

Egyetemi doktori (PhD) értekezés tézisei

Intelligens Webrendszerek

Nagy Zsolt

Témavezető: Dr. Boda István



DEBRECENI EGYETEM
Informatikai Tudományok Doktori Iskola

Debrecen, 2014

Tartalomjegyzék

1 A doktori értekezés előzményei és célkitűzései.....	1
Bevezetés.....	1
Motiváció és irodalmi áttekintés.....	2
Az Intelligens Web.....	5
Webfejlesztés.....	8
2 Az értekezés új tudományos eredményei.....	11
1. tézis - Adaptív fejlesztési módszer.....	12
2. tézis - Integrált tervezési minta.....	13
3. tézis - Modell a webrendszerek gyorsabbá tételére.....	15
4. tézis - Implicit adatgyűjtés, egy új modell.....	16
5. tézis - Reszponzív és tartalomfüggő megjelenítés.....	17
6. tézis - Adatvédelem szakértői szemmel.....	19
3 Previous research and the motivation for the thesis.....	20
Introduction.....	20
Motivation and literature overview.....	21
Intelligent Web.....	24
Web engineering.....	26
4 The new scientific results of the thesis.....	29
Thesis 1 – Adaptive Development Method.....	30
Thesis 2 – Integrated Design Pattern.....	31
Thesis 3 – Model for faster web system performance.....	33
Thesis 4 – Implicit data collection, a new model.....	34
Thesis 5 – Responsive and content-aware presentation.....	35
Thesis 6 – Privacy from forensic point of view.....	37
5 Irodalomjegyzék / References.....	39
6 Publikációk / Publications.....	48

1 A DOKTORI ÉRTEKEZÉS ELŐZMÉNYEI ÉS CÉLKITÚZÉSEI

Bevezetés

Disszertációmban összefoglaltam az elmúlt 5 év kutatómunkáját, mely során igyekeztem minél részletesebben elemezni az intelligens webrendszerekkel kapcsolatban eddig elért nemzetközi eredményeket, felkutatni, felismerni azon területeket, ahol mind tudományos, mind szakmai értelemben érdekes kihívások jelentkeznek, hasznos eredmények születhetnek.

Kutatómunkám során átdolgoztam a webrendszerek fejlesztési folyamatát, majd egy, a mai kor igényeinek megfelelő, MVC alapokon nyugvó tervezési mintát kínáltam, mely jól alkalmazható kliens-szerver oldali keretrendszerek integrálására is. Az intelligens webrendszerek kimeneti adatainak megfelelő megjelenítése kulcsfontosságú a felhasználói élmény és – nem utolsósorban – a piaci profit szempontjából, így egyrészt egy új modellt javasoltam meglévő és új webportálok gyorsabbá tételére, másrészt kidolgoztam egy új, AJAX alapú implicit adatgyűjtő módszert, mellyel igen hatékonyra tehető az információgyűjtés. A felhasználói interfész rezponzivitása, intelligenciája szintén kiemelt jelentőséggel bír, így javasoltam egy univerzális, rezponzív, tartalomfüggő rendszermodellt, mely tetszőleges témájú webrendszerhez használható, végezetül felkutatam, rendszereztem és bemutattam annak veszélyét, hogy a webrendszerek hol, milyen adatokat gyűjtenek, tárolnak rólunk, felhasználókról.

Értekezésem nem titkolt célja, hogy közelebb hozza egymáshoz az akadémiai kutatásokat az iparban, üzleti életben alkalmazott technológiákkal. Sokszor éri vád a tudományos élet szereplőit, hogy kutatási eredményeik zöme pusztán elméleti jelentőséggel bír, távol áll a valóságtól, nehezen vagy csak évek múlva hasznosítható az iparban.

Különösen érzékeny terület ebből a szempontból az informatika, ahol rendkívül gyorsan változik, fejlődik mind a hardver, mind a szoftvertechnológia.

Az elmúlt évek során abban a szerencsés helyzetben voltam, hogy főiskolai oktatói és aktív szoftver- és webfejlesztői munkáim éppúgy hozzájárultak egy-egy témakör részletesebb elemzéséhez, megvilágításához, mint az igazságügyi szakértői vagy a CCNA instruktori feladataim során szerzett tapasztalatok.

Igyekeztem az üzleti életből vett legfrissebb, aktuális problémákat górcső alá venni és azokra olyan megoldást kínálni, mely felhasználja és alkalmazza a tudományos eredményeket; ugyanakkor az elméleti eredmények mellett gyakorlati, azonnal használható és bevethető módszereket is ajánl.

Összességében tehát a kérdés, amire a választ kerestem: Hogyan tehetjük jobbá a mai intelligens webrendszereket?

Motiváció és irodalmi áttekintés

2012-ben, útban egy nemzetközi konferenciára, a repülőgépen kezembe akadt a Traveller magazin aktuális száma, ahol Andrew Hankinson, "Holiday 2.0: are we changing the way we travel forever?" című írásában 4 oldalon keresztül boncolgatja az online foglalási rendszerek és a közösségi oldalak turizmusban betöltött kiemelkedő szerepét. Sorra veszi azokat a webportálokat és szolgáltatásokat, melyek segítségével a komplett nyaralását le tudta szervezni, utazással, szállással, étkezéssel, fakultatív programokkal, baráti találkozókkal.

Elemzi a jelenséget mind újságírói, mind befektetői szemmel, végül több következtetést is levon, mellyel saját tapasztalataink alapján mélyen egyet kell értenünk: minden piaci szereplő számára elkerülhetetlen és a talpon maradás feltétele az Internet adta online (e-) kereskedelmi lehetőségek kiaknázása. Mit sem ér azonban a látványos webportál, a gondos

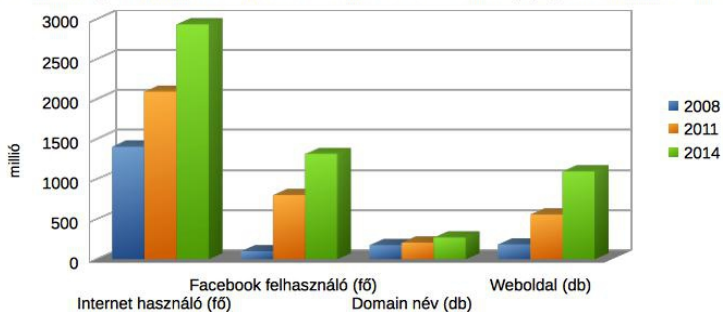
marketingmunkával megfogalmazott szállásajánlat, ha a látogató, az utazni vágyó nem bíz az adott weboldalon. A bizalom a mai Internetes életünk egyik legértékesebb pénzneme, statisztikák igazolják [19] [20], hogy a közösségi portálok, a közösség, és azon belül a közösség véleményét leginkább befolyásoló személyek (véleményformálók, vagy a Barabási-paradigma szóhasználatában: hálózati „központok”, „döntésvezetők”, „befolyásos felhasználók”, „befolyásolók”, ld. Barabási [21] 2013:144-136 és 280) véleménye meghatározó szerepet játszik üzleti döntéseinkben, az utazni vágyók jelentős része bíz inkább utastársaik közösségi oldalakon közzétett véleményében, mint például egy utazási iroda által leírtakban.

Egy webportállal, egy informatikai rendszerrel szemben egy másik fontos bizalmi kérdés is felmerül: ez pedig nem más, mint a személyes és pénzügyi adataink védelme. Ebben az esetben már nem egy helyes vagy helytelen üzleti döntés a tét, hanem - olykor – a teljes egzisztenciánk. Így, amikor naponta adatlopásokról, megfigyelésekről, lehallgatásokról hallunk, ne lepődjünk meg azon, ha a felhasználók rendkívül bizalmatlanok. Ez a fajta bizalmatlanság jócskán megnehezíti az intelligens webrendszerek készítőinek a dolgát, és kiemelt feladatot ró a marketing- és informatikai szakemberek számára, hogy megfelelő bizalmat ébresszenek, tudatosítsanak egy-egy márkában, szolgáltatásban, webportálban.

Valóban megváltoztatja életünket és vásárlási szokásainkat az Internet? Kétségtelen. Az Internetet használók és azon információt kereső felhasználók száma is folyamatosan növekszik. Míg 2008-ban ez a szám 1,4 milliárd volt, 2014-ben már 2,9 milliárd, azaz a Föld lakosságának több mint 40%-a használja rendszeresen az Internet adta lehetőségeket [1. ábra]. Ez annyit tesz, hogy minden egyes másodpercben 7 új, információra éhes felhasználó jelenik meg az Internet világában.

Az Internet használatának alakulása 2008-2014 között

Forrás: www.verisigninc.com, www.internetworldstats.com, www.internetlivestats.com, royal.pingdom.com, www.statista.com



1. ábra: Az Internet használatának alakulása

Kutatásom úgy kezdeti, mint jelenlegi fő célkitűzése a mai web alapú rendszerek hatékonyabbá tétele, mind normál számítógépes, mind mobil platformon. A hatékony webrendszer intelligens; személyre (nem, életkor, foglalkozás, nemzetiség, érdeklődési kör, földrajzi, nyelvi preferenciák) és eszközre (számítógép, telefon, tablet vagy épp hűtőszekrény kijelző) szabott tartalmat szolgáltat a felhasználó számára. Teszi mindezt a lehető leggyorsabban, hisz a kutatások igazolják: mindössze 3 másodperc várakozás és már a látogatók 57%-a ott is hagyta a webes tartalmat. További statisztikák támasztják alá, hogy minden egyes másodperc késedelem dollármilliókat vesz ki a nagyforgalmú webportál tulajdonosok zsebéből [22].

Ennek fényében érthető, hogy miért van kiemelt jelentősége az üzleti világban a hatékony, gyors és a felhasználónak is tetsző webrendszereknek.

Korunk trendjeinek megfelelően egyre inkább az tapasztalható, hogy az Internet lesz éppúgy az elsődleges információ és hírforrás, mint a szolgáltatások és kereskedelmi tevékenységek színhelye. A mai kor embere Internetes oldalakon rendel terméket, szolgáltatást, Interneten olvassa el a napi híreket, sőt a Web 2.0 megjelenése óta blogot, fórumokat,

élménybeszámolókat ír, közösségi portálokon keresztül tájékozik barátai, rokonai felől.

A Web 2.0 fogalom először [1] cikkében jelent meg, ismertséget azonban Tim O'Reilly és Dale Dougherty szerzett a kifejezésnek, a 2005-ben szervezett Web 2.0 konferencián. Mára már a fogalom jóval több, mint marketing kifejezés, valójában új technológiák összefoglaló neve [2].

Az új technológiák új kihívások elé állítják a szoftverfejlesztőket is. A Web 2.0 korában a webfejlesztés már nem csak arról szól, hogy információkat jelenítsünk meg weboldalakon, mint inkább arról, hogy olyan webalkalmazásokat hozzunk létre, melyek lehetővé teszik a hatékony információmegosztást, biztosítják a gazdag felhasználói élményt, valamint kihasználják a közösség erejében rejlő kollektív intelligenciát.

A fenti felsorolás mindhárom pillére (információ megosztás, felhasználói élmény, intelligencia) fontos, mindegyik területhez kapcsolódóan végeztem kutatásokat, értekezésemben részletesen be is kívánom mutatni az ezzel kapcsolatos eredményeket.

Az Intelligens Web

Bár a Web igen gazdag információforrás, az adatok összegyűjtése, rendszerezése egyre komolyabb feladatot ró a felhasználókat kiszolgáló rendszerek számára. Épp ezért a kutatások és fejlesztések eredményeképpen újabb és újabb technológiák jelennek meg, melyek arra hivatottak, hogy megfelelő tartalommal szolgálják ki az egyes weboldalak látogatóit. Ezen legújabb technológiák összefoglaló neve az intelligens web.

A web intelligencia (WI) kifejezés 2000-ben [5] debütált, *Ning Zhong és társai* szerint az intelligens web a mesterséges intelligencia, a tudásreprezentáció, az adatbányászat, az intelligens ágensek, az intelligens közösségi hálók összefoglaló neve, illetve az ezen területeken szerzett ismeretek, eredmények új közegben történő alkalmazása.

A WI ezen túlmenően új problémák és kihívások megoldására sarkallja az információ technológia és a mesterséges intelligencia szakembereit. A

WI technológiák forradalmasítják az információgyűjtés, tárolás, feldolgozás, megjelenítés és megosztás eddig ismert technikáit, módszereit.

2000 után, szakmai körökben igen népszerű lett az intelligens webbel foglalkozni, a népszerűség oka nem véletlen, a WI, azon belül is a web megszemélyesítés egyike azon kutatási területnek, amely remek eszköztárát és gazdasági hasznot biztosíthat az e-business iparágnak. A technológia olyan képességei, mint a felhasználók vásárlási vagy böngészési szokásainak folyamatos figyelése, rögzítése, elemzése, minden eddiginél hatékonyabb eszközt ad a kereskedők kezébe ahhoz, hogy a lehető legpontosabban célozza, személyre szabja az értékesítendő termékét, szolgáltatását.

A felhasználói élmény növelésének egyik legjobb módszere a web megszemélyesítés, hisz segítségével tudjuk elérni azt, hogy minden egyes látogató ugyanazon weblap saját, személyre szabott változatát kapja. A technológiának köszönhetően a webszerverek a weboldalt akár valós időben képesek a vásárló igényeire igazítani, ezáltal biztosítva azt a különleges, egyedülálló felhasználói élményt, mely során a látogató úgy érzi, a weboldal szinte olvas a gondolatában.

A megszemélyesítés témakörével számos kutató foglalkozott már, így A.R Simon és társa [8] a célzott marketing szempontjából, Ning Zhong és társai a wisdom web szemszögéből [9], Su Ho Ha a vásárlói döntésekre gyakorolt hatását vizsgálta [10], míg Wang és Kobsa [11] a személyes adatok védelmét figyelembe véve dolgozott ki megfelelő módszert rá.

Problémák

Az intelligens web rendszerekkel kapcsolatos kutatások azonban szinte kizárólag arra irányulnak, hogy a módszer magját képező algoritmusokat minél jobban optimalizálják, ugyanakkor igen kevés szó esik arról, hogy a feldolgozás alapját képező adatokat milyen módszerrel, technológiákkal érdemes összegyűjteni, illetve a végeredményt, a kimeneti adatokat milyen formában érdemes tárolni a felhasználó számára.

Mind az input, mind az output kiemelt jelentőséggel bír, hisz bemenő adatok nélkül a legtokéletesebb algoritmus sem ér semmit, mint ahogy egy rossz, lassú webportál mögött is dolgozhat bármilyen hatékony mesterséges intelligencia, ha a potenciális vásárlók, látogatók inkább menekülnek az oldalról, mintsem hogy böngésszék azt.

Azt gondolhatnánk, hogy mi sem egyszerűbb annál, mint adatokat gyűjteni a felhasználókról, hisz az Internet épp erről, a milliárdnyi adatról szól. Az intelligens rendszerek szempontjából viszont a felhasználókkal és azok viselkedésével kapcsolatos adatok bírnak kiemelt jelentőséggel, melyeket kétféle módon szerezhetünk be: vagy megkérjük a felhasználót, hogy adja meg az általunk kért adatokat (űrlapok, kérdőívek, termékértékelések formájában például) – ez az úgynevezett explicit adatgyűjtés, vagy az intelligens rendszerünk a felhasználó tudta, zavarása nélkül, a háttérben gyűjti az adatokat (böngészési útvonal, kattintások, látogatott weblapok, vásárlási előzmények, IP cím, nyelvi beállítások stb.) - ezt nevezzük implicit adatgyűjtésnek.

A két technológia kombinálható, ám mindkettővel akadnak problémák: az explicit adatgyűjtés nehézkes, a felhasználók egyrészt bizalmatlanok, nem szívesen adnak meg adatokat, másrészt az időt is sajnálják arra, hogy a kedvünkért űrlapokat töltsenek ki.

Az implicit adatgyűjtés az előző problémákat kiküszöböli, továbbá nagy előnye, hogy segítségével hatalmas mennyiségű adatot áll módunkban összegyűjteni, ám a jellegéből adódóan felvet egy igen fontos személyiségi jogi kérdést: gyűjthetünk e a felhasználóról a tudta, beleegyezése nélkül bármilyen adatot? Egyáltalán, tisztában vagyunk e azzal, hogy abban a pillanatban, amikor megnyitunk egy weboldalt a számítógépünk mögött meghúzódó informatikai rendszer milyen adatokat, információkat gyűjtött és tárolt rólunk?

Webfejlesztés

Egy web-alapú rendszer fejlesztése jóval több feladatot ró a fejlesztőkre, mint egy hagyományos szoftverfejlesztés; a rendszer életciklusa, a fejlesztésének menete, a nyomkövetés és fenntartás, mind mind különbözik a klasszikus szoftverfejlesztéstől. Érthető hát, hogy a tradicionális fejlesztési módszertanok sok esetben a web-alapú rendszerekre nem igazak, pontosabban korrekcióra, kiegészítésre szorulnak. Remekül fogalmazta meg Powell 2000-ben kiadott könyvében a lényegét:

“A webfejlesztés ötvözi az újságkiadást a szoftverfejlesztéssel, a marketinget a számítástechnikával, a belső kommunikációt, a külső kapcsolatokkal, a művészetet a technológiával” [23].

Különböző szakirodalom különböző, alapjaiban mégis hasonló módon definiálja a webfejlesztés (web engineering) fogalmát. Ezek alapján összefoglalva elmondhatjuk, hogy a webfejlesztés nem más, mint módszer a webalkalmazás-fejlesztéssel kapcsolatos tudás fejlesztésére és szervezésére. Leginkább Murugesan és társai által megfogalmazottakkal értek egyet, mely szerint a webfejlesztés a tudományos, mérnöki és menedzsment tudományokkal kapcsolatos ismeretek szisztematikus alkalmazása a web-alapú rendszerek sikeres fejlesztése, üzembe helyezése és fenntartása érdekében [24].

A web alapú rendszerek jellemzőit többen, többször megfogalmazták már [25,26,27,28], ezekből is kitűnik, hogy a web-alapú rendszerek sok szempontból különböznek a klasszikus szoftverrendszerektől, igaz ez a fejlesztés és a tervezés folyamatára is.

A szoftverfejlesztési folyamat bonyolult és összetett, az egyes részfeladatokat, illetve az ezek közötti kapcsolatokat modellezni szükséges annak érdekében, hogy egyrészt áttekinthető, másrészt moduláris felépítésű legyen.

A legismertebb és legelterjedtebb a vízesés (waterfall) modell, mely a szakirodalomban először Winston W. Royce 1970-es cikkében jelent meg

[29], bár érdekesség, hogy maga a *waterfall* kifejezés az írásban egyetlen egyszer sem szerepel. Sőt épp az ilyen jellegű fejlesztési folyamat hibáit, hátrányait ismerteti.

A modell mégis jól alkalmazható abban az esetben, ha a rendszerrel szemben támasztott követelményeket már a fejlesztés legelején tudjuk. Ez természetesen alapos felmérő és kutatómunkát igényel, nem beszélve arról, hogy a megrendelőnek is igen felkészültnek kell lennie az elvárt funkcionalitást illetően, hisz a menet közbeni változtatások rendkívül költségesek olyannyira, hogy bizonyos lépcsőfokok után már gazdaságosabb a legelejéről újrakezdeni a fejlesztést.

Sajnos a való életben a megrendelő nem tudja teljes pontossággal definiálni a projekt elején, mit is akar; a követelmények menet közben változnak, finomodnak. Épp ezért a vízésés modellt az üzleti webfejlesztésben egyre kevésbé használják, ugyanis ezekre a változásokra a vízésés modell nincs felkészülve; ha valamely fejlesztési fázis lezárult, szinte lehetetlen azon változtatásokat eszközölni.

Egy olyan módszerre van szükség, mely rugalmas, lehetővé teszi azt, hogy menet közben rendszeresen konzultáljunk a megrendelővel, és ha módosításra van szükség, azt a legkisebb költséggel tegye lehetővé.

Már Royce cikkében is megjelenik az igény egy új, iteratívabb, interaktívabb módszer alkalmazására, nem véletlen, hogy az évek folyamán megannyi vízésés modell alternatíva, továbbfejlesztett módszertan született.

Bíró külön cikkben [30] ismerteti a fel-fel bukkanó módszertani divathullámok karakterisztikáit, a legújabb hullám még ma is tart és gyökerei egész 1988-ig, Boehm spirál [31] és Gilb evolúciós modelljéig [32] nyúlnak vissza.

Ez az új trend az agilis szoftverfejlesztés, mely sok szempontból épp az ellenkezője a vízésés modellnek. A fejlesztők, ahogy elkészült egy kisebb modul, átadják tesztelésre a felhasználónak, aki így kipróbálhatja, pontosíthatja az igényeit, az új igényekre gyorsan és hatékonyan reagálva

módosítják azt, majd ismét egyeztetnek a megrendelővel; a folyamatos konzultáció és iteráció az agilis szoftverfejlesztés alapja. A módszer garancia arra, hogy a felhasználó elégedett, azt kapja amit szeretett volna, még ha a fejlesztés elején nem is tudta kellő alaposággal megfogalmazni igényeit.

Problémák

Pusztán az agilis módszertani divathullámot meglovagolva mondhatnánk azt, hogy a mai kor fejlesztési folyamatának egyértelműen ezt választjuk, de a módszernek vannak hátrányai. A folyamatos konzultáció remek eszköz, de időigényes, sőt, ha a megrendelő épp nem ér rá, hátráltatja a fejlesztést. Az állandó revízió és változtatás lehetősége miatt jóval több szakembernek (grafikus, rendszertervező, front-end, back-end programozó) kell rendelkezésre állnia egyidőben, szemben például a vízéses modellel. Mindezek mellett a várható költségek és befejezési határidő is nehezen becsülhető.

Ésszerű feltételezés, hogy a két rendszer előnyeit kombinálva hatékonyabb metódust kínálhatunk a webfejlesztésre. A hibrid rendszer kialakításában nagy segítséget jelentett az üzleti élet szereplőivel való konzultáció, a saját fejlesztői tapasztalat, mely alapján azonosítani tudtuk azokat a hibákat, kritikus pontokat, amelyek egy nem megfelelő módszertan alkalmazása, vagy épp bármilyen módszertan figyelmen kívül hagyása miatt jelentkezhetnek egy alkalmazásfejlesztési folyamat során.

Mindezeket túl ismét hangsúlyozni szükséges a felhasználói interfész, az arculat és a design fontosságát, így a kifejlesztett hibrid módszer már erre vonatkozóan is tartalmaz fejlesztési modellt. Az eddigi modellek elegánsan átléptek ezen a ponton, vagy épp egybemosták a rendszertervvel, holott a megrendelő és a majdani felhasználók számára - érthető módon - a kinézet és érzet, az ún. „look & feel” kiemelt jelentőséggel bír.

2 AZ ÉRTEKEZÉS ÚJ TUDOMÁNYOS EREDMÉNYEI

A disszertáció 6, egymáshoz szorosan kapcsolódó fejezetet tartalmaz, melyekben a web alapú rendszerfejlesztés teljes folyamatát végigkísírem. A fejezetekről készült rövid áttekintésekben ismertetem a kiindulási problémát, a problémára adott új kutatási eredményeimet, módszereimet.

Fiatal kutatóként még épp, hogy csak betekintést kaptam a tudományos világba, így külön öröm számomra, hogy az értekezésben megfogalmazott eredményeimet, cikkeimet külföldi kutatók idézik, használják, így kérem engedtessek meg, hogy ezeket is feltüntessem, megemlítssem ott, ahol ez releváns.

Az *első fejezetben* a webes rendszerek fejlesztési folyamatát tekintetem át, a jelenleg használt modell hiányosságait felderítve a mai kor igényeinek megfelelő megoldást, új módszert dolgoztam ki.

A *második fejezet* a teljes fejlesztési folyamatból kiemeli a rendszertervezést, részletesen ismerteti a manapság alkalmazott hagyományos és divatos tervezési mintákat, majd az eddigiekhez képest új, ugyanakkor mégis MVC alapokon nyugvó tervezési mintát dolgoztam ki, mely egyben alkalmas a kliens-szerver keretrendszerek integrálására is.

A *harmadik fejezet* ismerteti a jelenleg piacvezető webrendszerek teljesítménybeli hiányosságait, a meglévő sebességnövelő technológiákat sorra veszi, ellenőrzi, majd egy olyan új folyamatmodellt kínál, mellyel gyors webrendszerek építhetők, a meglévő webrendszerek sebessége növelhető.

A *negyedik fejezet* egy olyan érdekes, Ajax-alapú, implicit adatgyűjtő technológiát mutat be, mellyel igen hatékonyra tehető az információgyűjtés, kiszolgálja a rezponzív megjelenítés igényeit, valamint ideális adatgyűjtési lehetőséget kínál az ajánlói rendszerek számára.

Az *ötödik fejezet* a szoftverrendszer felhasználóval közvetlenül kapcsolatban lévő komponensét hivatott intelligenssé, jobbá tenni. Az itt

kínált új modellel megvalósuló fejlesztések felhasználói interfésze intelligens módon alkalmazkodik az öt megjelenítő eszközhöz, annak típusához, képernyőfelbontásához.

Végezetül az utolsó, *hatodik fejezetben* felkutatom, elemzem és bemutatom a már kész intelligens rendszerek adatvédelmi kockázatait, rávilágítok arra, milyen veszélyeknek vannak kitéve személyes adataink a Web 2.0 korában.

1. tézis - Adaptív fejlesztési módszer

Probléma

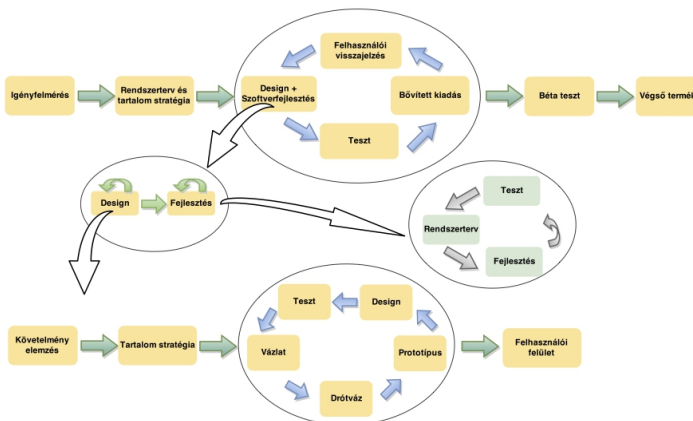
A szoftverfejlesztési folyamat bonyolult és összetett, az egyes részfeladatokat, illetve az ezek közötti kapcsolatokat modellezni szükséges annak érdekében, hogy egyrészt áttekinthető, másrészt moduláris felépítésű legyen. A legismertebb és legelterjedtebb a vízesés (waterfall) modell. A modell jól alkalmazható abban az esetben, ha a rendszerrel szemben támasztott követelményeket már a fejlesztés legelején tudjuk.

Sajnos a való életben a megrendelő nem tudja teljes pontossággal definiálni a projekt elején, mit is akar; a követelmények menet közben változnak, finomodnak. Épp ezért a vízesés modellt az üzleti webfejlesztésben egyre kevésbé használják, ugyanis ezekre a változásokra ez a modell nincs felkészülve; ha valamely fejlesztési fázis lezárult, szinte lehetetlen azon változtatásokat eszközölni.

Megoldás

Egy olyan módszerre van szükség, mely rugalmas, lehetővé teszi azt, hogy menet közben rendszeresen konzultáljunk a megrendelővel, és ha módosításra van szükség, azt a legkisebb költséggel tegye lehetővé. Ilyen módszernek ígérkezik az agilis szoftverfejlesztés, ám sok esetben az sem megfelelő. A két rendszert a saját kutatás-fejlesztési munkám eredményei alapján ötvöztem és ebből alkottam meg a mai modern webrendszerek fejlesztéséhez igazodó új folyamatmodellt, az adaptív fejlesztési módszert.

A friss kutatási eredményeket *Adaptive Design Process for Responsive Web Development* (DOI: 10.13140/2.1.3354.5601) című cikkemben is ismertettem.



2. ábra: Az adaptív rendszerfejlesztési modell

2. tézis - Integrált tervezési minta

Probléma

A mai webalkalmazások fejlesztése megfelelő tervezési minta nélkül nem lehetséges, hisz ma már egyszerre kell ellátni a gazdag kliens oldali programozási feladatokat a megszokott szerver oldali alkalmazásfejlesztéssel. Akár a kliens oldali akár a szerver oldali programozási munkáról beszélünk, a munka volumene megköveteli a tervezési minták használatát. Halmozottan igaz ez egy komplex webalkalmazásra, ahol a kliens-szerver oldali fejlesztés szükségszerűen elválaszthatatlan kapcsolatban áll egymással.

A legnépszerűbb tervezési minta mind a mai napig az MVC. A kutatás ezen fázisában arra kerestem a választ, hogy a mai modern fejlesztői

környezetekkel is ugyanúgy használható e a tervezési minta, tudok e esetleg egy új, jobb, a kliens és szerver oldali rendszereket összefogni képes architektúrát kínálni a fejlesztőknek.

Megoldás

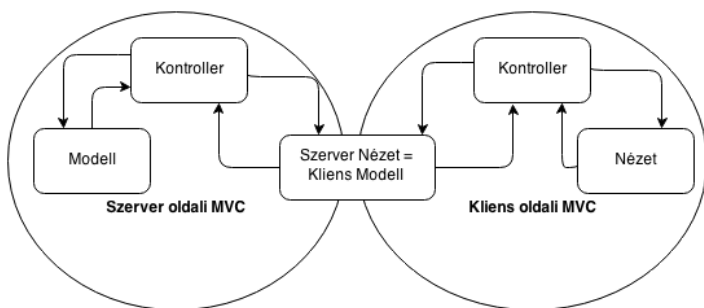
A különböző tervezési minták vizsgálata során arra a következtetésre jutottam, hogy az eredeti MVC architektúra kissé módosított, Cocoa verziója a megfelelő kiinduló állapot egy új integrált minta kidolgozására. Választásomat indokolta továbbá az a tény, hogy mind kliens, mind szerver oldalon az MVC keretrendszerek a legelterjedtebbek.

A kérdés továbbá az volt, hogyan lehet összekapcsolni a népszerű JavaScript MVC-eket a szintén igen elterjedt PHP MVC-vel úgy, hogy közben az összekapcsolt rendszerünk ugyanúgy megfeleljen a Modell-Nézet-Kontroller felépítésnek.

Szerver oldalról megközelítve egyértelműen a Nézet komponens az, ami további szegmentálásra szorul, hisz ennek bonyolultsága az, ami nehezíti a fejlesztési munkát. Lecserélve a Nézetet egy komplett kliens oldali MVC-re a rendszerünk MVC-ből M(MVC)C-é változik.

Amennyiben kliens oldalról vizsgáljuk a helyzetet, az MVC architektúra Modell komponense a sima HTML kód, a Nézet a CSS fájl (ahány CSS fájl, annyi nézet), míg a Kontroller maga a böngésző, illetve a böngésző képességeit kiterjesztő JavaScript programkód. Egy másfajta megközelítésben – különösen ha a kliens-szerver között asszinkron, például AJAX-alapú kommunikáció zajlik, a Nézet a HTML+CSS+adat kombinációjából megszületett felhasználói felület, a Kontroller szerepét JavaScript osztályok és metódusok töltik be, míg a Modell nem más, mint a webszervertől érkező adat.

Bármelyik szemléletet is tekintjük, kliens oldali megközelítésben a Modell az a komponens, melyen keresztül a rendszerünk illeszhető a szervertől oldali MVC rendszerhez. Ennek alapján az alábbi szemléletes ábra szemlélteti az új tervezési mintát.



3. ábra: Az integrált tervezési minta

A kidolgozott új mintát a SOFA2014 nemzetközi konferencián ismertettem, melyből már megjelentetésre elfogadott *Integrated Design Pattern for Intelligent Web Applications* címmel született 14 oldalas cikkem.

3. tézis - Modell a webrendszerek gyorsabbá tételére

Probléma

Ahogy a bevezetőben is ismertettem, megannyi kutatás igazolja: mit sem ér egy intelligens, remek matematikai modellre épülő webrendszer, ha a felhasználó nem győzi kivárni a megjelenítendő tartalmat. A sebesség fontosságának ékes bizonyítéka, hogy 2010-től a Google is felvette és alkalmazza a weboldalak sebességét, - mint értékelési paramétert - a PageRank rangsorolási algoritmusában. Több érdekes könyv és tudományos cikk jelent meg a weblapok gyorsabbá tételére, sőt a két legnagyobb kereső, a Google és a Yahoo is közzétett számos ajánlást és mérési módszert ezzel kapcsolatban, ám felmérések alapján a világon kevés olyan weboldal van, mely maradéktalanul megfelelne a javasolt technológiai irányelveknek. A világ, az Egyesült Államok, illetve Magyarország első 10 leglátogatottabb weboldalát megvizsgálva kiderül, hogy hazánk igen csak le van maradva ezen új javaslatok és módszerek alkalmazásában.

Megoldás

Kutatásom harmadik fázisában egy napi 4000 egyedi látogatót számláló turisztikai portálon teszteltem és alkalmaztam a saját fejlesztési tapasztalatokon alapuló, illetve a szakirodalom által javasolt módszereket. A hatékonyság-vizsgálat eredményeire építve egy olyan módszergyűjteményt alkottam meg, mely alkalmas nem csak meglévő weblapok átalakítására, hanem a tervezési fázisban is jól alkalmazható, így nagymértékben elősegítheti a gyorsabb intelligens webrendszerek születését.

Improved Speed on Intelligent Web Sites címmel 2013-ban közöltem ide vonatkozó eredményeimet, melyre **egyiptomi** kutatók is felfigyeltek, az International Journal of Advanced Computer Science and Applications folyóiratban megjelent *XML Schema-Based Minification for Communication of Security Information and Event Management (SIEM) Systems in Cloud Environments* (DOI: 0.14569/IJACSA.2014.050912) cikkükben hivatkoznak munkámra.

4. tézis - Implicit adatgyűjtés, egy új modell

Probléma

A személyre szabott tartalom megvalósításának legfőbb eszköze az ajánló rendszerek alkalmazása, használata. Azonban a jelenlegi kutatások java azzal a problémával szembesül, hogy igen nehéz megfelelő mennyiségű és minőségű valós adatot összegyűjteni, ezért a legtöbben vagy minta-adatbázisokon dolgoznak vagy – jóval kevesebb számban - olyan szerencsés helyzetben vannak, hogy valódi rendszereken tesztelhetik a különböző adat- vagy szövegbányászati módszereiket.

Am még az olyan nagy látogatottságú és ismert webrendszereknél, mint az amazon.com vagy az eBay.com is igen nehéz a felhasználókat rávenni arra, hogy minduntalan értékeléseket, preferencia értékeket adjanak meg; egyrészt időigényes feladat, másrészt a felhasználók bizalmatlanok, egyre kevésbé adnak ki bármilyen információt is magukról.

Megoldás

Épp ezért egyre inkább az implicit adatgyűjtés felé mozdulnak az ez irányú kutatások. Az implicit adatgyűjtés során a web alapú rendszerek a felhasználó tudta, beavatkozása nélkül képesek folyamatosan, igen nagy mennyiségű adatot gyűjteni, majd azokat feldolgozni. A kutatásom ezen részében egy új, AJAX alapú technológiát és modellt kínálok az adatgyűjtés hatékonyabbá tételére.

Bár az eredményeket még 2012-ben közöltem az *AJAX-Based Data Collection Method for Recommender Systems* című cikkemben, az ott ismertetett módszer még ma is aktuális, idén egy **koreai** szerző trió, Sung Moon Bae és társai idézték a *Utilization of Demographic Analysis with IMDB User Ratings on the Recommendation of Movies* (DOI: 10.7838/jsebs.2014.19.3.125) cikkükben.

5. tézis - Reszponzív és tartalomfüggő megjelenítés

Probléma

A személyre szabott tartalom rendkívül fontos, az még inkább, hogy azt a felhasználónak tetsző módon szolgáltatassák a rendszerek. A korai webrendszerek egyik legnagyobb hátránya, hogy a felhasználónak nem biztosították az ún. “alkalmazás érzetet”, a desktop alkalmazások során megszokott “kattintunk és már azonnal ott is a tartalom” a web kliens-szerver architektúrájának sajátosságából adódóan nem érvényesült; amíg a böngésző a kérésére kapott választ nem kapja vissza a szervertől, addig vár, így a felhasználó is.

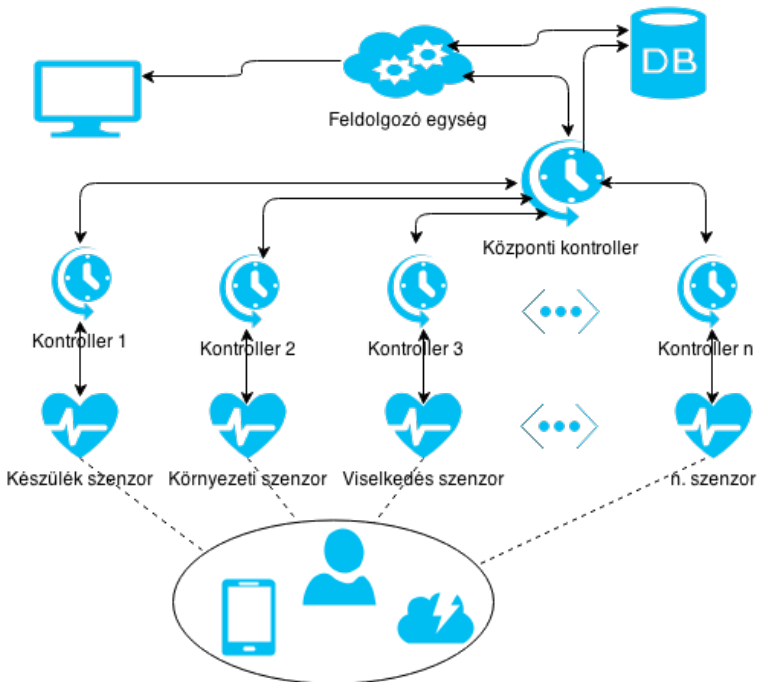
Mindannyian töltöttünk már súlyos másodperceket úgy a számítógép előtt, hogy vártuk, egy weblap betöltődjön, frissüljön. Ezt a hiányosságot a fejlesztők az AJAX asszinkron kommunikációs képességeivel igyekeznek orvosolni.

Mindezek mellett a mobil eszközök térhódításának köszönhetően, valamint amiatt, hogy az IPv6 bevezetésével akár a hűtőszekrényünkről vagy a kávéfőzőnkről is csatlakozhatunk az Internetre, egy új kihívással kell

szembenézniük a fejlesztőknek: már nem elég tartalomfüggő webrendszereket tervezni, megvalósítani, az eszközfüggő rezponzív weboldalaké a jövő.

Megoldás

Ennek megfelelően az ismertett modellel megvalósuló fejlesztések felhasználói interfésze intelligens módon alkalmazkodik az őt megjelenítő eszközhöz, annak típusához, képernyőfelbontásához. A felhasználóról szerzett információkat főbb jellemzőik alapján 3 különböző csoportba osztottuk, az egyes csoportok a Készülék, a Környezet és a Viselkedés kategóriákat jelentik és az alábbi szenzor-architektúra szerint végzik az adatgyűjtést.



4. ábra: Szenzor architektúra

A begyűjtött információk alapján aztán az alkalmazáserverünk összeállítja a felhasználó számára legmegfelelőbb tartalmat, felhasználói interfészt és megküldi a kliens eszköz számára.

6. tézis - Adatvédelem szakértői szemmel

Probléma

Joggal merülnek fel minduntalan személyiségi jogi kérdések az intelligens webrendszerek kapcsán, hisz napjaink egyik kulcskérdése az informatikai biztonság, a személyes adatok védelme. A láthatatlan és folyamatos adatgyűjtés – nem csak az intelligens rendszerek, hanem a teljes Interneten töltött életünk szempontjából – kiemelt jelentőséggel bír.

Igazságügyi szakértőként több olyan büntető ügyben is végeztem szakértői munkát, ahol az Internet használat által hagyott 'lábnyomokat' kellett felkutatni egy adott számítógépen.

Megoldás

A tapasztalatok meglepőek és tanulságosak. Volt szerencsém számos aspektusból vizsgálni a kérdést; úgy a felhasználó, mint a szakértő, a feltételezett bűnelkövető vagy a nyomozóhatóság szemszögéből. Kutatásom utolsó fázisa annak felderítésére és rendszerezésére irányult, hogy a webes rendszerek milyen információkat gyűjtenek, tárolnak a felhasználókról, hogyan, milyen eszközökkel, módszerekkel lehetséges ezeket az információkat felkutatni, illetve védekezni az ellen, hogy illetéktelen kezekbe kerüljön.

A témában született eredményeimről 2012-ben kettő cikket is írtam. Az első *Using Forensic Techniques for Internet Activity Reconstruction* néven, melyet **kínai** kutatók, Chen Long és társai *User browsing-data recovery of Google browser in private-browsing mode* (DOI: 0.3979/j.issn.1673-825X.2013.06.027) című írásukban idéznek, míg a második, a *Social media risks from forensic point of view* 2014-ben találtatott hasznosnak Mohammad Reza Keyvanpour és **iráni** kutatótársai számára a *Digital*

Forensics 2.0 (DOI: 10.1007/978-3-319-05885-6_2) publikációjukhoz használták fel. Külön öröm, hogy ez utóbbi cikkemet az **amerikai Védelmi Minisztérium** (Department of Defense) által fenntartott Cyber Security & Information Systems Information Analysis Center (CSIAAC) weboldala¹ is említi.

3 PREVIOUS RESEARCH AND THE MOTIVATION FOR THE THESIS

Introduction

In this dissertation I summarized my research work of the past five years, when I focused on the detailed analyses of the international results in intelligent web systems, in order to locate and identify areas, where interesting challenges and results may appear both in scientific and professional way.

As a result, it is concluded that research on intelligent web systems almost exclusively directed to the optimization of methods of core algorithms that work on existing data, although little is said about which technologies and methods are suitable for data collection and on the ways and forms that should be used to display the output data to the user.

In my research, I developed a new AJAX-based implicit data collection method, a new software development design pattern that fits in the nowadays modern methodologies, a method collection that is useful in improving performance of today's web systems and finally I searched, systematized and now I present the risk of web systems; where and how data is collected, stored about us, about the users.

Proper display of output data of intelligent web systems is a key regarding the aspect of user experience and market profit, so at first, I proposed a new model for existing and new web portals to make them faster

[1https://sw.csiac.org/techs/abstract/549148#.VDI_hb6A3dk](https://sw.csiac.org/techs/abstract/549148#.VDI_hb6A3dk)

and on the other hand, I have created a universal, responsive, content-aware system, which is suitable for any themed web systems.

As an unconcealed goal, my dissertation aims to bring closer to each other the academic research area and the technologies used in industry and business life. Academic researchers are often accused of that most of their research results are purely theoretical, away from reality or just takes years to be utilized for the industry. From this point of view information technology, which changes rapidly, is particularly sensitive area, as well as hardware and software technology.

I tried to address the latest, current problems of real industrial and business life, find solutions that use and apply the scientific results; beside the theoretical results, I offer practical, immediately usable and deployable methods as well.

To sum it up, the question that needs to be answered is: How can we make nowadays used intelligent web systems better?

Motivation and literature overview

In 2012, on my flight towards an international conference I got the current issue of Traveller journal, where Andrew Hankinson's "Holiday 2.0: are we changing the way we travel forever?" article described the important role of online booking systems and social media sites in tourism. He analyzed websites and services, which helped him to organize his whole holiday including traveling, accommodation, catering, excursions and meeting friends.

He analyzes the phenomenon both from the aspect of a journalist and an investor, then draw a number of conclusions that fully match our experience and opinion: it is a question of staying alive for all market participants to take all the advantages of Internet based (e-) business possibilities. However, a great website or a well-advertised accommodation worth nothing if a visitor, the tourist does not trust in the given webpage. Trust is

the most valuable currency of our Internet life, statistics proof [19] [20], that social media sites, the community and the opinion of the most influential people in the community have significant role in our business decisions, majority of travelers trust more in other travelers' social media comments than in a brochure of a travel agency.

There is another question of confidence related to a website or an information system: this is about the protection of our personal and financial information. In this case the question is not a right or a wrong business decision, but –sometimes- our whole existence is at stake. Thus, if we hear news about phishing, wire-tapping or surveillance scandal day by day we should not be surprised if users are mistrustful. This fact complicates the life of intelligent web system builders and imposes heavy tasks on marketing and IT experts to arouse confidence in a brand, a service or a website.

Does the Internet really change our life and shopping habits? No doubt. The number of Internet users and searchers have been growing continuously. While in 2008 this number was 1.4 billion, in 2014 it was 2.4 billion ergo more than 40 percent of Earth's population uses the Internet regularly (Fig. 5.). It means that in every single second 7 new information-hungry user appears in the world of Internet.

Internet usage statistics between 2008 and 2014

Source: www.verisigninc.com, www.internetworldstats.com, www.internetlivestats.com, royal.pingdom.com, www.statista.com

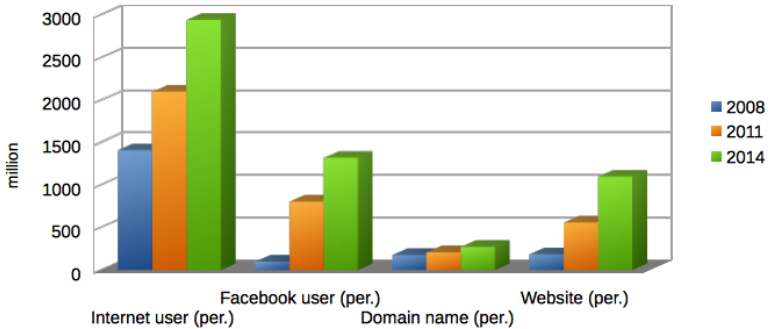


Fig. 5.: Internet usage between 2008 and 2014

Both the initial and the current objective of my research work is to make web based systems more effective either in desktop and mobile environment. An efficient web system is intelligent: personalized (gender, age, job, nationality, interest, geographical, language preferences) and provides device (computer, phone, tablet or refrigerator) based content. Doing it as fast as possible is based on statistical reports 57 percent of users abandon web sites if the page does not load in 3 seconds. According to other reports, every second delay in page loading time gets out millions of dollars from the pocket of online retailers [22].

It may be now understandable, why the effective, fast and user-friendly web systems have an extreme importance in the business world.

According to the trends of our time, Internet is the primary information and news source as well as the place of service and commercial activities. People of today order product or service on the Internet, read daily news on web pages and since the appearance of Web 2.0 they comment blogs, forums, write travelogue, get informed via social media sites about their relatives or friends.

The Web 2.0 concept initially appeared in the article of DiNucci [1] although Tim O'Reilly and Dale Dougherty gained popularity to the phrase

at a Web 2.0 conference in 2005. Nowadays it is much more than a marketing phrase, in fact it is generic name of new technologies [2].

New technologies set up new challenges for software developers. In the age of Web 2.0 web development is not about displaying information on web pages, it is about building web applications which are capable to share information efficiently, provide rich user experience and harness the collective intelligence of the community.

All the mentioned three pillars (information sharing, user experience, intelligence) are important, my researches belong to all domain, in my dissertation I will present my related results.

Intelligent Web

Although Web is a rich information source, data collection, organization gives more serious task to systems that serves the users. That is why newer and newer technologies appear as a result of researches and developments, in order to serve proper content to web users. The generic name of these newest technologies is intelligent web.

The phrase of web intelligence (WI) appeared in 2000 [5], in the opinion of *Ning Zhong et al.* intelligent web is a generic term of artificial intelligence, knowledge representation, data mining, intelligent agents and intelligent social network concepts and novel application of information collected on the mentioned fields.

Additionally, WI stimulates IT and artificial intelligence experts to solve new problems and challenges. WI technologies revolutionize known methods and technologies of information collection, storage, process and presentation.

After 2000, it was quite popular dealing with intelligent web, the popularity was not by accident, WI and especially web personalization within is one of that research areas that gives brilliant toolkit and economical benefit to e-business industry. Capabilities of the technology like monitoring, storing and analyzing users' shopping or browsing habits

gives more efficient tools ever into the hands of merchants to precisely target, customize their product or service.

One of the best methods of growing user experience is web personalization, it helps us to achieve that every single user sees his or her customized version of the same webpage. Thanks to the technology, web servers are capable to fit web pages to customer's need in real time, thereby ensuring that special, unique user experience where customer feels the webpage can almost read in his/her mind.

Several researchers dealt with the domain of web personalization, thus A.R Simon et al. [8] from the aspect of targeted marketing, Ning Zhong et al. from the wisdom web point of view [9], Su Ho Ha examined the impact of effect on customer decisions [10], while Wang and Kobsa [11] worked out a proper method considering privacy.

Problems

Researches related to intelligent web systems are facing only the problem to make core algorithms more effective, only few words are mentioned about how, with what kind of method, technologies can someone collect initial data and in what kind of form we should provide the results to the user.

Both input and output have significant role as even the most perfect algorithm worth nothing without input data as even an effective artificial intelligence can work behind a slow or bad web portal if potential customers, visitors escape from that specific website instead of browsing it.

We may say, what could be simpler than collecting data about users, as Internet is exactly about billions of data. From the side of intelligent systems data about users and users' behavior has important role, it can be collected in two ways: we may ask the user to provide information (e.g. forms, surveys, ratings) – this is the so called explicit data collection or our intelligent system collects data (browsing paths, clicks, visited pages, shopping history, IP address, language settings, etc.) in the background without disturbing users – we call this method implicit data collection.

Two technologies can be combined, however both have disadvantages: explicit data collection is cumbersome, users are mistrustful and reluctant to give out information and on the other hand they also regret time to fill out forms for our favor.

Implicit data collection eliminates the previously mentioned problems and another advantage of it that we can collect huge amount of data by using this technology. However, due to the nature of it, it raises an important question: can we collect any kind of user data without the users' permission? Do we even know what kind of information is collected and stored by the underlying information system at the moment when we open a webpage?

Web engineering

Development of a web-based systems gives much more tasks to developers than a traditional software development; the life cycle of the system, development process, tracking and maintenance all differ from normal software development. Therefore it is clear why traditional development methods are not suitable for web-based system in many cases, they need clarifications, corrections. Powell excellently put it to the point in his book published in 2000:

“(Web engineering) involve a mixture between print publishing and software development, between marketing and computing, between internal communications and external relations, and between art and technology” [23].

Different literatures define the concept of web engineering on different but basically in a similar way. We can summarize that web engineering is nothing more than a method for developing and organizing knowledge related to web application development. I mostly agree with Murugesan et al., they say web engineering is the systematic application of scientific, engineering and management knowledge in order to develop, deploy and maintain web-based systems [24].

Features of web-based systems were formulated many times, many ways [25,26,27,28], these show, web-based systems differ from traditional software systems, it is so true for development and design process.

Software development process is complicated and complex. Certain parts of it and the relations among them require modeling in order to make the process modular and easily understandable.

The most known and widespread model is the waterfall model that was firstly mentioned by Winston W. Royce in his 1970 article [29], although it is worth to mention that the *waterfall* concept itself was never mentioned in the paper. In fact, the article describes the drawbacks and disadvantages of this kind of development process.

This model is excellent in cases when we know all the system requirements at the very beginning of the development process. It requires thorough survey and research work, not to mention that customer should be well prepared according to the required functionalities as changes during the development are costly enough after certain steps to restart the whole development process.

Unfortunately, in real life, customers cannot define exactly what they want at the beginning of the project; requirements are changed and refined during development process. That is why the waterfall model is declined in many cases during business web development. The waterfall model is not prepared for this kind of changes; when a development phase has been completed, it is almost impossible to make any changes on it.

We need a method, which is quite flexible and makes it possible to modify the design and system development process and architecture based on continuous customer consultations at the lowest possible cost.

A need was appeared for an application of a new, more iterative and interactive method in Royce's paper, that is why so many waterfall model alternatives and improved methodology were born in the past decades.

Biró describes the characteristics of regularly appearing methodologies in a separate article [30], the newest hype still exists, its roots go back into 1988 to Boehm's spiral [31] and Gilb's evolution model [32].

This new trend is the agile software development that is the opposite of the waterfall model in many cases. When they are ready, developers give smaller modules to the customers for testing, where the user can try it, can clarify his / her needs and developers can efficiently modify the module reacting for the new requirements, then they conciliate with the customer again; continuous consultation is the basis of agile software development. This method can guarantee user satisfaction, the customer gets what he / she wants even though he / she could not draw exactly the requirements at the very beginning of the project.

Problems

Riding only on this new agile method fashion way, we can definitely say to choose this for today's development process, however this method has drawbacks. Continuous consultation is a great tool, but time consuming and even if customer has no time, it can pull back the development. Because of the possibility of permanent revision and changes, much more professionals (graphic designer, system designer, front-end, back-end developer) must be available at the same time opposed to the waterfall model. In addition, the anticipated cost and completion date is also difficult to estimate.

A reasonable assumption we can offer more efficient method for web development by combining the benefits of the two systems. Consulting with business stakeholders and my own development experience were great assistance for creating a hybrid system, where we could identify mistakes, critical points that may be reasoned by using inappropriate methodology or just ignoring and methodology in application development process.

Additionally, we should emphasize the importance of user interface and design, thus the developed hybrid method also includes development model for it. Current models elegantly crossed at this point or even mixed it with

the system plan, even though the „look & feel” has major importance for the customer and prospective users.

4 THE NEW SCIENTIFIC RESULTS OF THE THESIS

The dissertation consists of six chapters which are closely related to each other where I overview the whole process of a web-based system. In the short reviews of the chapters I describe the initial problem and give my new research results and methods for the discussed problem.

It is just happened that I got insight into the scientific world as a young researcher, therefore it is my pleasure, that foreign researchers cite and use my results and my articles, so please allow me to mention the citations where it is relevant.

In the *first chapter* I reviewed the development process of web-systems, detected the shortcomings of currently used models and developed a new method that meets the demands of today's need.

The *second chapter* highlights the system design from the entire development process, describes in detail the traditional and trendy design patterns used today and then I worked out a new, MVC-based design pattern that is also suitable for the client-server framework integration as well.

The *third chapter* describes the performance shortcomings of current market leader web systems, reviews existing speed increasing technologies and offers a new process model that can be used to build fast web systems or improve existing web portal speed performance.

Chapter four presents an interesting, AJAX-based implicit data acquisition technology, which can make data collection very effective, serves the needs of responsive presentation and offers ideal data collection possibility for the recommender systems.

The *fifth chapter* is intended to be used in making better the component of the software system directly contact with the user. The user interface

developed by the improvements of the new model adapts intelligently for its display device, for its type and its screen resolution.

Finally, in the last, *sixth chapter* I explore, analyse and present the privacy risks of existing intelligent systems, I point out what risks are faced in our personal data in the Web 2.0 era.

Thesis 1 – Adaptive Development Method

Problem

Software development process is complicated and complex. Certain parts of it and the relations among them require modeling in order to make the process modular and easily understandable. The most known and widespread model is the waterfall model. This model is excellent in cases when we know all the system requirements at the very beginning of the development process.

Unfortunately, in real life, customers cannot define exactly what they want at the beginning of the project; requirements change and refine during development process. That is why the waterfall model is declined in many cases during business web development. The waterfall model is not prepared for this kind of changes; when a development phase has been completed, it is almost impossible to make any changes on it.

Solution

We need a method, which is quite flexible and makes it possible to modify the design and system development process and architecture based on continuous customer consultations. Such a method is agile software development, although it cannot be used in many cases. Based on my research-development work, I combined both systems and created a new development model fitting today's modern web development process model, this is the adaptive development method.

I expounded my new results in the following article: *Adaptive Design Process for Responsive Web Development* (DOI: 10.13140/2.1.3354.5601).

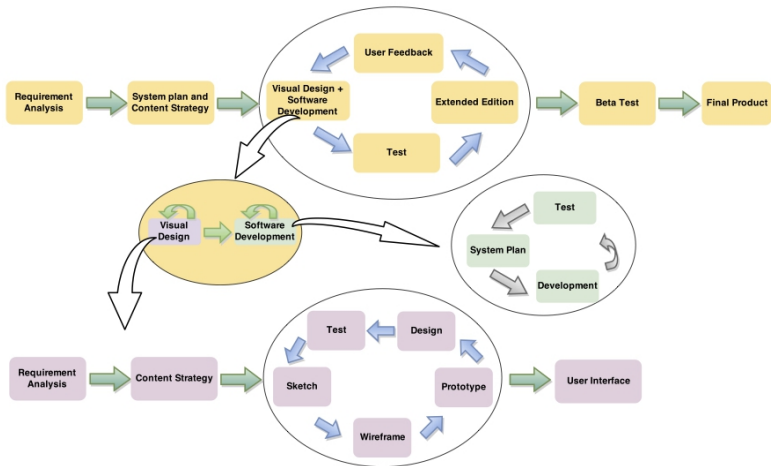


Fig. 1: Adaptive Development Method

Thesis 2 – Integrated Design Pattern

Problem

Nowadays it is impossible to develop web applications without proper design patterns, as developers must serve both rich client-side programming tasks, as well as usual server side engineering and coding. Whether the client side or server-side programming we are talking about, the volume of the work requires the usage of design patterns. It is cumulatively true for a complex application, where client and server side development is in necessarily indivisible relationship.

The most popular design pattern is MVC. In this phase of my research I am searching for the answer whether this pattern is still suitable for nowadays development environments, whether it is possible to offer to the

developers a new, better architecture integrating client and server side systems.

Solution

After examination of different design patterns I concluded that the original MVC architecture, or more precisely, its modified, Cocoa version is the perfect initial state for developing a new integrated design pattern. Furthermore, my choice is verified by the fact that MVC frameworks are the most popular development environments, both on the client and the server side.

Thus the real question is, how we can connect a JavaScript MVC with a PHP MVC framework in such a way that the new, integrated system also fits into the Model-View-Controller architecture.

Approaching from the server side, it is obvious that a View component needs further segmentation, as View complexity makes the development work harder. If we change the View component to a whole client-side MVC, our new system be-comes an M(MVC)C.

In the case of client-side MVC, the Model component of an MVC pattern is a simple HTML code, and the View is a CSS file, (one CSS file means one View), whereas the Controller is the browser itself that assigns HTML with CSS. From another approach, View is a user interface born from a combination of an HTML, CSS and data. JavaScript classes and methods play the role of a Controller, and the Model is the data coming from the web server.

Whichever aspect we take, the Model is the interface where a server-side MVC system could be attached. Based on this result the following figure describes the new design pattern.

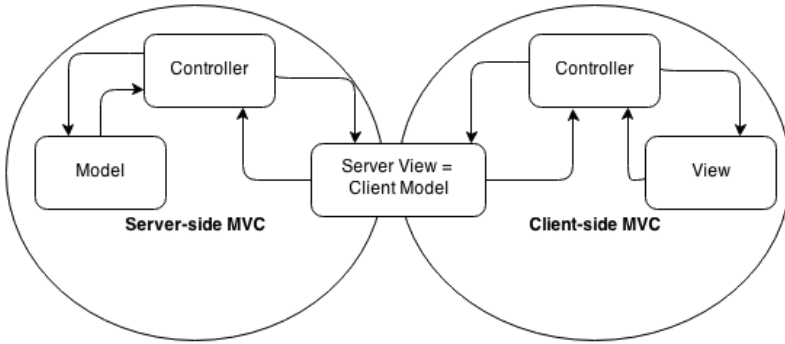


Fig. 2: Integrated Design Pattern

The developed new pattern was presented on SOFA2014 International Conference and my 14 pages article *Integrated Design Pattern for Intelligent Web Applications* was also accepted to appear.

Thesis 3 – Model for faster web system performance

Problem

As it is mentioned in the introduction, several research confirm: a web system, based on intelligent, excellent mathematical model, worths nothing if the user can hardly wait for displaying the content. Clear proof of the importance of speed is that from 2010 Google has also begun to use webpage speed as an evaluation parameter in its PageRank algorithm.

Several interesting book and scientific papers were published about making web pages faster, even the two biggest search engines, Google and Yahoo have also presented recommendations and measurement methods in this issue. However, based on surveys there are only few web pages in the world that fully meet the proposed technical guidelines.

Examining the top 10 visited webpages of the world, of the United States and of Hungary shows that our country is lagging behind using these new recommendations and methods.

Solution

In the third phase of my research I tested and applied methods proposed by literature and own experiences on a 4000 unique visitors a day web portal. Based on the results of performance analysis I created a method collection that is suitable not only for converting existing web pages, but can also be applied in design phase, thus greatly facilitate the births of faster intelligent web systems.

With the title *Improved Speed on Intelligent Web Sites* I published my related results in 2013, which arose the attention of **Egyptian** scientists, they cite my work in their *XML Schema-Based Minification for Communication of Security Information and Event Management (SIEM) Systems in Cloud Environments* (DOI: 0.14569/IJACSA.2014.050912) paper published in International Journal of Advanced Computer Science and Applications.

Thesis 4 – Implicit data collection, a new model

Problem

Using recommender systems is the main tool for implementing personalized web content. However, most of current research face a problem that it is very hard to gather real data in adequate quantity and quality, so the majority of scientists work on sample databases or the so fortunate minority of them can apply their data and text mining methods on real systems.

But even in the case of highly frequented and well known web systems like amazon.com or eBay.com it is very difficult to convince users to constantly fill out surveys, provide preference values. First of all, it is a time

consuming task, on the other hand, users are wary, they tend to give out information about themselves less and less frequently.

Solution

That is why related researches are moving toward to implicit data collection. During implicit data collection web based systems can continuously collect and process large amount of data in an unobtrusive way. In this part of my research I provide a new, AJAX-based technology to make data acquisition more efficiently.

Although I published the results in my *AJAX-Based Data Collection Method for Recommender Systems* title article in 2012, the method described therein is still current, a trio of **Korean** researchers, Sung Moon Bae et al cited it in their *Utilization of Demographic Analysis with IMDB User Ratings on the Recommendation of Movies* (DOI: 10.7838/jsebs.2014.19.3.125) paper.

Thesis 5 – Responsive and content-aware presentation

Problem

Personalized web content is extremely important just as the fact that systems can be able to provide it in a user friendly way. The major drawback of earlier web systems is that they have not provided the so-called “application feeling” to the user, the method “we click and get the content immediately” which was familiar in desktop environment has not been prevailed due to the characteristics of a client-server architecture.

The browser waits until it gets the answer for his request, so does the user as well. Each of us has spent several seconds in front of a computer waiting for web page loading or refresh. Developers intended to remedy this deficiency with the asynchronous communication capabilities of AJAX.

In addition, thanks to mobile device penetration and due to the introduction of Ipv6 as we can connect the Internet from our refrigerator or coffee maker as well, developers have to face a new challenge: it is not enough to develop content-dependent web systems, device-dependent webpages will own the future.

Solution

Accordingly, user interfaces developed with the mentioned model adapts intelligently to the device that display it, to its type and screen resolution. The information acquired about users is divided into three different groups: Device, Environment and Behavior. These categories carry on the data collection based on the following sensor architecture.

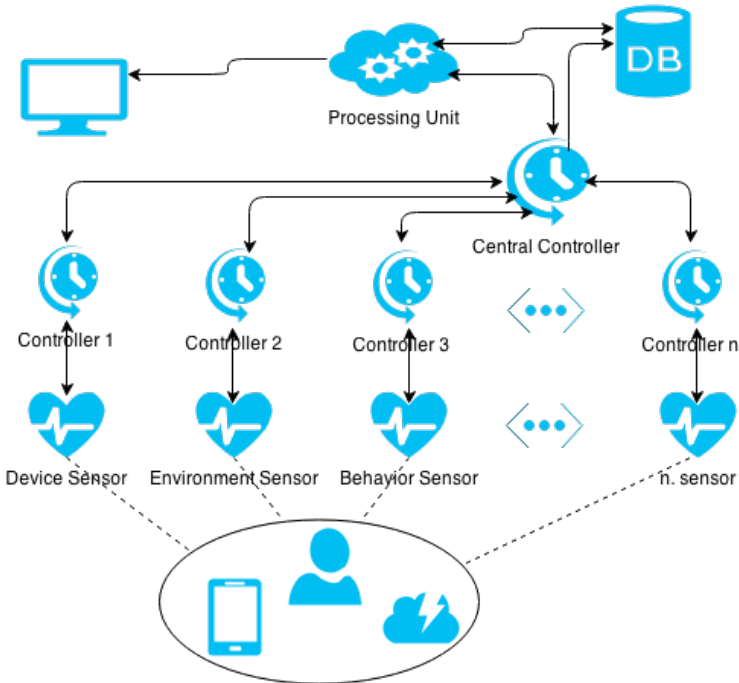


Fig. 6.: Sensor architecture

Based on the information collected, the application server assemble the most appropriate content for the user, create the user interface and send it to the client device.

Thesis 6 – Privacy from forensic point of view

Problem

Right to privacy issues constantly arise related to intelligent web systems, because information security and personal data protection are key issues today. The invisible and continuous data collection has paramount importance not only of intelligent systems, but in terms of our entire life spent on the Internet.

As a forensic expert I worked in several criminal cases, where I had to discover the footprints of Internet usage on a given computer.

Solution

Experiences are surprising and enlightening. I had the pleasure to examine several aspects of the issue; so the point of view of the user, of the expert, of the criminal or of the investigating authority. The last phase of my research is focused on finding out the kind of information that has been collected and stored about a user by web-based systems, how and by what kind of tools, methods is it possible to identify and to defence against unauthorized hands.

I wrote two papers about results in this area in 2012. The first one was *Using Forensic Techniques for Internet Activity Reconstruction*, which was cited by **Chinese** researchers, Chen Long et al cited it in their *User browsing-data recovery of Google browser in private-browsing mode* (DOI: 0.3979/j.issn.1673-825X.2013.06.027) article, while the second one, *Social media risks from forensic point of view* was useful for Mohammad Reza Keyvanpour and his **Iranian** researcher colleagues in 2014. They cited it in their *Digital Forensics 2.0* (DOI: 10.1007/978-3-319-05885-6_2) paper.

It is a special pleasure that the latter article is mentioned by the web page of Information Systems Information Analysis Center (CSIAC) supported by the **american Department of Defense**.

5 IRODALOMJEGYZÉK / REFERENCES

1. DiNucci, Darcy: "Fragmented Future" (PDF). Print 53 (4): 32, 1999
2. O'Reilly T.: What Is Web 2.0, <http://oreilly.com/pub/a/web2/archive/what-is-web-20.html>, 2005
3. Berners-Lee T.: Plenary talk at WWWF94, <http://www.w3.org/Talks/WWW94Tim/>, 1994
4. Berners-Lee T., Hendler J., Lassila O.: The Semantic Web. *Scientific American*, 284(5):34–43, 2001
5. Zhong N., Liu J., Yao Y. Y., Ohsuga S.: Web intelligence (WI), Proc. 24th IEEE Computer Society International Computer Software and Applications Conference (COMPSAC 2000), Piscataway, NJ: IEEE CS Press, pp. 469–470, 2000
6. Liu J.: "Web Intelligence (WI): What Makes Wisdom Web?", in Proc. of the Eighteenth International Joint Conference on Artificial Intelligence, 1596-1602, 2003
7. Terziyan V.: *Intelligent Web Applications*, University of Jyvaskyla, 2002
8. Simon A. R., Shaffer S. L.: *Data Warehousing and Business Intelligence for e-Commerce*, San Francisco, CA: Morgan Kaufmann, 2001
9. Zhong N., Liu J., Yao Y.: "In Search of the Wisdom Web", in special issue on Web Intelligence (WI), *IEEE Computer*, Vol. 35, No. 11 (November 2002) 27-31.
10. Su Ho Ha: Helping Online Customers Decide through Web Personalization Intelligent Systems, *IEEE* (Vol. 17 , Issue: 6), 2002, pp. 34-43

11. Wang Y., Kobsa A.: Technical Solutions for Privacy-Enhanced Personalization, Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies. Hershey, IGI Global, 2008
12. Su X., Khoshgoftaar T. M.: A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, vol. 2009, Nr. 4, 2009.
13. Pazzani M. J., Billsus D.: Content-Based Recommendation Systems. *The Adaptive Web, Lecture Notes In Computer Science*, Vol. 4321, 2007, pp 325–341
14. Burke R.: Knowledge-Based Recommender Systems. *Encyclopedia of Library and Information Science*, 69 (32), 2000
15. Pazzani M. J.: A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, Vol. 13, Nr. (5-6), 1999, pp. 393-408
16. Burke R.: Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User- Adapted Interaction*, Vol. 12 Nr. 4, 2002, pp. 331–370
17. Melville P., Mooney R. J., Nagarajan R.: Content-Boosted Collaborative Filtering for Improved Recommendations *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI-2002)*, pp. 187-192, Edmonton, Canada, July 2002
18. Sobeci J.: Implementations of Web-based Recommender Systems Using Hybrid Methods, *International Journal of Computer Science & Applications* Vol. 3 Issue 3, 2006, pp 52 – 64
19. Nagy Z.: Turisztikai portálok szolgáltatásai most és a jövőben, V. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia Kiadványa, Nyíregyháza, 2011, pp. 98-102
20. Lütters H.: web 2.0 in the tourism industry – status quo, ITB Berlin, Berlin, 2010

21. A.L. Barabási: Behálózva. A hálózatok tudománya, Budapest, Helikon Kiadó, 2013
22. Nagy Z.: Improved Speed on Intelligent Web Sites, Recent Advances in Computer Science, Rhodes Island, Greece, 2013, pp. 215-220
23. Powell T. A.: Web design: The complete guide. New York: McGraw-Hill, 2000.
24. Murugesan S. Ginige A.: Web Engineering: Introduction and Perspectives. Software Engineering, 1999
25. Deshpande Y., et al.: Web engineering. Journal of Web Engineering, 1(1), 3-17, 2002
26. Deshpande Y., Hansen S.: Web Engineering: Creating a Discipline among Disciplines, IEEE Multimedia, Special issues on Web Engineering, vol 8, no 2, pp 82-87, 2001
27. Pressman R. S.: Web Engineering: An Adult's Guide to Developing Internet-Based Applications, Cutter IT Journal, vol 14, no. 7, pp 2-5, 2001
28. Ginige A., Murugesan S.: The essence of Web engineering: Managing the diversity and complexity of Web application development. IEEE Multimedia, 8(2), 22-25., 2001
29. Royce W. W.: Managing the Development of Large Software Systems, Proceedings of IEEE WESCON 26 (August): 1-9, 1970
30. Biró M.: The Software Process Improvement Hype Cycle, UPGRADE - CEPIS 10: (5) pp. 14-20.,2009
31. Boehm B.: A Spiral Model of Software Development and Enhancement, IEEE Computer, IEEE, 21(5):61-72, 1988

32. Gilb T.: Principles of Software Engineering Management, Addison-Wesley, 1988
33. Adamkó A.: Web Alapú Információs Rendszerek Modellezése, Doktori Értekezés, Debreceni Egyetem, 2008
34. Ginige A., Murugesan S.: Web engineering: A methodology for developing scalable, maintainable Web applications. Cutter IT Journal, 14(7), 24-35., 2001
35. Viljami S.: Responsive Workflow,
<http://viljamis.com/blog/2012/responsive-workflow/>, 2012
36. Boulton M.: Responsive Summit Workflow,
<http://www.markboulton.co.uk/journal/responsive-summit-workflow>, 2012
37. UXMovement: Why it is important to sketch before you wireframe,
<http://uxmovement.com/wireframes/why-its-important-to-sketch-before-you-wireframe/>, 2012
38. Blodget H., Danova T.: The Future of Digital:2013,
<http://www.businessinsider.com/the-future-of-digital-2013-2013-11?op=1>, Business Insider, 2013
39. Reenskaug T.: The Original MVC Reports 1979,
<http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>, 2007
40. Apple Developer: Cocoa Core Competencies: Model-View-Controller,
<https://developer.apple.com/library/mac/documentation/general/conceptual/devpedia-cocoacore/MVC.html>, Szeptember 2013
41. Potel M.: MVP: Model-View-Presenter The Taligent Programming Model for C++ and Java. Taligent Inc., 1996
42. Greer D.: Interactive application architecture patterns.
<http://aspiringcraftsman.com/2007/08/25/interactive-application-architecture/>, 2007

43. Bower A., McGlashan B.: Twisting the triad: The evolution of the dolphin smalltalk mvp application framework. Tutorial Paper for ESUG, 2000.
44. Fowler M.: Presentation model
<http://martinfowler.com/eaDev/PresentationModel.html>, 2004
45. Smith J.: WPF Apps With The Model-View-ViewModel Design Pattern,
<http://msdn.microsoft.com/en-us/magazine/dd419663.aspx>, 2009
46. Microsoft Developer Network: Implementing the Model-View-ViewModel Pattern, <http://msdn.microsoft.com/en-us/library/ff798384.aspx>
47. Smith J.: Using MVC to Unit Test WPF Applications,
<http://www.codeproject.com/Articles/23241/Using-MVC-to-Unit-Test-WPF-Applications>, 2008
48. Rossi G., Schwabe D., Guimarães R.: Designing personalized web applications. In Proceedings of the 10th international conference on World Wide Web, ACM, 2001, pp. 275-284.
49. Pazzani M. J., Billsus D.: Content-Based Recommendation Systems. The Adaptive Web, Lecture Notes In Computer Science, Vol. 4321, 2007, pp 325–341
50. Su X., Khoshgoftaar T. M.: A Survey of Collaborative Filtering Techniques. Advances in Artificial Intelligence, vol. 2009, Nr. 4, 2009.
51. Dell’Aglia D., Celino I., Cerizza D.: Anatomy of a Semantic Web-enabled Knowledge- based Recommender System, <http://www.larkc.eu>, 2010
52. Marcotte E.: Responsive web design. A List Apart, 306., 2010,
<http://alistapart.com/article/responsive-web-design>
53. StrangeLoop Networks: Website abandonment happens after 3 seconds,
http://www.strangeloopnetworks.com/resources/info_graphics/web-performance-and-user-expectations/website-abandonment-happens-after-3-seconds/, 2010

54. Lohr S.: Impatient web users flee slow loading sites, New York Times, 2012 <http://www.nytimes.com/2012/03/01/technology/impatient-web-users-flee-slow-loading-sites.html>
55. Kissmetrics.com: Loading time, <http://blog.kissmetrics.com/loading-time/>, 2013
56. Ecoconsultancy.com: Slow loading websites cost retailers 1,73 bn in lost sales each year, <http://econsultancy.com/uk/blog/9790-slow-loading-websites-cost-retailers-1-73bn-in-lost-sales-each-year>, 2013
57. StrangeLoop Networks: Internet users have faulty perceptions of time, <http://www.strangeloopnetworks.com/resources/infographics/web-performance-and-user-expectations/internet-users-have-faulty-perceptions-of-time/>, 2013
58. Langville N., Meyer Carl D.: Deeper Inside PageRank, Internet Mathematics, Vol 1, No. 3, pp. 335-400, 2004
59. Econsultancy: It's official: Google incorporates website speed into your ranking, <http://econsultancy.com/uk/blog/5728-it-s-official-google-incorporates-website-speed-into-your-ranking>, 2013
60. Google Developer: Make the web faster, <https://developers.google.com/speed/>, 2013
61. Yahoo: Exceptional Performance, <http://developer.yahoo.com/performance/>, 2013
62. Souders S.: High Performance Web Sites. O'Reilly Media, 2007
63. Souders S.: Even faster web sites: performance best practices for web developers. O'Reilly Media, 2009
64. IETF: Hypertext Transfer Protocol version 2.0, <http://tools.ietf.org/html/draft-ietf-httpbis-http2-07>, 2013
65. Willis N.: Reducing HTTP latency with SPDY, <http://lwn.net/Articles/362473/>, 2009

66. IETF: Hypertext Transfer Protocol (HTTP) Keep-Alive Header, <http://tools.ietf.org/id/draft-thomson-hybi-http-timeout-01.html>, 2012
67. Sixrevisions.com: Free website speed testing, <http://sixrevisions.com/tools/free-website-speed-testing/>, 2013
68. StatCounter: <http://gs.statcounter.com/#resolution-ww-monthly-201306-201308-bar>, 2013 augusztus
69. Ramachandran S.: Web metrics: Size and number of resources, <https://developers.google.com/speed/articles/web-metrics>, 2013
70. Akamai Technologies: The State of the Internet, 2013, vol 6. num. 1 http://www.akamai.com/dl/akamai/akamai_soti_q113.pdf?WT.mc_id=soti_Q113
71. KSH: Statisztikai Tükör Távközlés, internet, kábeltelevízió, 2012, <http://www.ksh.hu/docs/hun/xftp/idoszaki/tavkint/tavkint12.pdf>
72. KSH: Statisztikai Tükör Távközlés, internet, kábeltelevízió, 2008, <http://www.ksh.hu/docs/hun/xftp/idoszaki/tavkint/tavkint08.pdf>
73. KSH: Statisztikai Tükör Távközlés, internet, kábeltelevízió, 2010, <http://www.ksh.hu/docs/hun/xftp/idoszaki/tavkint/tavkint10.pdf>
74. Techyfuzz.com: <http://techyfuzz.com/free-cdn-content-delivery-network-services-website/>, 2013
75. Wilde E., Gaedke M.: Web Engineering Revisited. In: BCS Int. Acad. Conf. 2008. p. 41-50.
76. Allaire J.: Macromedia Flash MX - A next-generation rich client. <http://download.macromedia.com/pub/flash/whitepapers/richclient.pdf>, 2002
77. Bozzon A., Comai S., Fraternali P., Carughi G. T.: Capturing RIA concepts in a web modeling language. In Proceedings of the 15th international Conference on World Wide Web WWW '06. ACM, New York, 907-908, 2006

78. Meliá S., Gómez J., Pérez S., Díaz O.: A Model-Driven Development for GWT-based Rich Internet Applications with OOH4RIA. Proc. 8th Int. Conf. on Web Engineering (ICWE'08). IEEE, pp.13 – 23, 2008.
79. Deitel P. J., Deitel H. M.: AJAX, Rich Internet Applications, and Web Development for Programmers, Prentice Hall International, 2008
80. Busch M., Koch N.: Rich Internet Applications. Technical Report 0902, Institute for Informatics, Ludwig-Maximilians-Universität München, 2009.
81. Garrett J. J. : Ajax: A New Approach to Web Applications:
<http://www.adaptivepath.com>, 2005
82. Costello E.: Remote Scripting with IFRAME,
<http://www.oreillynet.com/pub/a/javascript/2002/02/08/iframe.html>, 2002
83. Nagy Z.: Ajax 2 in 1: Interactive education and modern web technology, Journal of Applied Multimedia, vol.: 1, num.: 1, 2010, pp. 39-44
84. W3Trends: Usage of Javascript libraries for websites,
http://w3techs.com/technologies/overview/javascript_library/all, 2013
85. Pingdom: The Web loves jQuery, and here are the numbers to prove it,
<http://royal.pingdom.com/2012/06/20/jquery-numbers/>, 2012
86. W3Trends.com: Usage of server-side programming languages for websites,
http://w3techs.com/technologies/overview/programming_language/all,
2014 január / January 2014
87. WebTrends: Usage of content management systems for websites,
http://w3techs.com/technologies/overview/content_management/all, 2014
január / January 2014

88. Google Developers: Building Smartphone-Optimized Websites, <https://developers.google.com/webmasters/smartphone-sites/details>, 2013 december / December 2013
89. W3C: Media Queries, <http://www.w3.org/TR/css3-mediaqueries/>, 2012 június / June 2012
90. Nagy Z.: Social media risks from forensic point of view, International Journal of Computers and Communications 6 (4), 245-253, 2012
91. LiveCareer.com: Your Facebook Profile Could Affect Your Hiring Potential, <http://www.livecareer.com/press-releases/your-facebook-profile-could-affect-your-hiring-potential>, 2012 május / May 2012

6 PUBLIKÁCIÓK / PUBLICATIONS



DEBRECENI EGYETEM
EGYETEMI ÉS NEMZETI KÖNYVTÁR
PUBLIKÁCIÓK



Iktatószám: DEENKÉTK/383/2014.
Tételszám:
Tárgy: PhD Publikációs Lista

Jelölt: Nagy Zsolt

Neptun kód: PPFXF8

Doktori Iskola: Informatikai Tudományok Doktori Iskola

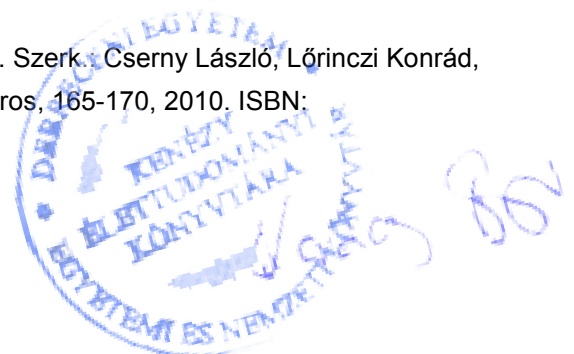
A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemény(ek) külföldi folyóiratban (2)

1. **Nagy, Z.:** Social media risks from forensic point of view.
Int. J. Computers Communications. 6 (4), 245-253, 2012. ISSN: 2074-1294.
2. **Nagy, Z.:** Ajax 2 in 1: Interactive education and modern web technology.
JAMPAPER. 5 (1), 39-44, 2010. EISSN: 1789-6967.

Magyar nyelvű konferencia közlemény(ek) (2)

3. **Nagy Z.:** Turisztikai portálok szolgáltatásai most és a jövőben.
In: V. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia Kiadványa, CD [elektronikus dokumentum]. Szerk.: Kerekes Benedek, Gát György, Nyíregyházi Főiskola, Nyíregyháza, 98-102, 2012. ISBN: 97896399099
4. **Nagy Z.:** Web 2.0: Az Intelligens Web.
In: Informatika Korszerű Technikai Konferencia 2010. Szerk.: Cserny László, Lőrinczi Konrád, Dunaújvárosi Főiskola Informatikai Intézet, Dunaújváros, 165-170, 2010. ISBN: 9789639915381





Idegen nyelvű konferencia közlemény(ek) (6)

5. **Nagy, Z.:** Adaptive Design Process for Responsive Web Development.

In: Advances in Information Science and Applications : Proceedings of the 18th International Conference on Computers (part of CSCC '14). [s.n.], [s.l.], 410-414, 2014. ISBN:

9781618042378

DOI: <http://dx.doi.org/10.13140/2.1.3354.5601>

6. **Nagy, Z.:** Improved Speed on Intelligent Web Sites.

In: Recent Advances in Computer Science: Proceedings of the 17th International Conference on Computers. Ed.: Ognian Nakov, et al, WSEAS Press, [Greece], 215-220, 2013. ISBN:

9789604743117

7. **Nagy, Z.:** Using Forensic Techniques for Internet Activity Reconstruction.

In: Recent Researches in Communications and Computers : Proceedings of the 16th WSEAS International Conference on Computers (part of CSCC '12). Ed.: Sandra Sendra, José Carlos Metrôlho, World Scientific and Engineering Society Press, [Greece], 248-253, 2012. ISBN:

9781618041098

8. **Nagy, Z.:** AJAX-Based Data Collection Method for Recommender Systems.

In: Recent researches in communications and computers : proceedings of the 16th WSEAS international conference on communications (part of CSCC '12). Ed.: Sandra Sendra, José Carlos Metrôlho, WSEAS Press, [Greece], 446-451, [2012]. ISBN: 9781618041098

9. **Nagy, Z.:** Intelligent Web System and its life essence: the AJAX.

In: Proceedings of the 8th International Conference on Applied Informatics (ICAI 2010) vol. 2.

Ed.: by Attila Egri-Nagy, Emőd Kovács, Gergely Kovásznai, Gábor Kusper, Tibor Tómacs, Eszterházy Károly Főiskola, Eger, 351-357, 2011. ISBN: 9789639894723

DOI: <http://dx.doi.org/10.13140/2.1.1367.9362>





DEBRECENI EGYETEM
EGYETEMI ÉS NEMZETI KÖNYVTÁR
PUBLIKÁCIÓK



10. **Nagy, Z.:** Intelligent Web Systems.

In: A Nemzetközi III. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia Kiadványa. Szerk.:
Vincze Krisztián, Szent Atanáz Görög Katolikus Hittudományi Főiskola, Nyíregyháza, 50-54,
2010. ISBN: 9789638780966

A DEENK a Jelölt által az iDEa Tudóstérbe feltöltött adatok bibliográfiai és tudományometriai ellenőrzését a tudományos adatbázisok és a Journal Citation Reports Impact Factor lista alapján elvégezte.

Debrecen, 2014.11.17.





Register number: DEENKÉTK/383/2014.
Item number:
Subject: Ph.D. List of Publications

Candidate: Zsolt Nagy
Neptun ID: PPFXF8
Doctoral School: Doctoral School of Informatics

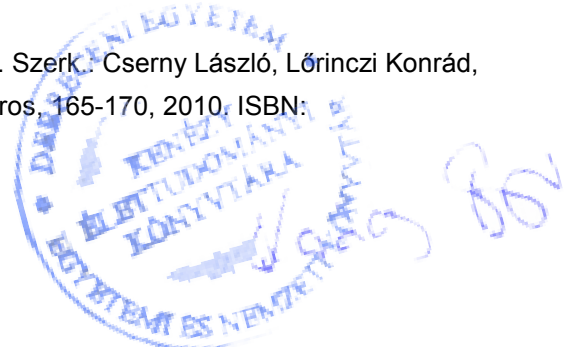
List of publications related to the dissertation

Foreign language scientific article(s) in international journal(s) (2)

1. **Nagy, Z.:** Social media risks from forensic point of view.
Int. J. Computers Communications. 6 (4), 245-253, 2012. ISSN: 2074-1294.
2. **Nagy, Z.:** Ajax 2 in 1: Interactive education and modern web technology.
JAMPAPER . 5 (1), 39-44, 2010. EISSN: 1789-6967.

Hungarian conference proceeding(s) (2)

3. **Nagy Z.:** Turisztikai portálok szolgáltatásai most és a jövőben.
In: V. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia Kiadványa, CD [elektronikus dokumentum]. Szerk.: Kerekes Benedek, Gát György, Nyíregyházi Főiskola, Nyíregyháza, 98-102, 2012. ISBN: 97896399099
4. **Nagy Z.:** Web 2.0: Az Intelligens Web.
In: Informatika Korszerű Technikai Konferencia 2010. Szerk.: Cserny László, Lőrinczi Konrád, Dunaújvárosi Főiskola Informatikai Intézet, Dunaújváros, 165-170, 2010. ISBN: 9789639915381





Foreign language conference proceeding(s) (6)

5. **Nagy, Z.:** Adaptive Design Process for Responsive Web Development.

In: Advances in Information Science and Applications : Proceedings of the 18th International Conference on Computers (part of CSCC '14). [s.n.], [s.l.], 410-414, 2014. ISBN:

9781618042378

DOI: <http://dx.doi.org/10.13140/2.1.3354.5601>

6. **Nagy, Z.:** Improved Speed on Intelligent Web Sites.

In: Recent Advances in Computer Science: Proceedings of the 17th International Conference on Computers. Ed.: Ognian Nakov, et al, WSEAS Press, [Greece], 215-220, 2013. ISBN:

9789604743117

7. **Nagy, Z.:** Using Forensic Techniques for Internet Activity Reconstruction.

In: Recent Researches in Communications and Computers : Proceedings of the 16th WSEAS International Conference on Computers (part of CSCC '12). Ed.: Sandra Sendra, José Carlos Metrôlho, World Scientific and Engineering Society Press, [Greece], 248-253, 2012. ISBN:

9781618041098

8. **Nagy, Z.:** AJAX-Based Data Collection Method for Recommender Systems.

In: Recent researches in communications and computers : proceedings of the 16th WSEAS international conference on communications (part of CSCC '12). Ed.: Sandra Sendra, José Carlos Metrôlho, WSEAS Press, [Greece], 446-451, [2012]. ISBN: 9781618041098

9. **Nagy, Z.:** Intelligent Web System and its life essence: the AJAX.

In: Proceedings of the 8th International Conference on Applied Informatics (ICAI 2010) vol. 2.

Ed.: by Attila Egri-Nagy, Emőd Kovács, Gergely Kovásznai, Gábor Kusper, Tibor Tómacs, Eszterházy Károly Főiskola, Eger, 351-357, 2011. ISBN: 9789639894723

DOI: <http://dx.doi.org/10.13140/2.1.1367.9362>





UNIVERSITY OF DEBRECEN
UNIVERSITY AND NATIONAL LIBRARY
PUBLICATIONS



10. **Nagy, Z.:** Intelligent Web Systems.

In: A Nemzetközi III. Nyíregyházi Doktorandusz (PhD/DLA) Konferencia Kiadványa. Szerk.:
Vincze Krisztián, Szent Atanáz Görög Katolikus Hittudományi Főiskola, Nyíregyháza, 50-54,
2010. ISBN: 9789638780966

The Candidate's publication data submitted to the iDEa Tudóstér have been validated by DEENK on the basis of Web of Science, Scopus and Journal Citation Report (Impact Factor) databases.

17 November, 2014

