

KONSTANTINUSZ

CONSULT TANÁCSADÓ KFT.

Web- és szoftver-ergonómia

1 TARTALOMJEGYZÉK

1	Tartalomjegyzék	2
2	Bevezetés	3
3	Használhatóság.....	3
3.1	Tanulhatóság	3
3.2	Hatékonyság	4
3.3	Megjegyezhetőség	4
3.4	Hibák.....	4
3.5	Elégedettség	5
4	Felhasználói felületek fontos tervezési szabályai.....	6
5	A mozgás	7
6	Web-es fejlesztés	7
6.1	Nem szeretünk gondolkodni	7
6.2	Ne bonyolítsuk túl a dolgokat.....	8
6.3	Nem olvasunk, pásztázunk!.....	8
7	Összegzés	9
8	Irodalomjegyzék	10

2 BEVEZETÉS

Csakúgy, amint az ergonómia alakítja a munkafolyamatokat, eszközöket az ember számára megfelelővé, kis erőfeszítéssel működtethetővé, úgy illeszti a szoftver-ergonómia a számítógépes rendszereket az ember kognitív és intellektuális tulajdonságaihoz, cselekvési, és észlelési jellemzőihez. Úgy alakítja az információközlés módját, hogy az az ember számára a lehető legkényelmesebb legyen. Azon túl, hogy a funkcionális elvárásokat teljesíti, egy rendszernek könnyen használhatónak kell lennie. A szoftver-ergonómia biztosítja, hogy a felhasználók minél könnyebben végezhessek el munkájukat, minél könnyebben tudjanak tájékozódni a szükséges funkciók és adatok között. Ez amellett, hogy elismertebbé, közkedveltebbé teszi a megfelelően kialakított alkalmazásokat, legtöbbször a használatuk során időt megtakarítva gazdaságosabbá teszi azok alkalmazását. Összességében a szoftver-ergonómia célja, hogy az alkalmazások jól használhatóak, lehetőleg egyszerűek, az emberre szabottak legyenek.

3 HASZNÁLHATÓSÁG

Használhatóság alatt egy minőségi tulajdonságot értünk, mely arra vonatkozik, hogy mennyi erőfeszítést igényel az adott rendszer használata. A használhatóság elég összetett tulajdonság, több komponens határozza meg:

- Tanulhatóság
- Hatékonyság
- Megjegyezhetőség
- Hibák
- Elégedettség

3.1 Tanulhatóság

A tanulhatósággal jellemezzük, hogy az adott rendszer mennyire illeszkedik az emberi agy memóriaszerkezetéhez. A megismerési folyamat az észleléssel kezdődik. Ehhez a rövid távú memóriára van szükség, mely azonban csak kevés, meghatározott (általánosan 7 ± 2) dolgot tud elraktározni, s – mint a nevéből adódik – ezeket is csak kevés ideig, mintegy 20 másodpercig. Az észlelés folyamata hierarchikus: először a nagyobb, színes, figyelemfelkeltő elemeket vesszük észre, majd lépésekben a többit. Ezért mindenképpen fontos, hogy a ténylegesen fontos elemek megfelelően ki legyenek emelve, megragadják a felhasználó figyelmét. Az észlelést követi az értelmezés, amikor is jelentés tulajdonítódik a látottakhoz. Kialakul az ún. célképzet: létrejön egy kognitív modell arra vonatkozóan, hogy mit akarunk csinálni. A modell a későbbiek során változhat, s természetesen gyakran változik is (a rendelkezésre álló információk számának növekedése következtében), vagy akár több modell is létrejöhet. Meg kell azonban jegyezni, hogy jellemzően,

ha egyszer sikerült valamilyen módon elérni célunkat, legközelebb is úgy fogunk cselekedni – habár lehetséges, hogy nem az volt a legoptimálisabb megoldás.

3.2 Hatékonyság

A rendszerünk hatékonyságát leginkább a következő kérdésekre adott válaszokkal mérhetjük:

- Használható-e arra, amire készült?
- Mennyire eredményesen végez el egy adott feladatot a felhasználó?
- Milyen gyorsan tudja elvégezni munkáját a kezelő?

Vegyük figyelembe, hogy pl. egy form kitöltése során milyen hibák ronthatják a hatékonyságot? Sok esetben – leginkább nagy mennyiségű szöveges adat bevitelekor – az egér használata jelentősen lassíthatja a folyamatot. Ilyenkor jelenthet előnyt, ha a sok kattintgatás helyett tabokkal is végig tudunk menni a felületen – minden akadály nélkül (természetesen figyelni kell a megfelelő sorrendre). A jó felület már az adatbevitel folyamán jelzi a hibákat, megakadályozva ezzel a továbbmenetel előtti, felugró hibaüzeneteket. Gyakori hibaforrást jelent azonban a validációt végző aszinkron hívások miatt elveszett fókusz (pl. hibás szöveget tartalmazó mezőben való visszatörlés esetén a fókusz átugrik egy a már megfelelő adatok miatt engedélyezésre került gombra, megzavarva ezzel a közben tovább gépelő felhasználót).

3.3 Megjegyezhetőség

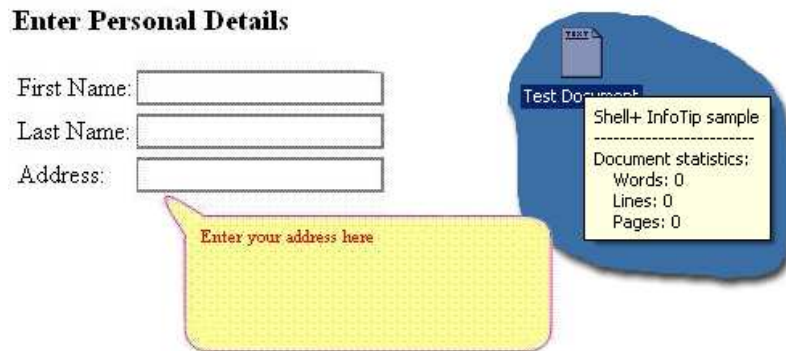
Ahhoz, hogy egy alkalmazás jól használható legyen, fontos, hogy kövessük a szabványos megoldásokat, tartsuk magunkat a már jól bevált konvenciókhoz. A felhasználó ugyanis akaratlanul is megjegyzi, hogyan jutott el egy bizonyos szinten elhelyezkedő menüponthoz, milyen lépések kellettek az adott feladat elvégzéséhez. Az így megszerzett tudást pedig önkéntelenül is megpróbálja alkalmazni a többi alkalmazás kezelése esetén is. Gondoljunk csak bele, milyen problémát okozna, ha minden alkalmazás másképp működne, az azonos jelzések más-más jelentéssel bírnának. Amennyiben tartjuk magunkat ezen szabályokhoz (pl. a menü szinte mindig a bal felső sarokban van, a szokásos menüpontok általában egyazon sorrendben helyezkednek el), nem csak azt tudjuk biztosítani, hogy az alkalmazás hosszabb használat nélküli idő eltöltése után is újra könnyen használható maradjon, de a már korábban megszerzett tudását felhasználhassa a kezelő, valamint az itt szerzett tapasztalatok is alkalmazhatóak legyenek más rendszerek esetében is.

3.4 Hibák

Hibák alatt értjük, hogy:

- mennyi hibát ejt a felhasználó használat közben
- mennyire súlyosak ezek a hibák
- az ejtett hibák könnyen korrigálhatóak-e

Egy jól megtervezett alkalmazásban – a megfelelő jelzésekkel – nagyon sok lehetséges hibát ki lehet szűrni. A validáló eszközök mellett alkalmazhatunk tooltip-eket vagy infotip-eket (1. ábra) a felhasználó segítésére. Ezek megkímélik a felhasználót a hosszú dokumentációk böngészésétől, s szükség esetén azonnal elláthatják őt a szükséges információkkal, vagy épp tanácsokkal. Nagy szerepe van még a megfelelő elnevezések használatának is, hiszen nincs is annál nehezebb, mint egy olyan mező kitöltése, melynek neve nem teszi egyértelművé az általa várt tartalmat.



1. ábra: Tooltip és infotip

3.5 Elégedettség

Az elégedettség mutatja, hogy – a felhasználó szemszögéből nézve – mennyire kényelmes használni a rendszert, mennyire jellemző a használat közbeni fáradtság, feszültség. Mutatja, hogy összességében milyen a felhasználói élmény. Egy alkalmazás esetén elégedett a kezelő, ha egyszerűen, az általa megszokott vagy épp logikusnak vélt módon el tudta végezni az általa kívánt feladatot, míg egy weboldal tekintetében elégedett lehet a felhasználó, ha könnyen és gyorsan megtalálta, amit keresett. Éppen ezért van nagy szerepe a rendszer tervezése és fejlesztése alatt is a felhasználók bevonásának. Egyáltalán nem valószínű ugyanis, hogy egy átlagos felhasználó ugyanúgy fog gondolkodni, mint a rendszer tervezését és megvalósítását végző szakemberek.

4 FELHASZNÁLÓI FELÜLETEK FONTOS TERVEZÉSI SZABÁLYAI

- A rendszernek mindig tájékoztatnia kell a felhasználót arról, hogy éppen mi történik. Fontos, hogy ha a felhasználó valamit végrehajt, arról legyen érthető visszajelzés. Például egy egyszerű keresésnél, a felhasználó beírja a kulcsszavakat, és rányom a keresésre. A rendszer elindítja a keresést, mivel azonban sok az adat, egy-két másodpercig eltarthat a lekérdezés. Ha nem történik semmi szemmel látható a képernyőn, akkor szinte biztos, hogy még 3-4-szer rá fog kattintani a keresés gombra. Ezt elkerülhetjük valamilyen visszajelzéssel (pl. a képernyő leszürkítése, pörgő-forgó ikon), illetve emellett érdemes a gombokat is letiltanunk erre az időre.
- A rendszernek a felhasználó nyelvét kell beszélnie. Azokat a szavakat, terminológiákat kell használnia, amit kezelője ismer, és használ.
- Biztosítani kell a reverzibilitást: a felhasználónak lehetőséget kell biztosítani, hogy megszakíthassa, visszavonhassa a már elkezdett folyamatot.
- A rendszernek konzisztensnek kell lennie: az azonos dolgokat azonos, a hasonló dolgokat hasonló módon és helyen kell jelölni. Ez biztosítja a könnyebb megjegyezhetőséget.
- Meg kell védeni a felhasználót attól, hogy végzetes hibát kövessen el. A nem triviális, destruktív műveleteket ki kell védeni, illetve megerősítést kell kérni!
- Minél kevesebb információra kell emlékeznie a felhasználónak, annál jobb. Ne kényszerítsük a felhasználót arra, hogy egyik oldalról a másikra meg kelljen jegyeznie dolgokat.
- Ne csak egyfajta felhasználónak készítsük az alkalmazást! Biztosítsunk gyorsításokat pl. gyors billentyűkkel a haladó felhasználóknak, tegyük lehetővé bizonyos lépések átugrását, illetve nyújtsunk támogatást a kezdőknek!
- A hibaüzeneteknek mindig érthető, magyarázó hangnemben kell megjeleníteniük. Ha van lehetőség rá, megoldási javaslatot is tehetünk.
- A legjobb, ha használható a rendszer felhasználó kézikönyv nélkül is, de mindenképpen szükséges dokumentációt, segédletet is biztosítani.
- Ügyeljünk az esztétikusságra! A felhasználói felület (UI – UserInterface) minőségét befolyásolja a felhasznált színek száma és fajtája, a grafikai objektumok élessége, felbontása, valamint az olvashatóság. A színes felület vonzóbb, gyorsabban áttekinthető, könnyebben megjegyezhető, azonban ügyeljünk arra, hogy a szín csak hozzáadott érték legyen, színtévesztők számára is egyértelmű legyen! Először inkább egyszínűre tervezzük meg a felületet, hiszen a rossz felületet színekkel nem lehet helyrehozni. Kiemelni csak kevés elemet érdemes. Legfeljebb 7 színt használjunk egy oldalon, mivel a túl sok szín között nehéz eligazodni, a túl harsány, vibráló színek pedig fárasztják a szemet.

- A felületi elemek elhelyezése legyen a szemnek kellemes, használjunk azonos sorközöket, az egyazon célt szolgáló elemek ugyanúgy nézzenek ki, méretükben, színükben ne térjenek el! A hasonló feladatokat ellátó elemeket csoportosítsuk, de csak egy domináns részfelület legyen!

5 A MOZGÁS

A leghíresebb, a felhasználók egerrel végzett mozgásáról végzett tanulmány Paul Fitts nevéhez fűződik, aki a felületi elemek elérhetőségét vizsgálta. Megállapította, hogy egy elem elérhetősége függ a méretétől, és a kurzor aktuális pozíciójától való távolságától. Ezek alapján az 5 legkönnyebben elérhető pont a képernyőn: az eger aktuális pozíciója (távolság 0), illetve a felület 4 sarka, ugyanis ezek mérete végtelennek tekinthető. Gondoljunk csak bele, ha a sarokba szeretnénk kattintani (pl. a Start menü eléréséhez), nem szükséges pozícionálnunk, akármeddig húzhatjuk az egeret, ha elérte a képernyő szélét, onnantól akármikor kattinthatunk, el fogjuk találni. Az eger aktuális pozícióját használják ki a helyzet érzékeny menük.

6 WEB-ES FEJLESZTÉS

Napjainkban egyre nagyobb szerephez jut az Internet. Szinte minden – korábban különálló alkalmazás – megtalálható webes formában. Éppen ezért válik mind fontosabbá használóinak megismerése, szokásaiknak tanulmányozása, megfigyelése. Itt is érvényes az, hogy próbáljuk minél könnyebben használhatóvá tenni a webes alkalmazásokat, könnyebben áttekinthetővé tenni a weboldalakokat, segíteni az érdeklődőt, hogy könnyen és gyorsan megtalálja a keresett információt. Kiemelt szerepe van az ergonómiának a webes fejlesztés esetén, ugyanis sok esetben az elérhető tartalmak ingyenesek, viszont gyakran más oldalán is elérhetőek. Így pedig csak azzal lehet biztosítani, hogy a felhasználók az adott oldalt válasszák, ha mindent megteszünk, hogy megkönnyítsük számukra a használatot. Ehhez az internetezők viselkedésének analizálásával további szabályokat alkothatunk a webes fejlesztés tekintetében.

6.1 Nem szeretünk gondolkodni

Az internetet sok mindenre használjuk, de leginkább információszerzésre. Az információhoz gyorsan, egy-két kattintással szeretnénk hozzájutni. Ha egy cég weblapjáról nem derül ki rögtön mivel foglalkozik, már tovább is állunk. Ha egy oldal túlzsúfolt, nem logikus, másikat keresünk. Alapjában véve nem szeretünk klikkelni, így fontos információkat ne tegyünk a nyitóoldaltól 3 kattintásnyinál nagyobb távolságra. Bár abban az esetben nem számít, hányszor kell kattintani, ha mindet gondolkodás nélkül megtehetjük, és pontosan tudjuk hová megyünk. Ezért elengedhetetlen az egyértelmű, logikus navigációs rendszer. Mindig tudnia kell a felhasználónak, hol van, honnan jött, és hova mehet innen. Ha eltéved az oldalon, valószínűleg bezárja azt, nem fog visszamenni a kiindulási ponthoz.

6.2 Ne bonyolítsuk túl a dolgokat

Minél egyszerűbb a felület, annál könnyebben használható. A felhasználók szeretik a megszokott dolgokat. Ha a dolgok mindig ugyanúgy működnek, nem kell aggódniuk, hogy mi fog a következő pillanatban történni, mert korábbi tapasztalataikra alapozhatnak.

6.3 Nem olvasunk, pásztázunk!

Hatalmas hiba azt gondolni, hogy minden szöveg, melyet egy oldalon elhelyezünk hasznos, elolvasásra kerül. Szemmozgás követés (*eye-tracking*) vizsgálatok ugyanis kimutatták, hogy az emberek F alakban pásztázzák végig az oldalakat. Először egy vízszintes vonalon pásztáznak végig az oldal tetején, majd egy kicsit lejjebb, még egy vízszintes vonalat tesznek, aztán a bal oldalon haladnak az oldal alja felé. Ez természetesen függ az oldal tartalmától is: másképp pásztázunk egy kereső oldalon, mint egy webshopban egy termék információs oldalán. Az alábbi képeket (2. ábra) egy ilyen eye-tracking vizsgálat során készítették.



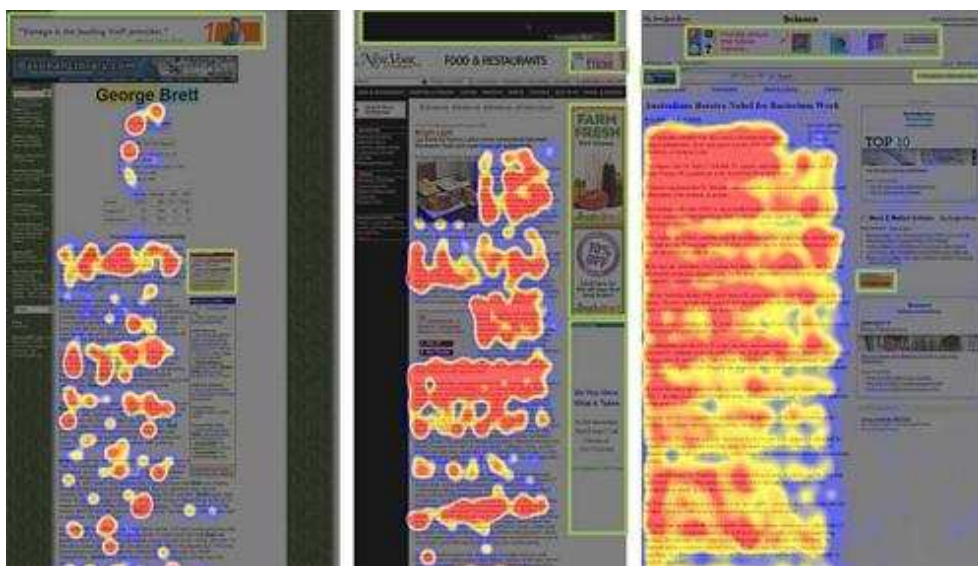
2. ábra -Pásztázó mozgás – Eye-tracking diagram

Az első kép egy cég weboldala, a második egy termék oldala, a harmadik pedig egy keresés megtekintése során készült. A vörös részek a legolvasottabbak, a kék részeknek kevésbé szenteltek figyelmet, míg a szürke területeket meg sem tekintették. Ezekből a vizsgálatokból világosan látszik, hogy az emberek nem olvassák el az egyébként gondosan megszerkesztett tartalmat, csak átfutnak rajta, kulcsszavakat keresve. Ezt figyelembe véve, az első két bekezdésben kell elhelyezni a legfontosabb tartalmat, így meg van az esély rá, hogy az első sorok után is tovább olvasnak. Ezekben is inkább az első két szó legyen figyelemfelkeltő kulcsszó, hogy azokat pásztázás közben könnyen megtalálják! Ez alapján nem tévedünk nagyot, ha az alábbi képlettel próbáljuk leírni, mennyi a valószínűsége az adott tartalom elolvasásának:

$$\text{Elolvasás valószínűsége} \approx \frac{1}{\text{Szöveg mérete}}$$

Ez a szelektív figyelem csak fokozódott a bannerek elterjedésével, ugyanis kialakult az internetezőkben az ún. *banner-vakság*. **A Hiba! A hivatkozási forrás nem található.** jól jelzi, hogy

a hirdetésnek látszó képeket a felhasználók teljes mértékben figyelmen kívül hagyják, beleértve a bannernek látszó design elemeket is. Ezért érdemes elkerülni ezek használatát!



3. ábra: Banner-vakság

7 ÖSSZEGZÉS

Habár a szoftver-ergonómia egy viszonylag új tudományág, fejlődése kétségkívül dinamikus, jelentősége pedig egyre inkább megkérdőjelezhetetlen. Igazán jó felületet, könnyen és hatékonyan kezelhető alkalmazást, az átlagos internetező számára könnyen áttekinthető, értelmezhető weblapot készíteni ugyanis nem könnyű.

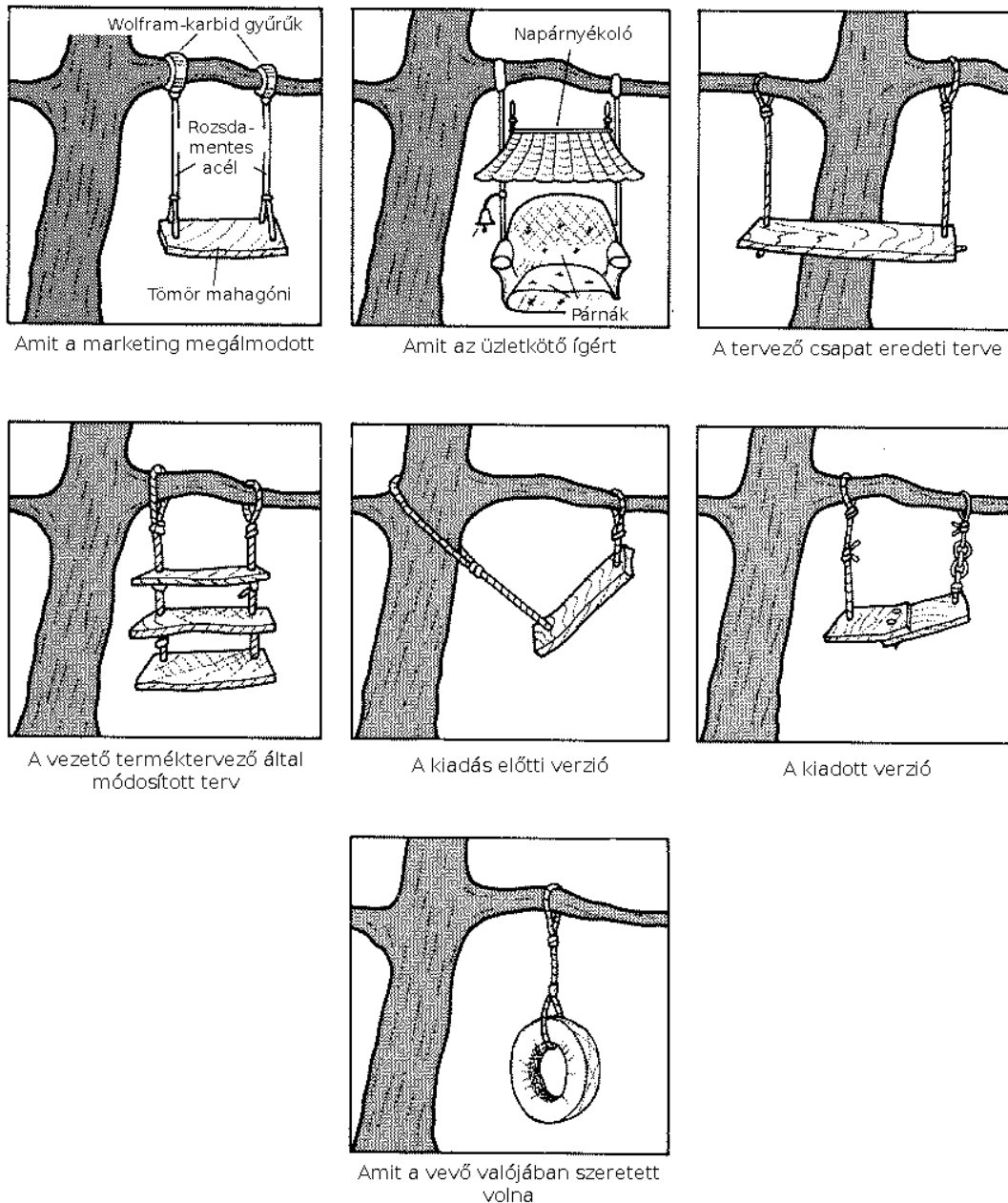
A fejlesztők átlagos felhasználóknak gondolják magukat, pedig nem azok. Így pedig az általuk fontosnak vélt funkciók más (a tényleges felhasználó) számára teljesen feleslegesnek bizonyulhatnak, s gyakran bizonyulnak is. Szem előtt tartva azonban a fentebb leírt szabályokat, figyelembe véve a felhasználók általános viselkedését, azok egyedi igényeit, sikerülhet jól használható felületet készíteni.

Nem csoda hát, hogy egyre nagyobb jelentőséget tulajdonítanak ennek a tudománynak, hiszen segítségével nem csak népszerűbbé válhat alkalmazásunk, vagy épp látogatottabbá weblapunk, de hatékonyabbá tehetjük azok használatát, üzemeltetésüket pedig gazdaságosabbá.

Természetesen ezen írás – már csak terjedelme miatt is – nem tudja bemutatni a szoftver-ergonómia tudományának minden részletét, megpróbálja azonban kiemelni ezen átfogó terület legfontosabb részeit. Dióhéjban elmondható, hogy nagyságrendekkel jobb alkalmazásokat készíthetünk, ha legalább az alábbi néhány egyszerű alapszabályt megpróbáljuk betartani:

- Törekedjünk az egyszerűsége, esztétikusságra!
- Figyeljünk a konzisztenciára!
- Maradjunk következetesek!
- Mindig tájékoztassuk a felhasználót!

És végezetül talán mindezek mellett a legfontosabb, melyet sose felejtünk el: mindenképp azt próbáljuk meg létrehozni, amit a megrendelő kért tőlünk!



4. ábra: A megrendeléstől az átadásig

Az esettanulmány elkészítéséhez munkájával hozzájárult Kovács Judit a Neuron Szoftver Kft. alkalmazottja.

8 IRODALOMJEGYZÉK

- <http://www.useit.com/alertbox/>
- <http://www.useit.com/alertbox/>
- <http://www.useit.com/alertbox/>
- Steve Krug - Don't make me think! A CommonSenseApproachto Web Usability, SecondEdition