést lehessen alkalmazni az egyes felhasználói csoportoknál, ugyanis az alternatív CSS fájlokat JScript kódból könnyen lehet le- és felcsatolni, így könnyedén változtatható az oldal vizuális megjelenése. Magyarozatként álljon itt a használt sablon CSS deklarációja. Az első deklaráció a mindig betöltendő alap CSS fájlt csatolja. Itt a „REL” tulajdonsága „stylesheet” és nincs megadva „TITLE” tulajdonság. Ez jelzi, hogy ez a lap mindig kötelezően betöltendő. A második deklaráció abban különbözik az elsőttől, hogy itt már a „TITLE” tulajdonság meg lett adva. Ez az ún. alapértelmezett stíluslap.

A többféle megjeleníthetőség miatt arra törekedtünk, hogy szeparált CSS fájlokban legyenek az oldalak megjelenítését megadó kódok. S a CSS technológián belül is a dinamikus CSS csatolást láttuk ésszerűnek. Ez azt jelenti, hogy az oldalhoz tartozik egy, vagy több alap CSS fájl, melyek minden esetben betöltődnek (ezek tartalmazzák az alapvető formázásokat, melyeknek mindig azonosnak kell lenniük). Továbbá tartoznak hozzá úgynevezett „alternatív” CSS fájlok, melyek az oldal elemeinek vizuális megjelenítéséért felelnek. Ez a technológia teszi lehetővé azt, hogy eltérő megjelenítést lehessen alkalmazni az egyes felhasználói csoportoknál, ugyanis az alternatív CSS fájlokat JScript kódból könnyen lehet le- és felcsatolni, így könnyedén változtatható az oldal vizuális megjelenése.

Magyarozatként álljon itt a használt sablon CSS deklarációja:

- Az első deklaráció a mindig betöltendő alap CSS fájlt csatolja. Itt a „REL” tulajdonsága „stylesheet” és nincs megadva „TITLE” tulajdonság. Ez jelzi, hogy ez a lap mindig kötelezően betöltendő.
- A második deklaráció abban különbözik az elsőttől, hogy itt már a „TITLE” tulajdonság meg lett adva, de a „rel” tulajdonság továbbra is „stylesheet”, ami azt jelzi, hogy az alapér-

telmezett stíluslapokhoz tartozik. Tehát mindig betöltődik. Ez az ún. alapértelmezett szín-séma.

- A harmadik deklarációban a „REL” tulajdonság már „alternate stylesheet” és van „TITLE” tulajdonsága is. Ezek jelzik a böngészőnek, hogy ez a stíluslap alternatív megjelenítést tartalmaz, amit programból lehet betölteni JScript segítségével. (Ezek az alternatív stíluslapok, amennyiben átvált a felhasználó valamelyikükre, úgy felülírják az „Alapértelmezett” szín-séma stíluslap beállításait és ugyanannak az oldalnak teljesen más külsőt kölcsönöznek.

Ezáltal annyi alternatív megjelenítést lehet létrehozni, amennyit csak szeretne az ember. Az új JScript lehetőségekkel már komplett inicializációs fájlokat lehet írni egy-egy adott weboldalhoz, tehát meg lehet csinálni, hogy ha a felhasználó átállította a megjelenítést, akkor a következő belépéskor a beállított megjelenítése töltsdjön be és ne az alapértelmezett. Így az ebben megadott színek stb. lesznek az „alapértelmezettek”. A harmadik deklarációban a „REL” tulajdonság már „alternate stylesheet” és van „TITLE” tulajdonsága is. Ezek jelzik a böngészőnek, hogy ez a stíluslap alternatív megjelenítést tartalmaz, amit programból lehet betölteni JScript segítségével. Ezáltal annyi alternatív megjelenítést lehet létrehozni, amennyit csak szeretne az ember. Az új JScript lehetőségekkel már komplett inicializációs fájlokat lehet írni egy-egy adott weboldalhoz, tehát meg lehet csinálni, hogy ha a felhasználó átállította a megjelenítést, akkor a következő belépéskor a beállított megjelenítése töltsdjön be és ne az alapértelmezett.

```
<link rel="stylesheet" type="text/css"
href="styles/layout.css" media="screen"/>
<link rel="stylesheet" title="CSS.black"
type="text/css" href="styles/black.css"
media="screen"/>
<link rel="alternate stylesheet" title
="CSS.color1" type="text/css"
href="styles/style3.css" media="screen"/>
```

- Oldal alján menü. A legtöbbször utálják a „vissza az oldal tetejére” hivatkozást. Számukra sokkal jobb lenne, ha ott is lenne lehetőség eldönteni, hogy hová akar továbbmenni. Ezért alkalmaztuk a láblécmenüt, illetve a gyorsbillentyűket. Egy idő után meg-



tanulják a felhasználók, hogy az Alt+Shift+... kombinációval egyszerűen átugorhatnak egy-egy másik oldalra, illetve az Alt+Shift+H (mint header) visszaugrasztja őket az adott oldal tetejére.

- A régi MATARKA-ban az űrlapok kódolása vizuálisan rendben volt, de „vaksági” szempontból teljesen rossz volt. Ugyanis a képernyőolvasó félreérthetően, nem egyértelműen jelentette be az egyes elemeket. A vezérlőelemekhez tartozó címkéket nem tudta párosítani, így vakon, az ernyőolvasóra hivatkozva sokszor nem lehetett eldönteni, hogy mit is jelent az adott vezérlőelem.
- A régi MATARKA-t csak mint kiindulási alapot használtuk arra, hogy felmérjük, milyen sablonokra van szükség, vagyis, hogy milyen oldalszerkezeteket használunk. Ez nem változott, hiszen a MATARKA funkciója sem változott.
- Színek. Mivel az oldal kódja és az oldal megjelenését szabályozó CSS fájlok szeparáltak, így bármikor lehetőség van arra, hogy „újrarajzoljuk” a MATARKA felületét anélkül, hogy ez a felület kódját érintené. A fent felvázolt dinamikus CSS csatolás pedig azt is lehetővé teszi, hogy mindenki más séma szerint használja, hiszen ha van több „skin” akkor a felhasználó kiválaszthatja, hogy me-

lyik szimpatikus neki, és azután már automatikusan azt fogja neki betölteni a böngésző, ha a MATARKA oldalára téved.

- A sablonban látszik, hogy van egy szkriptcsatolás. Ez a JQuery, amely lehetővé teszi, hogy „feldobjuk” az oldalt. Egyszerűen lehet vele vizuális effekteket csinálni. Mint például azt, hogy a találati lista (2. ábra) nemcsak szimplán megjelenik, hanem mintegy „előtűnik” a semmiből – mindössze egy kódsor az egész. A viszonylag egyszerű, átlátható szerkezet a CSS3 és a JQuery segítségével egy nagyon ütős és modern élményt adhat a felhasználóknak. Nem szabad elfelejteni, hogy a MATARKA elsődlegesen a kutatást szolgálja és nem játék, tehát a dizájn nem mehet a használhatóság rovására. Viszont „tarkítani” lehet. Vagyis például az egyes menük és azok elemei kaphatnak kis ikonokat, mint ahogy a Windowsban is van. Illetve a találati listákat is lehet kis ikonokkal „gazdagítani”, de ez már részletkérdés, hogyha a teljes webhely szerkezete letisztul, akkor ezek beillesztése viszonylag könnyű. Hiszen például a menü elemeinél csupán az adott elem „list-style” tulajdonságcsoportját kell megfelelően beállítani a CSS-ben, hogy egy ikon jelenjen meg előtte.

2. ábra Találatok – összesítő oldal

## Milyen volt és milyen lett

A webes akadálymentesítés nem más, mint a weboldalak olyan módon való megalkotása, hogy abból a képernyőolvasó program a vak személy által is „fogyasztható” tartalmat hozzon létre, és ez a tartalom egyezzen meg a látó személy által vizuálisan érzékelt tartalommal. Vagyis, ha egy látó és egy vak beszélget ugyanarról a weboldalról, akkor ne lehessen félreértés.

A MATARKA első verziója – ismeretek hiányában – nem tartalmazott szinte semmiféle akadálymentesítést. Így az oldalak tartalma esetlegesen volt csak címsorokkal tagolva. Legtöbbször ún. „virtuális címsorok” voltak, amik nem mások, mint sima <p> elemek félkövéren formázva. Sajnos a képernyőolvasók ezekkel semmit nem tudnak kezdeni, hiszen ők nem a formázás, hanem a DOM alapján

értelmezik a dokumentumot. Így a navigáció az oldalakon nagyon nehézkes volt, hiszen csupán a fel és a le nyilakkal lehetett sorról sorra lépkedve elérni az egyes dolgokat. Továbbá hiányoztak a virtuális dokumentum szakaszolását lehetővé tevő, az ún. virtuális felhasználói felület kialakításához szükséges ARIA tulajdonságok, melyek azért fontosak, mert a vak személy számára ezek teszik lehetővé azt, hogy mintegy „elképzelje” az oldalt. Ezek leginkább szakasz-, illetve vezérlőelem-jelzők, hiszen egy vak személy csupán így kaphat támpontot arról, hogy az oldal éppen melyik szakaszában tartózkodik, vagy éppen milyen „vezérlőelemen” van. (A képernyőolvasó programok egy része képes arra, hogy ezeket a szakaszokat, vezérlőelemeket egy listában megjelenítse és onnan kiválasztva a vak személy azonnal az adott elemre tudjon navigálni.)

Keresés Index Folyóiratok Kosár Súgó EHM kereső RSS

**Tudományos és műszaki tájékoztatás**

Teljes szöveg: [tmt.omikk.bme.hu/archiv.html](http://tmt.omikk.bme.hu/archiv.html) - válogatott cikkek 1954/1 számtól, utolsó 6 hónap jelszóval  
Honlap: [tmt.omikk.bme.hu](http://tmt.omikk.bme.hu)  
RSS

**62. évf. 1. sz. / 2015**

Kosárba MARC formátum

**AJÁNLÓ**

- Fonyó Istvánné** 1  
A digitális írástudástól a tudástárig...

**CIKKEK**

- Székelyné Török Tünde - Virág Gabriella** 3-12  
Az Eötvös Loránd Tudományegyetem, Digitális Intézményi Tudástára. Az EDIT bemutatása - Két konkrét állományrész feldolgozásának ismertetésével
- Új böngészőt fejleszt a Microsoft** 12
- Egervári Dóra** 13-21  
Felnőttek a digitális dzsungelben. A felnőttoktatás és az információs műveltség kapcsolata

**MŰHELYMUNKÁK**

- Áts József** 22-27  
Az MTMT és az intézményi hozzárendelés

3. ábra A TMT egy friss számának tartalomjegyzéke

A jelenlegi verzióban a gyakorlati akadálymentesítésre törekedtünk, aminek során sok problémával találkoztunk, olyanokkal, amelyekre nem is számítottunk. Ezek az alkalmazott modern technológiák által „generált” problémák, melyek közül az egyik legfontosabb az, hogy a lekérdezések futtatása után a jobb oldali adatterületen megjelenő információkról a vak semmi visszajelzést nem kap. A látó számára érthető vizuális jelzések, mint mondjuk, hogy a keresés éppen folyamatban van, vagy hogy hiba lépett fel, a vak számára „rejtve” maradnak. Hiszen nem történik olyan esemény a böngészőben (oldalújratöltés), ami a képernyőolvasó programot arra készítetné, hogy újra renderelje az oldalt. Erre szolgál megoldásul majd a következő verzióban beépítésre kerülő, „második” ARIA tulajdonságcsoport, melyek használatával elérhető lesz, hogy az oldalon belüli, újratöltés nélküli tartalomterület-frissítéseket azonnal bejelentse a program és a vakok is ugyanazt a felhasználói élményt legyenek képesek megélni, amit a látók. Jelenleg a szabvány által „preferált” megoldás, az ugrató billentyű került alkalmazásra, aminek a használata jelen esetben kényelmetlen és „diszkriminatív”, hiszen a látó azonnali vizuális információkat kap, míg a vak csak akkor jut ezekhez, ha az ugrató billentyűvel a frissülő területre ugrik és ott a fel-le nyilakkal felolvastatja annak tartalmát (3. ábra).

## Irodalom

1. KÁLMÁN Zsófia – KÖNCZEI György: A Taigetosztól az esélyegyenlőségig. Budapest, Osiris Kiadó, 2002. p. 21.
2. BARTAL Tamás honlapja.  
<https://bartal.org/web-accessibility-akadalymentes-weboldalak.html>

3. Web Akadálymentesítési Útmutató (WCAG) szabvány.  
<http://www.w3.org/TR/WCAG20/>  
Magyar fordítás:  
<http://www.w3c.hu/forditasok/WCAG20/>
4. W3C Magyar Iroda honlapja.  
<http://www.w3c.hu/szolgaltatasok/akadalymenteshonlap.html>
5. WebAIM web accessibility in mind.  
<http://webaim.org/projects/practitionersurvey>
6. ÁRPÁDHÁZY-GODÓ Csaba: Akadálymentes-e a web? Magyar múzeumok online. 2014-09-16.  
[http://www.magyarmuzeumok.hu/tema/2021\\_akadalymentes-e\\_a\\_web](http://www.magyarmuzeumok.hu/tema/2021_akadalymentes-e_a_web)
7. „Informatika a Látássérültekért” Alapítvány honlapja.  
<http://www.infoalap.hu/felmeres/>

Beérkezett: 2015. VIII. 12-én.



### **Burmeister Erzsébet**

a Miskolci Egyetem  
Könyvtár, Levéltár, Múzeum  
főkönyvtárosa.  
E-mail: [erzsi@uni-miskolc.hu](mailto:erzsi@uni-miskolc.hu)



### **Árpádházy-Godó Csaba**

magánvállalkozó.  
E-mail: [arpadhazy@hotmail.com](mailto:arpadhazy@hotmail.com)

---

## **E számunk megjelenését önkéntes munkájával segítette:**

*Árpádházy-Godó Csaba  
Barton Dávid  
Berke Barnabásné  
Burmeister Erzsébet  
Császár János  
Fonyó Istvánné  
Hegyközi Ilona  
Kiszl Péter*