**Célkitűzés:**

A tárgy célja megismertetni a hallgatókkal a vizuális és beszéd interfész technológiákat az ember-gép kapcsolatban (HCI), külön hangsúlyt fektetve az intelligens környezetekben (város, otthon, munkahely) működő alkalmazások és szolgáltatások személyre szabott, hely- és környezetfüggő interfészeire. A tárgy keretén belül részletes bemutatásra kerülnek a felhasználói interfész elemei, a szoftver-ergonómia alapelvei, a szoftverek ergonómiai szempontból történő kiértékeléseinek módszerei. Az elméleti témakörök ismertetésével párhuzamosan gyakorlatok keretében is feldolgozzuk az anyagot. Ezenkívül a hallgatók gyakorlati feladatok megoldásával igazolják a témakörben szerzett jártasságukat. A kurzus végére a hallgatók megtanulják a felhasználói interfész tervezéséhez, teszteléséhez, minősítéséhez szükséges alapelveket, hogy azt majd gyakorlatban is alkalmazhassák a későbbi munkájuk folyamán.

A TANTÁRGY TEMATIKÁJA:

1. Bevezetés

* Alapfogalmak, definíciók

2. Ember és környezete közti modalitás típusok

* Beszéd interfész
* Vizuális interfész
* Taktilis interfész (tapintás)
* Multimédia HCI
* Interfész modalitások együttes kezelése és szinkronizálása
* Beszédinterfész
* Beszédkommunikáció

3. Vizuális interfész

* Iteratív tervezés alapelvei
* Iteratív tervezés módszerei

4. Felhasználói interfész technikák

* Irányelvek
* Aranyszabályok a tervezésben
* 5. Felhasználói interfész alapelvek és példák
* Menürendszer
* Szöveg dialógus
* Grafikus interfész
* Interfész a weben
* Dialógus rendszerek
* Intelligens környezet

6. Felhasználói interfész mobil eszközökön

* Általános alapelvek
* Operációs rendszer-függő kérdések
* Modalitás-függő kérdések

7. Tervezési irányelvek (design guidelines)

* Fókuszcsoport módszer
* Conjoint analízis
* Design space analízis
* GOMS modell

8. Honlapok használhatósága

* Különleges felhasználói felületek (pl. multimédia, groupware)
* Mindenki számára használhatóság (W3C WAI)

9. Felhasználói interfész kiértékelés

* Kiértékelés kritériumai
* Kiértékelési eljárások

10. Intelligens környezetek

* Hely- és környezetfüggő interfészek
* Alkalmazkodás egyéni igényekhez

11. Felhasználói interfész esettanulmányok

1. előadás:

* Bemutatkozás, adminisztrációs kérdések, időbeosztást stb.
* HCI angol terminológia
* HCI történelmi áttekintése
* HCI a gyakorlatban
* HCI angol terminológia
* Gartner Hype Cycle ([**2009**](http://www.gartner.com/it/page.jsp?id=1124212), [**2015**](http://www.gartner.com/newsroom/id/3114217)),
* aktuális HCI-t érintő technológiák

Néhány a weben is elérhető olyan oldal, amely kapcsolódik az előadáshoz. Az anyagoknak sokszor csak egy része szerepel az előadásokon, illetve az előadások más interneten nem elérhető források alapján is készülnek, ezért az előadáson elhangozottak különbözhetnek az itt szereplő művek tartalmától. Az itt található linkek függetlenek az oktatótól, ezért tartalmért nem tud felelőséget válalni, illetve tartalma változhat. Ezek mellett az angol nyelvű Wikipedia általában segítséget ad, ez külön nem lesz feltűntetve:

* [**Fülöp Géza: Az információ**](http://mek.oszk.hu/03100/03118/03118.pdf)
* [**Gartner 2009**](http://www.gartner.com/it/page.jsp?id=1124212)

2. előadás:

Felhasználó központú tervezés User Experience Design (UXD) entails conducting user research exercises with intended users of a system. User research reveals users’ needs and preferences through user observations, one-on-one interviews, and creative activities that encourage users to express their emotions, motivations, and underlying concepts and beliefs about the steps involved in task procedures. By understanding the human emotions, motivations, and beliefs that surround a task, a user interface can be designed to accommodate and support user behaviors in a way that users will experience as natural and satisfying.

 Felhasználók

 Feladatok

 Környezet

Felhasználó központú tervezés

 Szabvány:

 A felhasználó-központú tervezés indokoltságának kifejtése

 A felhasználó-központú tervezés előkészítő tervezése

 A felhasználó-központú tervezés alapelvei

 A felhasználó-központú tervezés tevékenységei

Alapelvek

 A felhasználók aktív bevonása és a felhasználói, valamint a feladatból adódó követelmények megértése

 A felhasználók és a technológia szerepének megfelelő kijelölése és szétválasztása

 A tervezési megoldások iterációja

 Multidiszciplináris tervezés Több tudományt, szakterületet érintő.

Tevékenységek

 A használat kontextusának meghatározása és megértése

 A felhasználói és szervezeti követelmények meghatározása a használhatósággal kapcsolatos célkritériumok meghatározása

 Megoldások készítése

 A megoldások értékelése a követelmények alapján

[**Fejlesztési modellek**](http://subhendu.net/ucd-based-development-and-evaluation-of-social-media-website/3/)

* Requirements gathering – Search for the user requirement of the system
* Requirements specification – Compile, Specify and finalise the requirements
* Design – Developing the initial designs and prototypes
* Evaluation – Doing user-based assessment of the design and interaction

Vízesés modell (ellenpélda)





Iteratív

Felhasználók bevonása

 Megfigyelés

 Modellek

 Bevont felhasználók

 Tesztelés és kiértékelés

Kik a felhasználók

 Megrendelő

 Szervezeten belüli felhasználók

 Végfelhasználók

Csillag





Fejlesztő csapat

 Vezetők

 Marketingesek

 Sw mérnökök

 Programozók

 Grafikusok

 HCI specialisták

 Egyéb kutatók

 Fejlesztői munkaerő

 ...

 Felhasználók

Információk begyűjtése

 Megfigyelni a felhasználót

 Eredeti környezet

 Laboratóriumi környezet

 Megfigyelés hatása

Információk begyűjtése

 Interjú

 Strukturált

 Flexibilis

 Kérdőívek

 Zárt kérdések

 Skálák

 Nyitott kérdések

7+/-2

 Memória kapacitás

 Menü szervezés

 Választási opciók

2 mp szabály Gyors

 Reagálni kell a rendszernek

 Homokóra

3 kattintás

 Ha nem kapja meg elpártol

 Lehet több bizonyos feltételek mellett

 Hol

 Honnan

 Hová

Pareto elv **Az eredményeid 80%-a a tevékenységeid 20%-ból ered.** Ez egy általános megfigyelés, azonban természetesen ez az arány lehet még magasabb is, pl. 99/1. Nem is az a fontos, hogy mi a konkrét arány, csupán az, hogy megértsd, vannak olyan tevékenységek az életedben amelyek az eredményeid, boldogságod, elégedettséged, munkabeli sikereid – és a sort még lehetne folytatni a végtelenségig – jelentős részét okozzák. És vannak azok a tevékenységek, amelyekre rendkívül sok idő rámegy, mégis csekély mértékben járulnak hozzá a sikereidhez.

 20 % -> 80 %

 Meghatározni

 Jól kidolgozni

Ajánlott könyv: [**Debbie Stone, Caroline Jarrett, Mark Woodroffe, Shailey Minocha: User Interface Design and Evaluation**](http://www.amazon.com/Interface-Design-Evaluation-Interactive-Technologies/dp/0120884364)
[**Google könyv formátum**](http://books.google.com/books?id=VvSoyqPBPbMC&dq=Debbie+Stone,+Caroline+Jarrett,+Mark+Woodroffe,+Shailey+Minocha:+User+Interface+Design+and+Evaluation&printsec=frontcover&source=bn&hl=hu&ei=LU-XTIzOE82UswabyPxa&sa=X&oi=book_result&ct=result&resnum=4&ved=0CC0Q6AEwAw#v=onepage&q=Debbie%20Stone%2C%20Caroline%20Jarrett%2C%20Mark%20Woodroffe%2C%20Shailey%20Minocha%3A%20User%20Interface%20Design%20and%20Evaluation&f=false)

* [**30 Usability Issues To Be Aware Of**](http://www.smashingmagazine.com/2007/10/09/30-usability-issues-to-be-aware-of/)
* [**vízesés és csillag modell**](http://lipas.uwasa.fi/~mj/hci/hci5.html)
* [**Introduction to User-Centered Design**](http://www.usabilityfirst.com/about-usability/introduction-to-user-centered-design/)

3. előadás:

Aranyszabályok

 1. Konzisztenciára

 2. Shortcuts (rövidebb út)

 3. Hasznos visszajelzések

 4. Értelmes végű párbeszédek

Aranyszabályok II.

 5. Egyszerű hibakezelés

 6. Undo

 7. Írányítás érzése a felhasználónál

 8. Csökkentsük a rövid idejű memória terhelését

[**Fitts törvénye**](https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/fitts-s-law)

Az 1954-ben megalkotott Fitts törvény kimondja, hogy egy cél eléréséhez szükséges idő a cél méretének és közelségének függvényével írható le.

A webdizájnerek ezt a törvényt használják arra, hogy intuitív felületeket hozzanak létre. Ezért van minden web 2.0-ás szájton böhöm nagy logo regisztrálok felirattal, ezért nagyok a menü elemek és ezért emelik ki a kattintható címeket.

A nagyítás azonban sok helyet vesz el és bizonyos elemek kiemelésével rengeteg más elemet el kell nyomni. Az oldalak tervezésénél természetesen a legtöbb esetben tesznek rá. Tudják, hogy a marketingesek a regisztrációs átkattintásokat mérik, a sikeresség a “nyomd meg a piros gombot” effektus kiváltásában van. Lényegtelen, hogy az újonnan létrejövő kereskedelmi célú címlapokon lényegében minden elem a nagy gombok töltelékét adja, kerülőutak, amik minden esetben az átkattintásra ösztönöznek, ez a dolgok rendje.



 Vegyük figyelembe tervezéskor

Fordított piramis

A fordított piramis elv módszere szerint a lényeges információk a szöveg elejére kerülnek, utána következik a kifejtés. Azaz a szöveget úgy célszerű kifejteni, hogy a legfontosabb információ kerüljön a nyitóoldalra, míg a szűkebb közönségnek szóló információk a megfelelő aloldalakon legyenek elérhetőek. Így a túlságosan részletes információk nem terhelik fölöslegesen az általános információkra kíváncsi érdeklődőt.

 Írói stílus eredetileg

 Konkluzió elöl

 Többi hátul

quick'n'dirty-solutions Egyes funkciók elég, ha valahogy működnek először.

 Éppen megfelelő

 Nem fogja a legjobbat megkeresni

graceful degradation

The premise for graceful degradation is to first build for the latest and greatest, then add handlers for less capable devices. In other words, focus on the majority before catering to those outside the mainstream. This is quite similar to the visual approach of web design, where the first priority is to make it look good to most visitors.

 Ha valami nem elérhető

 Csökkentett mód használható legyen

So, **graceful degradation** is the practice of building your web functionality so that it provides a certain level of user experience in more modern browsers, but it will also *degrade gracefully* to a lower level of user in experience in older browsers. This lower level is not as nice to use for your site visitors, but it does still provide them with the basic functionality that they came to your site to use; things do not break for them.

**Progressive enhancement** is similar, but it does things the other way round. You start by establishing a basic level of user experience that all browsers will be able to provide when rendering your web site, but you also build in more advanced functionality that will automatically be available to browsers that can use it.

Walk-Up-And-Use Design

A Walk-up-and-use system is a system that needs to be so self-explanatory that first-time or one-time users can use the system effectively without any prior introduction/training.

Examples are automatic teller machines (ATMs), ticket machines, public informations systems, museum displays, etc.

With such variables, it's difficult to design a “walk up and use” application that will satisfy everyone. Perhaps the “skins” concept may be appropriate here.

 Egyszeri felhasználók kiszolgálása

 Ha elsőre nem sikerül, nem tér vissza?

Cliffhanger-Effect (Zeigarnik-Effekt)

 Bizonytalanságot nem szereti az ember.

 Hirdetők kihasználják

Széles körnek szóljon

 Sok fajta képességű felhasználó

 Sok féle igény

Kezelhető információ mennyiség

 Kognitív terhelés

 Formok kitöltése

Színek

 Színek nélkül is értelmes maradjon a tartalom

 Színtévesztők ([**www.vischeck.com**](http://speechlab.tmit.bme.hu/hci/www.vischeck.com))

Hang és audió

 Egyenértékű alternatíva

 Hibajelzés zajos környezet

Tiszta nyelvezet

 Felhasználói körnek megfelelő

 Udvarias

 Több szintű kommunikáció

Struktúra

 Struktúra helyes használata

 Azonos feladatokra azonos struktúra

Táblázatok

Időben változó elemek

 Engedjük, hogy a felhasználó megállítsa

 Növeli a kognitív terhelést

 Elvonja a figyelmet

 Informatív is lehet

Hibakezelés

 Visszavonható

 Értelmes hibaüzenetek

Szoftverek minősége

 Funkcionalitás

 Megbízhatóság

 Használhatóság

 Hatékonyság

 Karbantarthatóság

 Hordozhatóság

 [**ISO 9126**](https://en.wikipedia.org/wiki/ISO/IEC_9126)

Használhatóság

 Érthetőség

 Tanulhatóság

 Működtethetőség

Használhatóság

  [**Quesenberry (2003) The Five Es (http://www.wqusability.com)**](http://www.wqusability.com/articles/getting-started.html)

 Effective,

 Efficient,

 Engaging,

 Error tolerant

 Easy to learn

Használhatóság

 Eredményes

 teljes és pontos ==> elérje a célját

 Hatékony

 gyors, (pontos)

 Megnyerő, kellemes

 stílus, megelégedés

 Hibatűrő

 megelőzés

 kijavítás

 Könnyen tanulható

 kezdeti állapot

 elmélyülés

4. előadás:

 BPdialog vendégelőadás

 [**http://www.budapestdialog.hu**](http://www.budapestdialog.hu/)

 [**http://rogtonjovok.hu**](http://rogtonjovok.hu/)

5. előadás:

Kiértékelés

 Miért?

 Diagnosztikus, hibakereső kiértékelés Helyzetfeltáró értékelési mód, ami a meglévő hibák kiküszöbölésére fokuszál.

 Mérés típusú kiértékelés Mérésekből statisztika készíthető.

Kiértékelés korai fázisban

 Felhasználói követelmények ellenőrzése

 Előrejelzése a használhatóságnak

Kiértékelés a későbbi fázisokban

 Majdnem kész rendszeren

 Felhasználói igényekkel mennyire találkozik a UI

 Az itt felfedezett dolgok már kevésbé kerülnek be az aktuális tervezésbe

 Következő verzióba, vagy upgrade-be

[**Kiértékelés: 433.oldal**](https://books.google.hu/books?id=VvSoyqPBPbMC&pg=PA665&lpg=PA665&dq=%22Process%20of%20usability%20evaluation+433%22&source=bl&ots=d6M1OZkLTa&sig=JVpNqb0mk-wCbq8sSohQkbEEZ_Q&hl=en&sa=X&ved=0ahUKEwjPmbeA4cLJAhXBFg8KHZP1Ca8Q6AEIRjAG#v=onepage&q=process%20of%20usability%20evaluation&f=false)



Végeredmény

 Elvárásoknak megfelelő eredmény Sikerült-e megvalósítani, amit szerettünk volna? A használhatósági elvárásoknak eleget tettünk?

 Nem biztos eredmény Nem tökéletes megvalósítás.

 Elvárásokkal nem találkozó eredmény Egyáltalán nem azt sikerült összehozni, mint amit szerettünk volna, ez nem jó.

 Megválaszolandó kérdések

 Mi a célja a kiértékelésnek? How do you move from wanting to improve the usability of your web site, product or software to taking action?

 Milyen adatokat kell összegyűjteni?

 Mit kell tesztelni? Szoftver funkciói.

 Milyen megkötések vannak?

 Milyen felhasználókat válasszak? Leendő felhasználók és potenciális felhasználók között.

 Milyen folyamatokkal értékeltessek ki? Kipróbálás majd kérdőív.

 Hol értékeltessek ki? Lehetőleg a célcsoportban.

 [**Quesenberry (2003) The Five Es (http://www.wqusability.com)**](http://www.wqusability.com/articles/getting-started.html)

* **Effective:** How completely and accurately the work or experience is completed or goals reached
* **Efficient**: How quickly this work can be completed
* **Engaging**: How well the interface draws the user into the interaction and how pleasant and satisfying it is to use
* **Error Tolerant:** How well the product prevents errors and can help the user recover from mistakes that do occur

**Easy to Learn:** How well the product supports both the initial orientation and continued learning throughout the complete lifetime of use

* **Effective**: Watch for the results of each task, and see how often they are done accurately and completely. Look for problems like information that is skipped or mistakes that are made by several users.
* **Efficient:** Time users as they work to see how long each task takes to complete. Look for places where the screen layout or navigation make the work harder than it needs to be.
* **Engaging:** Watch for signs that the screens are confusing, or difficult to read. Look for places where the interface fails to draw the users into their tasks. Ask questions after the test to see how well they liked the product and listen for things that kept them from being satisfied with the experience
* **Error Tolerant:** Create a test in which mistakes are likely to happen, and see how well users can recover from problems and how helpful the product is. Count the number of times users see error messages and how they could be prevented.

**Easy to Learn**: Control how much instruction is given to the test participants, or ask experienced users to try especially difficult, complex or rarely-used tasks. Look for places where the on-screen text or work flow helps…or confuses them

Múzeum

Adatok

 Mennyiségi

 Minőségi

Peremfeltételek

 Pénz

 Idő

 Elérhető eszközök

 Elérhető tesztelők, beszervezés költsége

 Elérhető kiértékelők

Dokumentálás

 Nem hosszú

 De le kell írni

 Csapat többi tagja is informálva legyen

Ki értékeljen ki?

 Ki a valódi felhasználó?

 Egyedül vagy párban?

 Hány tesztelő kell? Reprezentatív?

 Tartalék tesztelők

 Ösztönző eszközök Csoki

 Toborzó átvilágító Bízunk benne, hogy tisztességes választ kapunk, nem trollkodják el.

Tesztelő típusok

 Kollégák hasonló rendszeren

 Kollégák ismeretlen rendszeren

 Családtagok, barátok

 Valódi felhasználók

 Segítők, nonprofit szervezetek

Időbeosztás

 Mennyi idő legyen?

 kb. 30-90 perc /menet

 Időbeosztás tartalékokkal

Feladatok előkészítése

 Feladatok

 Főbb funkciók

 Feladatok amik a felhasználóknak fontosak Megvalósítottuk?

 Új elemek

 Kritikus elemek

 Marketing szempontból fontos részek

 Tervezéskor is használt szcenáriók

 Kitalálás: kiértékelő v. tesztelő

Hol?

 Helyszíni

 valós környezet Ők kattintatanak?

 Kontrollált környezet

 pontosabb infó Vagy mi kattintottunk helyettük.

Adatok gyűjtése

 Loggolás

 Beszélgetés

 Jegyzetelés

 Interjú tesztelés után

 Kérdőívek

 Videó

 Audió

 Szemkövetés

 Mi van, ha nem egyezik bele?

Előkészületek a kiértékeléshez

 Személyek

 Betanító

 Dokumentáló

 Technikus

 Megfigyelők

 Házigazda

 Toborzó

 Az egyszemélyes kiértékelő

Előkészületek

 Menetrend írása

 Beleegyező nyilatkozatok

 Próba teszt

Adatok elemzése

 Adatok összegyűjtése

 Összesítés, statisztikai elemzések

 Használhatósági problémák azonosítása

 Értékelés felhasználók tulajdonságát figyelembe véve

 Ajánlott változtatások ==> Kiértékelési beszámoló

Speciális interfészek

Érzeti interfész

 Nyomás-tapintás (mechanikai)

 Hideg-meleg (hőmérsékleti)

 Fájdalom

 Mechanikai

 Nyomás

 Tapintás

 Vibráció

 Bizsergetés, csiklandozás (Tickle)

Érzeti intefész játékhoz

Érinthető hologram

Gesztus interfész

 Természetes

 Kéz és ujjak mozgását

 Arc mozgást

 Web funkciók

 Kesztyűk

 Wii

Arc követés

 Érzelem detektálás

Mozgási interfész

 Locomotion interface

 A mozgás a virtuális és a valós térben is

 Kinect

 Leap motion

Ajánlott könyv: [**Philip Kortum: HCI Beyond the GUI: Design for Haptic, Speech, Olfactory, and Other Nontraditional Interfaces**](http://www.amazon.com/HCI-Beyond-GUI-Nontraditional-Technologies/dp/0123740177)
[**Elsevier**](http://www.sciencedirect.com/science/book/9780123740175)

* [**Project Natal**](http://www.xbox.com/en-US/live/projectnatal/)
* [**Motion Activated Headphones**](http://www.sonyericsson.com/putyourearsincontrol/)
* [**3rd Space Vest**](http://tngames.com/)
* [**Virtual Laser Keyboard**](http://www.virtual-laser-keyboard.com/)
* [**Touchable Holography**](http://www.alab.t.u-tokyo.ac.jp/~siggraph/09/TouchableHolography/SIGGRAPH09-TH.html)
* [**First Commercial Motion Capture MIDI Controller**](http://www.sonalog.com/product.htm)
* [**Projection-Based Olfactory Display**](http://vrlab.meijo-u.ac.jp/~yanagida/scent/index.html)
* [**Data Gloves**](http://www.5dt.com/hardware.html#glove)

6. előadás:

####  IBM Watson vendégelőadás Erről tudunk valamit? **IBM Watson is a technology platform that uses natural language processing and machine learning to reveal insights from large amounts of unstructured data.**

* [**IBM Watson: How it Works**](https://www.youtube.com/watch?v=_Xcmh1LQB9I)
* [**BM Watson: Smartest Machine ever built - Documentary**](https://www.youtube.com/watch?v=uDBZnaoJVlk)

7. előadás:

 Féléves feladat, hallgatói beszámolók Nem kell.

8. előadás:

 HTTP protokoll, WWW <https://hu.wikipedia.org/wiki/HTTP>

A **világháló** (angol eredetiben *World Wide Web, WWW* vagy röviden *Web)* az [interneten](https://hu.wikipedia.org/wiki/Internet) működő, egymással úgynevezett [hiperlinkekkel](https://hu.wikipedia.org/wiki/Hiperlink) összekötött [dokumentumok](https://hu.wikipedia.org/w/index.php?title=Dokumentum&action=edit&redlink=1) rendszere. A rendszert [webböngésző](https://hu.wikipedia.org/wiki/Webb%C3%B6ng%C3%A9sz%C5%91) program segítségével lehet elérni. Ez a program képes megjeleníteni az egyes dokumentumokat, „[weblapokat](https://hu.wikipedia.org/wiki/Weblap)”. A felhasználó a lapokon található hiperlinkek segítségével további lapokat kérhet le, amelyeken újabb hiperlinkek lehetnek. A rendszer „háló”-jellegét is ez adja; a dokumentumok a háló csomópontjai, míg a hiperlinkek a háló szálai, amelyeken keresztül egy vagy több lépésben tetszőleges csomóponthoz eljuthatunk.

 Szemkövető rendszerek, példák, Fitt szabály szemléltetése, esettanulmány

* Web 2.0 definíció, tervezési irányelvek, technológia példák A **web 2.0** (vagy **webkettő**) kifejezés olyan [internetes](https://hu.wikipedia.org/wiki/Internet) szolgáltatások gyűjtőneve, amelyek elsősorban a közösségre épülnek, azaz a felhasználók közösen készítik a tartalmat vagy megosztják egymás információit. Ellentétben a korábbi szolgáltatásokkal, amelyeknél a tartalmat a szolgáltatást nyújtó fél biztosította (például a [portáloknál](https://hu.wikipedia.org/w/index.php?title=Port%C3%A1l&action=edit&redlink=1)), webkettes szolgáltatásoknál a szerver gazdája csak a keretrendszert biztosítja, a tartalmat maguk a felhasználók töltik fel, hozzák létre, osztják meg vagy véleményezik. A felhasználók jellemzően kommunikálnak egymással, és kapcsolatokat alakítanak ki egymás között. Az [interaktivitás](https://hu.wikipedia.org/wiki/Interaktivit%C3%A1s) és a fogyasztók egymással folytatott kommunikációja miatt napjainkban alig van olyan oldal, amely köré ne szerveződne valamilyen közösség.
* [Közösségi oldalak](https://hu.wikipedia.org/wiki/Ismerets%C3%A9gi_h%C3%A1l%C3%B3zat) (*Social networking*, például: [iWiW](https://hu.wikipedia.org/wiki/IWiW), [Orkut](https://hu.wikipedia.org/wiki/Orkut), [Facebook](https://hu.wikipedia.org/wiki/Facebook),[[2]](https://hu.wikipedia.org/wiki/Web_2.0#cite_note-2) [MySpace](https://hu.wikipedia.org/wiki/MySpace), [LinkedIn](https://hu.wikipedia.org/wiki/LinkedIn),[[3]](https://hu.wikipedia.org/wiki/Web_2.0#cite_note-3) [Mindenki.hu](https://hu.wikipedia.org/w/index.php?title=Mindenki.hu&action=edit&redlink=1" \o "Mindenki.hu (a lap nem létezik)), [Barátikör.com](https://hu.wikipedia.org/wiki/Bar%C3%A1tik%C3%B6r.com), [MyVIP](https://hu.wikipedia.org/wiki/MyVIP))
* [Képmegosztó oldalak](https://hu.wikipedia.org/w/index.php?title=K%C3%A9pmegoszt%C3%B3_oldal&action=edit&redlink=1) (például: [Flickr](https://hu.wikipedia.org/wiki/Flickr), [Indafotó](https://hu.wikipedia.org/w/index.php?title=Indafot%C3%B3&action=edit&redlink=1),[[4]](https://hu.wikipedia.org/wiki/Web_2.0#cite_note-4) [Picasa](https://hu.wikipedia.org/wiki/Picasa), [Photobucket](https://hu.wikipedia.org/wiki/Photobucket), [SmugMug](https://hu.wikipedia.org/wiki/SmugMug), [Zooomr](https://hu.wikipedia.org/wiki/Zooomr%22%20%5Co%20%22Zooomr), [Open Photo Project](https://hu.wikipedia.org/wiki/Open_Photo_Project))
* [Videómegosztó portálok](https://hu.wikipedia.org/w/index.php?title=Vide%C3%B3megoszt%C3%A1s&action=edit&redlink=1) (Például: [YouTube](https://hu.wikipedia.org/wiki/YouTube), [Google Videos](https://hu.wikipedia.org/w/index.php?title=Google_Videos&action=edit&redlink=1), [IndaVideó](https://hu.wikipedia.org/w/index.php?title=IndaVide%C3%B3&action=edit&redlink=1))
* [Blogok](https://hu.wikipedia.org/wiki/Blog), [mikroblogok](https://hu.wikipedia.org/wiki/Mikroblog) (például [Twitter](https://hu.wikipedia.org/wiki/Twitter), [Jaiku.com](https://hu.wikipedia.org/w/index.php?title=Jaiku.com&action=edit&redlink=1), [Plurk](https://hu.wikipedia.org/wiki/Plurk))
* Online irodai alkalmazások (Például: [Google Calendar](https://hu.wikipedia.org/wiki/Google_Calendar), [Google Docs & Spreadsheets](https://hu.wikipedia.org/wiki/Google_Docs_%26_Spreadsheets), [Zoho](https://hu.wikipedia.org/w/index.php?title=Zoho&action=edit&redlink=1)[[5]](https://hu.wikipedia.org/wiki/Web_2.0#cite_note-5) [ThinkFree Online](https://hu.wikipedia.org/w/index.php?title=ThinkFree_Online&action=edit&redlink=1" \o "ThinkFree Online (a lap nem létezik)))
* [Wikipédia](https://hu.wikipedia.org/wiki/Wikip%C3%A9dia) és más [wikik](https://hu.wikipedia.org/wiki/Wiki) (szabadon szerkeszthető ismerettárak)
* [Fórumok](https://hu.wikipedia.org/wiki/F%C3%B3rumrendszer)
* [Aukciós oldalak](https://hu.wikipedia.org/wiki/Aukci%C3%B3s_oldal) (például: [eBay](https://hu.wikipedia.org/wiki/EBay), [Vatera](https://hu.wikipedia.org/wiki/Vatera))

 DHTML, AJAX DHTML allows authors to add effects to their pages that are otherwise difficult to achieve. In short words: scripting language is changing the [DOM](https://en.wikipedia.org/wiki/Document_Object_Model) and page style.

Simply put, DHTML is the combination of HTML, CSS and JavaScript.

* Animate text and images in their document, independently moving each element from any starting point to any ending point, following a predetermined path or one chosen by the user.
* Embed a ticker that automatically refreshes its content with the latest news, stock quotes, or other data.
* Use a form to capture user input, and then process, verify and respond to that data without having to send data back to the server.
* Include rollover buttons or drop-down menus.
* AJAX is **a developers dream**, because you can:

Update a web page without reloading the page
Request data from a server - after the page has loaded
Receive data from a server - after the page has loaded
Send data to a server - in the background Az **Ajax** (Asynchronous [JavaScript](https://hu.wikipedia.org/wiki/JavaScript) and [XML](https://hu.wikipedia.org/wiki/XML)) interaktív [webalkalmazások](https://hu.wikipedia.org/wiki/Webalkalmaz%C3%A1s) létrehozására szolgáló webfejlesztési technika. A [weblap](https://hu.wikipedia.org/wiki/Weblap) kis mennyiségű [adatot](https://hu.wikipedia.org/wiki/Adat) cserél a [szerverrel](https://hu.wikipedia.org/wiki/Szerver) a háttérben, így a lapot nem kell újratölteni minden egyes alkalommal, amikor a felhasználó módosít valamit. Ez növeli a honlap [interaktivitását](https://hu.wikipedia.org/wiki/Interaktivit%C3%A1s), sebességét és használhatóságát.

Az Ajax a következő technikák kombinációja:

* [**XHTML**](https://hu.wikipedia.org/wiki/XHTML) (vagy [HTML](https://hu.wikipedia.org/wiki/HTML)) és [CSS](https://hu.wikipedia.org/wiki/CSS) a tartalom leírására és formázására.
* [**DOM**](https://hu.wikipedia.org/wiki/DOM) kliens oldali script nyelvekkel kezelve a dinamikus megjelenítés és a már megjelenített információ együttműködésének kialakítására.
* [**XMLHttpRequest**](https://hu.wikipedia.org/wiki/XMLHttpRequest) objektum az adatok aszinkron kezelésére a kliens és a [webszerver](https://hu.wikipedia.org/wiki/Webszerver) között. Néhány Ajax keretrendszer esetén és bizonyos helyzetekben [IFrame](https://hu.wikipedia.org/wiki/IFrame)-et használnak XMLHttpRequest objektum helyett.
* [**XML**](https://hu.wikipedia.org/wiki/XML) formátumot használnak legtöbbször az adattovábbításra a kliens és a szerver között, bár más formátumok is megfelelnek a célnak, mint a formázott HTML vagy a sima szöveg.

9. előadás:

 Kereső optimalizálás (SEO), on-site, off-site SEO, a SEO és a Web 2.0 kapcsolata A **keresőoptimalizálás** (angolul *Search Engine Optimization*, azaz **SEO**) a [weblap](https://hu.wikipedia.org/wiki/Weblap) [keresőmotorokban](https://hu.wikipedia.org/wiki/Keres%C5%91motor) való megjelenésének javítását jelenti "normál" avagy nem fizetett keresési eredményekben. Általánosságban minél előbb, azaz jobban rangsorolva a találati oldalon és többször jelenik meg egy oldal a találati listában, annál több látogatást szerez a kereső használóitól. A SEO különböző területeket célozhat meg, mint például a [képkeresés](https://hu.wikipedia.org/w/index.php?title=K%C3%A9pkeres%C3%A9s&action=edit&redlink=1), [videókeresés](https://hu.wikipedia.org/w/index.php?title=Vide%C3%B3keres%C3%A9s&action=edit&redlink=1), [helyi keresés](https://hu.wikipedia.org/w/index.php?title=Helyi_keres%C3%A9s&action=edit&redlink=1) vagy [tudományos keresés](https://hu.wikipedia.org/w/index.php?title=Tudom%C3%A1nyos_keres%C3%A9s&action=edit&redlink=1" \o "Tudományos keresés (a lap nem létezik)),[[1]](https://hu.wikipedia.org/wiki/Keres%C5%91optimaliz%C3%A1l%C3%A1s#cite_note-aseo-1) újságcikk-keresés avagy iparág-specifikus keresés.

 Page Rank szavazós modell, sztochasztikus szörföző modell

A **PageRank** az [informatikában](https://hu.wikipedia.org/wiki/Informatika) egy olyan [algoritmus](https://hu.wikipedia.org/wiki/Algoritmus), amely [hiperlinkekkel](https://hu.wikipedia.org/wiki/Hiperlink) összekötött [dokumentumokhoz](https://hu.wikipedia.org/w/index.php?title=Dokumentum&action=edit&redlink=1) számokat rendel azoknak a hiperlink-hálózatban betöltött szerepe alapján. (Ezt a számot szintén PageRanknek nevezik.) A PageRank a [Google](https://hu.wikipedia.org/wiki/Google_keres%C5%91) [internetes](https://hu.wikipedia.org/wiki/Internet) [keresőmotor](https://hu.wikipedia.org/wiki/Keres%C5%91motor) legfontosabb eleme. [Larry Page](https://hu.wikipedia.org/wiki/Larry_Page) és [Sergey Brin](https://hu.wikipedia.org/wiki/Sergey_Brin) (a Google alapítói) fejlesztették ki [1998](https://hu.wikipedia.org/wiki/1998)-ban a [Stanford Egyetemen](https://hu.wikipedia.org/wiki/Stanford_Egyetem).

A Google arra a feltételezésre épít, hogy a [weboldalak](https://hu.wikipedia.org/wiki/Weboldal) készítői általában azokra az oldalakra linkelnek a saját lapjukról, amiket jónak tartanak, vagyis minden hiperlink felfogható egy-egy szavazatként a céloldalra. Minél több szavazatot kap egy oldal, annál fontosabb, de azt is figyelembe kell venni, hogy a szavazatot leadó oldal mennyire fontos. (Ez egy [rekurzív definíció](https://hu.wikipedia.org/w/index.php?title=Rekurz%C3%ADv_defin%C3%ADci%C3%B3&action=edit&redlink=1): az a fontos oldal, amire fontos oldalak mutatnak.) A PageRank a fontosság számszerűsítése.

A PageRanket úgy is felfoghatjuk, mint annak a [valószínűségét](https://hu.wikipedia.org/wiki/Val%C3%B3sz%C3%ADn%C5%B1s%C3%A9g), hogy odatalálunk az oldalra. A valószínűséget a [sztochasztikus](https://hu.wikipedia.org/w/index.php?title=Sztochasztikus&action=edit&redlink=1) szörfözővel modellezzük, aki a weben bolyong, és minden lépésben véletlenszerűen, [egyenletes eloszlás](https://hu.wikipedia.org/wiki/Egyenletes_eloszl%C3%A1s) szerint kiválaszt egyet az oldalon található linkek közül, és azon halad tovább. (Más szóval [véletlen bolyongást](https://hu.wikipedia.org/w/index.php?title=V%C3%A9letlen_bolyong%C3%A1s&action=edit&redlink=1) végez a hiperlinkek alkotta [irányított gráfon](https://hu.wikipedia.org/w/index.php?title=Ir%C3%A1ny%C3%ADtott_gr%C3%A1f&action=edit&redlink=1).) Hogy ne essen csapdába valamelyik olyan részgráfban, amiből nem vezet kifelé link, a modellt kiegészítjük egy további elemmel: a szörföző minden lépésben *1-d* valószínűséggel elunja magát, és egy (egyenletes eloszlás szerint) véletlenszerűen választott weblapra ugrik.  A sztochasztikus szörföző modellel definiált PageRank tehát egy valószínűségi eloszlás lesz: egy oldal PageRankje annak a valószínűsége, hogy nagyon sok véletlenszerű kattintás (és ugrás) után éppen arra az oldalra érkezünk. (A PageRank reciproka az oldal várható visszatérési ideje, azaz annak a [várható értéke](https://hu.wikipedia.org/wiki/V%C3%A1rhat%C3%B3_%C3%A9rt%C3%A9k), hogy az oldalról elindulva hány lépés múlva érünk vissza oda.)

 Vakbarát honlapok

Érdemes a lehető legegyszerűbbre tervezni a vakos lapokat, mert a vak felhasználók sokféle rendszert használnak és fognak használni Magyarországon.

Célszerű olyan web lapot tervezni, amely nagyon puritán kinézetű, de a szöveges része tartalmas. Ez biztosan jó lesz mindenféle felhasználónak, bármilyen rendszert is használjon.

Egy nagyon jó vakos web felület sem tud olyan jó lenni, mint egy személyre szabott, önálló intelligenciával rendelkező számítógép. Hosszan olvasgatni egy könyvet hálózaton nem lehet. Tekintettel kell lenni arra, hogy a vak felhasználók sok esetben a letöltött állományokat akarják offline felolvastatni.

Ha a navigációt segítő struktúrált állományokat fogja Windows rendszerben valaki vakon olvasni, azt is inkább off-line fogja tenni. A Windows-os vakos programfejlesztésben elindult egy olyan irányzat, hogy a Microsoft-os programokat, mint egy motort használják csak, és külön programot fejlesztenek látássérülteknek, ilyen a fentebb már említett, fejlesztés alatt lévő LookOUT rendszer.

Off-line működő, nagyon egyszerű interfésszel rendelkező vakos olvasó programra mindenképpen szüksége van a vakoknak Windows alatt is.

Milyen legyen egy vakos web-es felület? Karakteres. Ha a tervező nem használ grafikát, akkor sokkal könnyebb jó vakos felületet tervezni, mintha egyszerre szeretné a látó és a vak felhasználó igényeit kielégíteni. A külön vakos felületen ezt meg lehet oldani.

Az oldalakat, a látó oldalakhoz hasonlóan, hierarchikus struktúrába kell rendezni. Javasoljuk, hogy az egyes választási lehetőségek az egyes lapokon mindig egymás alá kerüljenek. Ezekből, ha lehet, hétnél ne legyen több egy oldalon, így azt az idősebb emberek is jól meg fogják tudni jegyezni. A linkekhez tartozó szövegek legyenek legalább teljes mondatok. Természetesen az olyan típusú linkek, ahol a név csak azt tartalmazza, hogy "További információk itt", nem megfelelőek, mert az ernyőolvasók, ha a linkeken haladnak végig tabulátorral, akkor csak magát a linket olvassák fel.

 W3C WAI, felhasználói célcsoportok

**Web accessibility means that people with disabilities can use the Web**. More specifically, Web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web. Web accessibility also benefits others, including[older people](https://www.w3.org/WAI/bcase/soc.html#of) with changing abilities due to aging.

## WCAG conformance and user experience

[WCAG conformance](http://www.w3.org/TR/WCAG20/#conformance) focuses primarily on the web page as a unit of conformance. So to claim a certain level of accessibility conformance, a page must meet all WCAG Success Criteria up to and including that level. However, conformance also covers the concept of processes:

When a Web page is one of a series of Web pages presenting a process (i.e., a sequence of steps that need to be completed in order to accomplish an activity), all Web pages in the process conform at the specified level or better.

This acknowledges the importance of task completion in evaluating accessibility – but leaves you, as an author, with the responsibility of deciding and defining which processes need to be supported. If you substitute the concept of “processes” with “goal success”, you can see how this approach can make a connection between technical conformance and accessible user experience.

#  W3C WCAG 2, tervezési irányelvek, A, AA, AAA szintű ajánlások

## Perceivable

* Provide [**text alternatives**](http://www.w3.org/WAI/WCAG20/quickref/#text-equiv) for non-text content.
* Provide [**captions and other alternatives**](http://www.w3.org/WAI/WCAG20/quickref/#media-equiv) for multimedia.
* Create content that can be [**presented in different ways**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#content-structure-separation),
including by assistive technologies, without losing meaning.
* Make it easier for users to [**see and hear content**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#visual-audio-contrast).

## Operable

* Make all functionality available from a [**keyboard**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#keyboard-operation).
* Give users [**enough time**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#time-limits) to read and use content.
* Do not use content that causes [**seizures**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#seizure).
* Help users [**navigate and find content**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#navigation-mechanisms).

## Understandable

* Make text [**readable and understandable**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#meaning).
* Make content appear and operate in [**predictable**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#consistent-behavior) ways.
* Help users [**avoid and correct mistakes**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#minimize-error).

## Robust

* Maximize[**compatibility**](http://www.w3.org/WAI/WCAG20/quickref/Overview.php#ensure-compat) with current and future user tools.

# W3C Web Content Accessibility Guidelines 2.0

* Conformance Level "A": all Level A success criteria are satisfied;
* Conformance Level "AA": all Level A and AA success criteria are satisfied;
* Conformance Level "AAA": all Level A, AA, and AAA success criteria are satisfied;

The Web Content Accessibility Guidelines 2.0 explain how to make Web content accessible to people with disabilities. Conformance to these guidelines will help make the Web more accessible to users with disabilities and will benefit all users.

 HTML5+CSS3+JavaScript

10. előadás:

 HTML5+CSS3+JavaScript főbb újdonságai

 3D DEMO

 honlapok mobil eszközökön

### **Mik a legfontosabb szempontok a weboldal mobilra optimalizálása során?**

Az egyik **a képernyő mérete és felbontása**. Ha weboldala nem igazodik az okostelefon vagy tablet képernyőjének méretéhez, folyamatosan görgetni kell vízszintesen, ez hamar idegesítő válik és látogatói hamar elhagyják weboldalát.
A tabletek és okostelefonok nagyon változatos felbontással és képátlóval kaphatók. A mobilra optimalizált weboldalt úgy készítjük el, hogy **a különböző átmérőjű és képátlójú eszközön, a weboldal elemei igazodjanak a kijelző méretéhez**. Ugyanez vonatkozik a megjelenő betűk méretére.  Hiába a legjobb tartalom, ha weboldalunk szövege a kis betűméret miatt olvashatatlan telefonon.

A hagyományos weboldalak elrendezése két vagy háromhasábos. **A mobilra optimalizált weboldalnál az egy hasábos elrendezést célszerű választanunk**. Ettől még az asztali gépen megjelenő honlap maradhat több hasábos, a lényeg, hogy amikor okostelefonról vagy tabletről látogatják a honlapot, ezeket a hasábokat elcsúsztassa egymás alá,  hogy elkerüljük a vízszintes görgetést.

**Nagyon fontos odafigyelni a navigációra.** A hagyományos weboldalaknál a menüpontok általában felül vízszintesen egymás mellett jelennek meg .  Ez a megoldás okostelefonokon kimondottan kényelmetlen, ezért vagy **függőlegesen egymás alatt**érdemes megjeleníteni a menüpontokat, vagy egy menü gombot készíteni a fejlécbe, amelyre kattintva egy **lenyíló lista** jelenik meg.

Az érintőképernyős mobil eszközök esetén az ujjal történő kattintás sokkal pontatlanabb, mint amikor egeret használunk. **A gombok méretét, illetve a linkek hosszát** úgy kell megválasztanunk, hogy az ujjal történő kattintásnál egyértelmű legyen, melyik gombra vagy linkre kattintottak.

### **Milyen módszerek vannak egy weboldal mobilra optimalizálására?**

**A már meglévő honlapok esetében is van megoldás**: a régi honlapot érintetlenül hagyjuk, és elkészítjük annak okostelefonokra és tabletekre optimalizált változatát.
Az új weblapok készítésénél még könnyebb a dolgunk: a reszponzív (responsive) webdesign segítségével egy olyan oldalt fejlesztünk, amely könnyen kezelhető asztali számítógépeken és mobil eszközökön egyaránt, és a designja is vonzó.

### **A reszponzív design lényege**

Az újfajta technológia jóvoltából a weblap tartalma a honlapot böngésző eszköztől függetlenül elérhető, tehát **ugyanaz a tartalom jelenik meg egy számítógépen, tableten vagy okostelefonon is**, csak a honlap designja, felépítése igazodik a kijelző méretéhez. A módszer alkalmazását a Google is ajánlja, ugyanis így a honlapok betöltődése gyorsabb, és a tartalomduplikáció is elkerülhető.

### **A reszponzív webdesign előnyei**

• költséghatékony
• keresőbarát, mert gyorsabb a honlap betöltődésének sebessége, és csak egy HTML-kódot generál
• időt takarítunk meg vele, hiszen a tartalmat csak egyszer kell feldolgozni
• bizalmat ébreszt látogatóinkban, vásárlóinkban, mert honlapunk mindenféle eszközön esztétikusan jelenik meg
• minden modern böngésző támogatja

Első és legfontosabb szempont, hogy elég **gyorsan töltött-e be a honlap**. A mobil felhasználók (sem) fogják megvárni, ha 8-10 másodperc alatt nem töltődik be a honlap. **2-4 másodperc** az elfogadható átlag.

**Látja-e teljes egészében a megadott weboldalt**? Rossz esetben a honlap “kilóg” a képernyőből és nem olvasható.

Ha látja, a következő szempont, hogy **olvasható-e a megjelenített tartalom, betűtípus**? Lehetséges, hogy nagyítania kell a képernyőt, ezt kerülni érdemes.
A jó mobil weboldal **könnyen navigálható**. A tartalom “elfér” a mobil kijelzőn és **jól olvasható betűtípussal** írják a szövegét. A képek jó felbontásúak, szintén elférnek a képernyőn.
**A menüpontok listában azonnal látszanak** vagy egy jól felismerhető gombbal elérhetők.

 képernyő méret és képernyő felbontás

 reszponzív felületek (eredete, funkciója, alap elrendezések)

Now a days, there's a lot of scope for platform independent applications. Benefits of a cross-platform HTML5 apps -

1. If your app is not a resource intensive one (Which requires lots of resources on your device) then you can use HTML5 based apps which look very similar to native applications.
2. **Build Beautiful User Interface in a very short time -**You can use the existing html5 and javascript libraries on github to make your app look too professional. You can build a very beautiful User Interface very fast.
3. **Portability:**While developing your web app, keep the front end and backend completely separate, so that you can instantly port your website to a mobile app.
4. **Screen Compatibility:**Using responsive methods of HTML5 helps you to make UI very easily. You don't have to go through the headache of screen adjustments for various devices. Use the responive (Bootstrap) method and your app will be compatible with devices of all screen sizes.
5. **Easy to develop:** Developing a native application is a tough and tedious task. It takes a lot of time. Using html5 and javascript on the front-end lets you develop fast, and it also provides you with existing libraries which makes your job too easy.
6. **Use the same code everywhere:**Run your application on android, ios or blackberry, anywhere with little or no change in your code. Never bother about the make and size of the device.

 'Mobile first' tervezési irányelv

1. Egyszerű navigációval készülnek
2. Böngésző-függetlenek
3. Gyorsan betöltődnek
4. Jól jelenítik meg a képeket
5. Csak a lényeget tartalmazzák

By starting mobile first with your responsive website you are able to make the tough decisions about the content. Mobile first often lends itself to a linear layout model, although this is a general truth and not necessarily a rule.

The great thing about this approach is it forces you into a content hierarchy. The most important piece of content is on top, followed by the next, followed by the next. Unlike a desktop approach where you can put two pieces of content side by side for equal footing, a mobile approach means one is above the other.

 mobil eszközökre optimalizált felületek

 haszon > fájdalom

 alapgondolkodás (default thinking)

 mobiltelefonokról általánosságban

11. előadás:

 mobil böngészők

A **mobile browser** is a [web browser](https://en.wikipedia.org/wiki/Web_browser) designed for use on a [mobile device](https://en.wikipedia.org/wiki/Mobile_device) such as a [mobile phone](https://en.wikipedia.org/wiki/Mobile_phone) or [PDA](https://en.wikipedia.org/wiki/Personal_digital_assistant). Mobile browsers are optimized so as to display Web content most effectively for small screens on portable devices. Mobile browser software must be small and efficient to accommodate the low memory capacity and low-bandwidth of wireless handheld devices. Typically they were stripped-down web browsers, but some more modern mobile browsers can handle more recent technologies like [CSS 2.1](https://en.wikipedia.org/wiki/Cascading_Style_Sheets), [JavaScript](https://en.wikipedia.org/wiki/JavaScript), and [Ajax](https://en.wikipedia.org/wiki/Ajax_%28programming%29).

 okostelefonok piaci részesedése



 Andorid általános tulajdonságok

Az **Android** [Linux kernelt](https://hu.wikipedia.org/wiki/Linux_%28rendszermag%29) használó [mobil operációs rendszer](https://hu.wikipedia.org/wiki/Mobil_oper%C3%A1ci%C3%B3s_rendszer),[[7]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-AndroidOverview-7) elsősorban érintőképernyős mobil eszközökre ([okostelefon](https://hu.wikipedia.org/wiki/Okostelefon),[táblagép](https://hu.wikipedia.org/wiki/T%C3%A1blag%C3%A9p)) tervezve. Fejlesztését az [Android, Inc.](https://hu.wikipedia.org/w/index.php?title=Android,_Inc.&action=edit&redlink=1) kezdte el, amit 2005-ben a [Google](https://hu.wikipedia.org/wiki/Google_Inc.) felvásárolt,[[8]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-8) majd az [Open Handset Alliance](https://hu.wikipedia.org/w/index.php?title=Open_Handset_Alliance&action=edit&redlink=1)nevű szövetség folytatta.[[9]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-AndroidAnnouncement-9) A fejlesztők [Java](https://hu.wikipedia.org/wiki/Java_%28programoz%C3%A1si_nyelv%29) nyelven írhatnak rá [menedzselt kódot](https://hu.wikipedia.org/w/index.php?title=Menedzselt_k%C3%B3d&action=edit&redlink=1), az eszközt a Google által fejlesztett Java programkönyvtárakon keresztül vezérelve.[[10]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-10) Az [első androidos telefon](https://hu.wikipedia.org/w/index.php?title=HTC_Dream&action=edit&redlink=1) 2008 októberében jelent meg.[[11]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-11)

Az **Android** platform abból a célból született, hogy egységes nyílt forrású operációs rendszere legyen a mobil eszközöknek (és itt elsősorban az okostelefon kategóriát kell érteni, mintsem egyszerű mobiltelefonokat). Az elképzelés alapja egy Linux alapú operációs rendszer volt, amelyet úgy alakítanak át, hogy képes legyen problémák nélkül kezelni a mobil eszközök integrált hardvereit (érintőképernyő, WiFi, HSDPA, Bluetooth stb.). Az első lépéseknél nem volt szó Java nyelvről, azonban a Google 2005 júliusában megvásárolta az Android nevű céget, és új irányt adott a fejlesztésnek: a Linux kernel fölé egy virtuális gép került, amely a felhasználói felület kezeléséért és az alkalmazások futtatásáért felelős.

Ez nem ment egyik napról a másikra, és a Google az első évben igen csöndesen dolgozott, 2007 elején kezdtek kiszivárogni olyan hírek, hogy a Google belép a mobil piacra. Az iparági pletykák végül igaznak bizonyultak, bár sok esetben túlzó állításokat és rémhíreket lehetett olvasni a híroldalakon. 2007. november 5-én az Open Handset Alliance bejelentette az Android platformot, kiknek tagjai között több tucat mobil technológiában érdekelt céget megtalálunk, akik érdekeltek egy szabad és nyílt forrású platform bevezetésében.

Napjainkra az Android platform iránt a mobiltelefon- és a tabletgyártók izgalmát leszámítva nagy érdeklődés mutatkozik a gépjárművek fedélzeti számítógépét és navigációját szállító cégek, illetve az ipari automatizálás irányából is, hiszen minden olyan helyen kényelmes az Android, ahol alapvetően kicsi a kijelző, limitáltak az erőforrások és az adatbevitel nem egérrel és/vagy billentyűzettel történik.

A platform egyszerűnek tűnik, és messziről nézve egyszerű is, vessünk egy pillantást az alábbi ábrára:

Mint láthatjuk, a platform alapját a vörös színnel jelölt Linux kernel adja, amely tartalmazza a hardver által kezelendő eszközök meghajtó programjait. Ezeket azon cégek készítik el, amelyek az Android platformot saját készülékükön használni kívánják, hiszen a gyártónál jobban más nem ismerheti a mobil eszközbe integrált perifériákat. Ez a kis méretű kernel adja a memória kezelését, a folyamatok ütemezését és az alacsony fogyasztást elősegítő teljesítmény-kezelést is.

A kernel szolgáltatásait használják a Linux rendszerekben meglévő különféle programkönyvtárak, mint a libc, az SSL vagy az SQLite; ezek C/C++ nyelven vannak megvalósítva, és a Linux kernelen futnak közvetlenül. Részben ezekre épül a [Dalvik virtuális gép](https://hu.wikipedia.org/wiki/Dalvik), amely egyáltalán nem kompatibilis a Sun virtuális gépével, teljesen más az utasítás készlete, és más bináris programot futtat. A Java programok nem egy-egy .class állományba kerülnek fordítás után, hanem egy nagyobb Dalvik Executable formátumba, amelynek kiterjesztése .dex, és általában kisebb, mint a forrásul szolgáló .class állományok mérete, mivel a több Java fájlban megtalálható konstansokat csak egyszer fordítja bele a Dalvik fordító. A virtuális gép más, mint a Java alatti megszokott virtuális gép, vagyis a Java csak mint nyelv jelenik meg!

A kék színnel jelölt részekben már csak Java forrást találunk, amelyet a virtuális gép futtat, s ez adja az Android lényegét: a látható és tapintható operációs rendszert, illetve a futó programokat. A virtuális gép akár teljesen elrejti a Linux által használt fájlrendszert, és csak az Android Runtime által biztosított fájlrendszert láthatjuk.[[13]](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29#cite_note-JavaForum.hu_-_Android_fejleszt.C3.A9s-13)

 Android fragmentáció problémája

In [computer programming](https://en.wikipedia.org/wiki/Computer_programming), **fragmentation** is when a combination of software and hardware do not provide a consistent, top-level experience for the vast majority of its user-base. The cause is when a finite combination of both software and hardware is made available to consumers.

 term being used in the [Android](https://en.wikipedia.org/wiki/Android_%28operating_system%29) development community is *Android fragmentation*.[[3]](https://en.wikipedia.org/wiki/Fragmentation_%28programming%29#cite_note-androiddef-3) Fragmentation within Android is when a variety of versions of the Android platform, combined with a mixture of hardware result in the inability for some devices to properly run certain applications.[[3]](https://en.wikipedia.org/wiki/Fragmentation_%28programming%29#cite_note-androiddef-3) Despite [Google](https://en.wikipedia.org/wiki/Google) upgrading its Android operating system to version 4.4, also known as *KitKat*, users continue to use the earlier versions of the operating system, primarily *Gingerbread*. The cause is primarily because hardware manufacturers of the devices are not able to upgrade to the later operating system for a number of reasons. As a result, applications written for one version will not operate consistently on the other, and vice versa

Fragmentation also becomes a problem when your hardware becomes outdated. However, this is not just unique to Android. Apple also runs into this, when their older iPhones can’t run the latest and greatest version of iOS. Windows machines also run into this when the machines no longer meet the minimum requirements to run the OS. Of course, one major difference in at least this regard is that the PC can be upgraded with a newer CPU, GPU, more RAM, additional storage, newer motherboard, etc…

 Android 6.0 újdonságok

## Google has added a memory manager

One of the complaints about Android Lollipop was that its memory-hungry nature introduced all sorts of performance problems in phones without absolutely loads of RAM.

## The clock has gone all stylish

Lollipop upped Android’s style game, and Marshmallow tries to take that a step further with a redesigned clock. It’s little more than a font shuffle, but does give the new software a bit of a different, sharper visual personality.

## App search bar and favourites

In order to help those who think the new apps menu is worse, and there will be some, there are some extra features to the apps area. First, there’s a search bar up at the top.

## The apps menu is crazy-different

One Android Marshmallow change all of you will notice is the way the new apps menu works. It’s totally different to Lollipop and the other previous versions of Android.

## attery optimisation now on a by-app basis

Android only really started to embrace proper a battery-saving mode with Android Lollipop, even though custom skins have had such features for years now. Android Marshmallow adds a whole new battery area called ‘optimisation’.

## The volume controls have changed yet againFingerprint scanners supported as standard

## USB Type-C/3.1 support

Platform független fejlesztési lehetőség HTML5 alapon

Given the hype and buzz surrounding [HTML5](http://mashable.com/category/html5/), it would be easy to believe that it is a technology that will do your laundry, mow the lawn and make you dinner. In truth, HTML5 isn't the second coming, and it isn't an officially ratified standard — yet. The spec continues to edge closer to completion, however, and when combined with JavaScript and CSS3, HTML5 can do some really incredible things.

This is particularly true for mobile devices. A *de facto* requirement for any modern mobile operating system is the inclusion of a modern HTML5-compliant web browser. The leading modern mobile platforms — iOS and Android — both use WebKit as their bases. Likewise, BlackBerry and HP/Palm are also using WebKit and Microsoft is going to release a mobile version of Internet Explorer 9 for Windows Phone 7.

What this means is that out of the box, modern smartphones and tablets support the bells and whistles that make HTML5 so special. It also means that developers can feel free to use those technologies when creating their applications and not have to worry that the device itself won't support a particular function.

 DEMO (HTML5, Android Studio, Monkey tool)

12. előadás:

 Féléves beszámolók Ok.

1. gyakorlat:

 Bemutatkozás

 Grafikus felhasználói interfészek

 Fitts törvényének levezetése és alkalmazása

2. gyakorlat:

 A gyakorlati feladatok ismertetése, feladatválasztás, csoport alkotás

 [**fóliák**](http://speechlab.tmit.bme.hu/hci/hci_gyak2_2015_hu.pdf)

 Tervezés

 Tervezés: Időjárás előrejelző alkalmazás Win8-ra

 Hasonló honlapok, hasonló Win8 alkalmazások

 metnet.hu oldal elemzése és újratervezése

 Dizájn tervek

 Felhasználói igényfelmérés

 "[**Időjárás mindenkinek**](http://apps.microsoft.com/windows/en-us/app/idojaras-mindenkinek/856cd991-0d35-40ad-badc-d52883a6315b)" Win8 alkalmazás iteratív fejlesztésének bemutatása

 Többnyelvű változat

3. gyakorlat: A **Project Glass**, vagy közismertebb nevén **Google Glass** egy kutatási és fejlesztési program, melyet azzal a céllal hoztak létre, hogy kifejlesszenek egy kibővített valóságot megjelenítő fejre illeszthető kijelzőt (HMD-head-mounted display).[[1]](https://hu.wikipedia.org/wiki/Google_Glass#cite_note-1) A Project Glass szemüveg az [okostelefonok](https://hu.wikipedia.org/wiki/Okostelefon) kijelzőjén megjelenő adatokhoz és ikonokhoz hasonlóan jeleníti meg a különböző információkat. Az eszköz hangvezérléssel működik, nincs szükség kézi irányításra, valamint képes kapcsolódni az [internethez](https://hu.wikipedia.org/wiki/Internet) is.[[2]](https://hu.wikipedia.org/wiki/Google_Glass#cite_note-2) Az Android operációs rendszert futtató szemüveg számos funkcióval bír és minimalista stílusú (egy vékony alumínium keretből áll, két orrnyeregvédővel ellátva).[[3]](https://hu.wikipedia.org/wiki/Google_Glass#cite_note-NYT-3)

 Okosszemüveg, kiterjesztett valóság

 SmartGlass Az okosszemüveg tehát egy mini számítógép és szemüveg kombinációja. Ehhez tartozik valamilyen kijelző is, amin sokféle információt és képet megjeleníthetünk. A lényeg az, hogy nem kell előkapni a zsebünkből egy mobilt például, és még csak félre se kell néznünk, mert minden a látóterünkben van.

 [**Top 5 Smart Glasses 2014**](https://www.youtube.com/watch?v=jXNOQsO-flc)

 Google Glass, Vuzix M100, Microsoft HoloLens, ODG R-7, Meta Space Glasses, Oculus Rift, ... [optical head-mounted display](https://en.wikipedia.org/wiki/Optical_head-mounted_display)

 Augmented Reality A **kiterjesztett valóság** ([angolul](https://hu.wikipedia.org/wiki/Angol_nyelv) *augmented reality, AR)* a valóság egyfajta virtuális kibővítése, amikor a mobilunk kamerájával szétnézve egy adott környéken megjelenik az éppen a kamerában látható boltok nyitvatartása vagy akár az adott irányban levő (éppen nem is látható) üzletek leírása és távolságuk.

A legnépszerűbb, valóságkiterjesztő szoftverek futtatására alkalmas eszközök jelenleg a [Google](https://hu.wikipedia.org/wiki/Google%2C_Inc.) [Android](https://hu.wikipedia.org/wiki/Android_%28oper%C3%A1ci%C3%B3s_rendszer%29) operációs rendszert futtató telefonok (például a hazánkban is kapható HTC G2 Hero) és az [Apple](https://hu.wikipedia.org/wiki/Apple) [iPhone](https://hu.wikipedia.org/wiki/IPhone) (3GS és 4 modellek). Fontos előfeltétele a működésnek, hogy az adott készülék képes legyen a helymeghatározásra (például [GPS](https://hu.wikipedia.org/wiki/Global_Positioning_System) segítségével) és arra, hogy „lássa”, merre néznek vele (iránytű mód).

 Definíciók Általában android alapú eszközök.

 [**Kiterjesztett valóság: ARmedia**](https://www.youtube.com/watch?v=9a63op7Q-6I)

 [**Raktározás: SAP & Vuzix**](https://www.youtube.com/watch?v=9Wv9k_ssLcI)

 [**Sportolás: Recon Jet**](https://www.youtube.com/watch?v=UbTsXRXj0ns)

 [**Lakberendezés: IKEA**](https://www.youtube.com/watch?v=vDNzTasuYEw)

 [**Reklám: Blippar**](https://www.youtube.com/watch?v=5BO5ihpd2zg)

 [**Turisztika, építészet: ARmedia**](https://www.youtube.com/watch?v=C5_xjDbrbR4)

 [**Játék: Zombies Everywhere**](https://www.youtube.com/watch?v=59An70D3xg8)

 Felhasználói interfész tervezése okosszemüvegre

 Vuzix M100 tulajdonságai Holografikus optika. A lencse elektronikusan sötétül vagy világosodik a kinti fényviszonyok függvényében.

 Adatbevitel, vezérlés Touchpad, Gombok, Hangvezérlés, Gesztus felismerés, Szemkövetés

 RunAssistant

 Mikrointerakciók

 PresHelper

 [**Prototype-On-Paper**](https://popapp.in/)

 Okosszemüvegek jövője

4. gyakorlat: Lásd előadás résznél.

 Kiértékelés

 [**10 Usability Heuristics**](https://www.youtube.com/watch?v=hWc0Fd2AS3s)

 Kiértékelés (AALFred)

 Angol, francia, lengyel, magyar, portugál változat tesztelése

 Kiértékelés (Időjárás mindenkinek)

 Előkészületek

 Feladatok

 Eredmények

 Újratervezés

5. gyakorlat

 Folyamatos beszédü diktáló rendszer

 Large Vocabulary Continuous Speech Recognition (LVCSR) fogalma Much slower processing, since the basic unit is a set of words (bi-grams, tri-grams etc.), it needs to have hundreds of thousands of words to match the audio against. The output however is a stream of words, making it richer to work with. It can surface new business issues, the queries are much faster, and the accuracy is higher than the phonetic approach[*[citation needed](https://en.wikipedia.org/wiki/Wikipedia%3ACitation_needed%22%20%5Co%20%22Wikipedia%3ACitation%20needed)*]. Most importantly because the complete semantic context is in the index it is possible to find and focus on business issues very rapidly.

Both [acoustic modeling](https://en.wikipedia.org/wiki/Acoustic_model) and [language modeling](https://en.wikipedia.org/wiki/Language_model) are important parts of modern statistically-based speech recognition algorithms. Hidden Markov models (HMMs) are widely used in many systems. Language modeling is also used in many other natural language processing applications such as [document classification](https://en.wikipedia.org/wiki/Document_classification) or [statistical machine translation](https://en.wikipedia.org/wiki/Statistical_machine_translation)

### Hidden Markov models

### Dynamic time warping (DTW)-based speech recognition**[[edit](https://en.wikipedia.org/w/index.php?title=Speech_recognition&action=edit&section=5" \o "Edit section: Dynamic time warping (DTW)-based speech recognition)]**

### Neural networks

### Deep Neural Networks and Other Deep Learning Models

 Diktálórendszer szerepe

Rendszerünk jelenleg néhány ezer szavas (kis-közepes méretű) szótár és erősen kötött nyelvi környezet esetén elfogadható hatékonysággal működik, és a nagyszótáros működés elérése céljából jelenleg is folyamatos fejlesztés alatt áll. A felismerő akusztikai szinten a közismert rejtett Markov-technológiát alkalmazza, de kísérleti szinten hasonló eredményeket tudtunk elérni egy neuronhálós megoldással is. A folyamatos diktálás nyelvi szintű algoritmikus támogatása az ún. szó n-gram-okat, különböző simítási módszereket, környezetfüggetlen nyelvtani modellezést, valamint MSD kód (morfoszintaktikai leírás) alapú szabályrendszert foglal magában, valamint támogatja a beszédhangok közötti hasonulások kezelését. Itt további cél a magyar nyelv sajátosságainak (a nem kötött szórendnek és a toldalékolásnak) hatékonyabb modellezése.

A beszédfelismerő akusztikai része a standard beszélőfüggetlen diktálási lehetőség mellett beszélőadaptációs (a beszélő hangjához hozzáigazodó, ezáltal a pontosságot nagymértékben növelni képes) modult is tartalmaz. Emellett jelenleg a különböző beszélőnormalizációs technikák (VTLN, CMN) rendszerbe integrálása is folyamatban van. Eddigi tapasztalataink szerint ezek a módszerek további jelentős javulást tudnak hozni a beszédfelismerési hatásfokban.

 Elvárások

n a rendszer ismerje fel a beszédet (általánosságban: beszédszöveg átalakítás) n beszéddetekció: annak felismerése, hogy beszéd van vagy nincs n zöngés/zöngétlen meghatározás, leginkább csak támogatja a beszédfelismerőket, de néha önálló feladatnak is tekintik n beszélő felismerése n beszélő azonosítása n mintaszöveg felvétele, amit belépéskor el kell mondani n véletlenszerűen kisorsolt minta: sok mintát vesznek fel, és ezek közül egyet véletlenszerűen sorsol a rendszer a belépéskor

 Lehetséges felhasználók

 Müködési környezet

 Megoldási lehetőségek: ki/bekapcsolás, pozicionálás, javítás

6. gyakorlat

 Féléves beszámolók Megvan.