

SINGLE-PAGE APPLICATIONS: TREND ILI BUDUĆNOST SINGLE-PAGE APPLICATIONS: TREND OR FUTURE

Slavimir Vesić, Miroslav Minović

REZIME: Nekoliko godina unazad na Web stranicama dizajnera i ljudi koji se bave razvojem aplikacija na Webu možemo uočiti trend upotrebe naizgled nove vrste Web aplikacija. Taj trend se produžava i nastavlja na stručne konferencije, predavanja i korisničke grupe u Srbiji, gde se govori o Web aplikacijama, Single-Page Applications koje se u odnosu na svoju interaktivnost veoma slične desktop aplikacijama. Single-Page Applications su Web aplikacije koje se mogu posmatrati kao napredak u bogatim internet aplikacijama zasnovanih na Ajax tehnologiji. Iako Ajax tehnologija nije nova tehnologija, veoma mali broj istraživanja i naučnih radova se bavi njenim uticajem na usability aspekte Web aplikacija. Usability aspekti Web aplikacija se često previđaju od strane programera i ljudi koji razvijaju Web, jer su oni više fokusirani na to šta novom tehnologijom može da se uradi. Zbog toga, često se dešava da određene Web aplikacije, iako su funkcionalno odlične, jednostavno se ne koriste, jer korisnici odlaze sa Web stranice. U osnovi opisanog problema su lakoća učenja, efikasnost, memorisanje, greške i zadovoljstvo kao usability komponente konkretne Web aplikacije. Osnova ideja ovog rada je da pokuša da sagleda usability aspekte Single-Page aplikacija, izanalizira ih i da predloži određene principe koji se mogu koristiti kao primer najbolje prakse.

KLJUČNE REČI: Single-Page Applications, usability, Ajax, Bogate Internet Aplikacije, Interakcija čovek-računar

ABSTRACT: Few years ago on a designers Web pages and people who develop applications on the Web, we can notice trend of new kind of Web applications. This trend expands and continues to professional conferences, lectures and user groups in Serbia, where we talk about Web applications, Single-Page Applications that are in proportion to its interactivity very similar to desktop applications. Single-Page Applications are Web applications that can be seen as progress in rich Internet applications based on Ajax technology. Although Ajax technology is not new technology, very few studies and scientific papers dealing with its impact on the usability aspects of a Web applications. Usability aspects of Web applications is often overlooked by programmers and people who develop the Web, because they are more focused on what the new technology can be done. Therefore, it often happens that certain Web applications, although functionally excellent, just do not use, because users leave the Web site. The basis of the aforementioned problems are learnability, efficiency, memorability, errors and satisfaction as usability components specified Web application. The basic idea of this paper is to try to examine the usability aspects of the Single-Page Application, to analyze them and to suggest certain principles that can be used as an example of best practice.

KEY WORDS: Single-Page Applications, usability, Ajax, Rich Internet Applications, HCI

1. UVOD

Poslednjih nekoliko godina na stručnim konferencijama, predavanjima i korisničkim grupama u Srbiji [1][2][3][4][5] se govori o relativno novom pristupu u programiranju Web aplikacija, Single-Page Applications, skraćeno SPA. SPA se na trenutak mogu učiniti kao potpuno novi trend Web aplikacija, ali SPA su zapravo nastavak i napredak u Web aplikacijama koje označavamo kao bogate Internet aplikacije (eng. Rich Internet Applications), skraćeno RIA. RIA u stručnim krugovima [6][7] se spominju još 2002. i 2003. godine, a postale su veoma popularne i u naučnim krugovima tokom 2005. godine i koncepta Web 2.0.

World Wide Web, skraćeno WWW, a opšteprihvaćeno Web nije isto što i Internet, najveća globalna mreža, već je njen najvažniji deo. Web se može definisati kao tehnološko socijalni sistem za interakciju između ljudi zasnovan na tehnologiji mreža. Kada se spomene tehnološko socijalni sistem misli se na sistem koji unapređuje ljudsku kogniciju, komunikaciju i kooperaciju [8]. Web 1.0 je prva generacija Weba i on je uglavnom bio samo za čitanje i označava se kao sistem kognicije. U to vreme preduzeća su obezbeđivala kataloge ili brošure da prikažu svoje proizvode i usluge. Korisnici su čitali brošure i

ukoliko su bili zainteresovani, oni su kontaktirali preduzeća da poruče proizvode i usluge. Katalogi i brošure su imali istu funkciju kao oglasi u novinama i magazinima i većina websajtova za elektronsku trgovinu su imali aplikaciju sa korpom u različitim formama i oblicima. Web 1.0 je statičan i ima jedan pravac u komunikaciji, u smeru od preduzeća ka korisnicima, zapravo može se reći da je Web 1.0 monodirekcionni. Sajtovi su sadržali statički HTML, koji se povremeno ažurirao. Osnovna svrha sajtova je da ostvare pristup na Webu. Njihova osnovna karakteristika je da nisu bili interaktivni.

Web 2.0 koncept su popularizovali Dale Dougherty i Tim O'Reilly, krajem 2004. godine na konferenciji O'Reilly Media and MediaLive International. Osnovna ideja Web 2.0 koncepta jeste njegovo razlikovanje od Web 1.0 koncepta koje se ogleda i u tome da Web 2.0 omogućava da se Internet posmatra kao platforma [9][10]. U literaturi [8] Web 2.0 se posmatra kao Web komunikacije. Za razliku od svog prethodnika Web 2.0 omogućava i čitanje i pisanje i može se posmatrati kao bidirekcionni, jer se komunikacija odvija u smeru od korisnika ka preduzećima i obrnuto. Pored toga, Web 2.0 omogućava mnogo veću interaktivnost. U **tabeli 1.** je prikazana osnova razlika Web 1.0 i Web 2.0 koncepta [8].

Web 1.0	Web 2.0
Reading	Reading/Writing
Companies	Communities
Client-Server	Peer to Peer
HTML Portals	XML, RSS
Taxonomy	Tags
Owning	Sharing
IPOs	Trade sales
Netscape	Google
Web forms	Web applications
Screen scraping	APIs
Dialup	Broadband
Hardware costs	Banwidth costs
Lectures	Conversation
Advertising	Word of mouth
Services sold over the web	Web services
Information portals	Platforms

Tabela 1. razlike između Web 1.0 i Web 2.0 koncepta

Sa popularizacijom trenda Web 2.0, stručna i naučna javnost je polako krenula da analizira koje su to pogodnosti koje treba da donesu RIA i jedna od najznačajnijih jeste da obogate i unaprede korisničko iskustvo na način koji je blizak Desktop aplikacijama. U odnosu na Web 1.0, Desktop aplikacije su interaktivnije i ideja RIA jeste da promeni interakciju korisnika i sajta od modela sa stranice na stranicu (eng. page-to-page) u neki bolji model. Ono što se misli pod interaktivnijim modelom se odnosi na razmenu podatka između pretraživača i servera na asinhroni način, tako da klijent, tj. pretraživač (eng. browser) ostane budan i odgovara na operacije korisnika (eng. responsive) dok se neprekidno vrši preračunavanje ili ažuriranje delova korisničkog interfejsa. Karakteristike interaktivnijeg modela se odnose i na uklanjanje nepotrebnog ponovnog učitavanja stranice (eng. page reload), obezbeđivanje opcije prevuci-i-ispusti (eng. drag & drop), kratko vreme odgovora na zahtev korisnika i multimedijalne animacije [11]. Zasnovano na navedenim karakteristikama, ono što je moguće ponuditi korisnicima RIA aplikacija su provera podataka uživo (eng. live validation), automatsko kompletiranje (eng. auto completion), periodično osvežavanje i bogati tekstualni editori. Ova pojava se može sažeti u viziju „Dokument je interfejs!“ [12].

RIA aplikacije se mogu grupisati u 3 kategorije prema tehnologijama u kojima su razvijane:

1. one koje su bazirane na Ajax tehnologiji
2. one koje se oslanjaju na Plug-In dodatke u pretraživačima, npr. Flash, Flex i Silverlight
3. one koje se bazirane na nekom run-time okruženju, npr. JavaFX

Poslednjih nekoliko godina udeo druge i treće kategorije navedenih RIA aplikacija na Webu je značajno opao [13]. Prema istraživanjima o udelu tehnologija koje koriste RIA aplikacije, aplikacije zasnovane na Ajax tehnologijama su najpopularnije i uzimaju najveći deo tržišta, do 88%. Ovaj trend se može objasniti napretkom u tehnološkoj dimenziji kao i napretkom u usability aspektu istih.

Napredak u tehnološkoj dimenziji se manifestuje u:

- mnogo bržem pristupu Internetu
- HTML5 i CSS3 standardizaciji
- boljim pretraživačima (eng. browsers) i jačim JavaScript engine-ima
- programerima sa mnogo više iskustva u programiranju Web aplikacija

SPA aplikacije se u literaturi spominju u radovima Mesbaha [14][15] i u disertaciji [16] istog autora, pri čemu se koristi i naziv Single Page Interface da opiše model interakcije SPA aplikacije. SPA aplikacije, kao nastavak RIA aplikacija pod uticajem Ajax tehnologije su zapravo Web aplikacije koje staju na jednu Web stranicu i imaju cilj da obezbede fluidnije, tj. neprekidnije korisničko iskustvo u odnosu na Web 1.0 aplikacije. Njihova osnovna karakteristika jeste da kompletan potreban HTML, CSS i JavaScript kod za izvršavanje se dopremi do klijenta, tj. pretraživača odjednom. Kada korisnik učini odgovarajuću akciju u pretraživaču i time pozove neku od funkcionalnosti poslovne logike, izvrši se tzv. parcijalno ažuriranje Web stranice, a ostali delovi ostaju netaknuti. SPA je sastavljena odjedne stranice, koja se može ažurirati nezavisno na svaku korisničku akciju, tako da nije potrebno da se kompletna stranica ponovo učita kao kod klasičnih Web aplikacija. Ukratko SPA je Websajt u kojem se stranica ne učitava ponovo (eng. reload) [17] i direktno utiče da se uveća nivo interaktivnosti, responzivnosti i zadovoljstva korisnika [14]. Arhitekturno gledano, dolazi do pomeranja od tzv. arhitekture tankog klijenta (eng. thin-client architecture) ka arhitekturi debelog klijenta (eng. fat-client architecture), što zapravo znači da se procesiranje poslovne logike aplikacija pomera sa servera na klijent. Dakle, dolazi do pomeranja sa Model-View-Controller, skraćeno MVC, uzora ka Model-View-ViewModel uzoru, skraćeno MVVM u projektovanju Web aplikacija. Dizajn je neizbežni deo svake aplikacije, pa tako i Web aplikacije. Sve je veći broj Web dizajnera i front-end developer-a, koji u ličnim Web stranicama, kao portfolio koriste single-page web design princip [18][19][20][21][22][23][24][25][26].

Može se postaviti pitanje zašto još na samom početku spominjanja Web 2.0 koncepta, dakle od 2005. godine, nije se izvršilo pozicioniranje na izradu Web aplikacija u Ajax tehnologiji, a time i SPA aplikacija? Odgovor na ovo pitanje se takođe može tumačiti kao nedostatak odgovarajući tehnoloških uslova: pristup Internetu je bio sporiji, JavaScript engine-i nisu bili dobri kao danas, tada je postojao HTML4 koji se različito ponašao u različitim pretraživačima i time davao različiti Document Object Model, skraćeno DOM. Pored toga, nisu postojali RESTful servisi, kao i JSON format koji je došao kao rezultat mobilnih aplikacija. Programeri su mnogo više bili fokusirani na Flash i ostale tehnologije.

Ukoliko danas posmatramo krajnje korisnike, možemo zaključiti da su njihovi zahtevi i očekivanja dosta veći nego što su bili 2005. godine. Na taj način dolazimo do drugog važnog aspekta, a to je usability. Usability je kvalitativan atribut koji procenjuje kako jednostavno se korisnički interfejs koristi [27]. Prema ISO 9241-11 standardu usability pokazuje u kojoj meri se proizvod može koristiti od strane specifičnih korisnika tako da se ostvare odgovarajući ciljevi efektivno, efikasno i zadovoljavajuće u specifičnom kontekstu upotrebe [28].

Prilikom projektovanja i programiranja aplikacije, programeri i dizajneri često zaboravljaju usability aspekte aplikacija i

fokusirani su na to šta određena nova tehnologija može da ponudi, što često vodi do odgovora zašto se neki Web sajt koristi ili ne. Ono što nije dovoljno istraženo u literaturi jeste kako SPA aplikacije utiču na usability koncepte i to pitanje je ujedno i centralna tema ovog rada. U drugom delu rada prikazaćemo radove iz prethodnih istraživanja, u trećoj glavi opisaćemo uticaj Ajax tehnologije na usability, zatim uticaj SPA na usability, kao i neke važne dizajn karakteristike Single Page Web dizajna. Na kraju ćemo dati zaključak i moguće pravce budućih istraživanja.

2. SRODNA ISTRAŽIVANJA

U naučnoj literaturi tokom godina je objavljen veliki broj radova na temu RIA, uporedo sa tadašnjim trendom Web 2.0. U radu [29] primenjeno je sistemsko mapiranje i klasifikacija literature o RIA u prethodnih 10 godina. Radom je obuhvaćeno 133 članaka. Klasifikacija članaka je sprovedena prema nekoliko kriterijuma, a prema jednom kriterijumu postoji kategorija usability i accessibility. Broj članaka u kojima se navodi usability i accessibility je 25. I ako većina članaka navodi usability kao jedan od važnih aspekata RIA aplikacija, zapravo veoma mali broj istraživanja se bavi istim, mnogo veći broj radova koji su zastupljeni odnose se na accessibility i standard WAI-ARIA [30]. Začudujuće je da je broj članaka koji obrađuju usability, veoma mali i iznosi 3, a ni jedan konkretno ne obrađuje RIA aplikacije zasnovane na Ajax tehnologiji.

U članku [31], Rosenberg opisuje šta je potrebno razmotriti prilikom primene Ajax tehnologije na primeru Yahoo Web maila. U istraživanju [32] je sprovedena analiza između Ajax i drugih koncepata putem ankete na primeru Web sajta za planiranje putovanja. Postavljene su i testirane su 4 hipoteze:

- H1: Zadovoljstvo korisnika sa upotrebom polja za unos (eng. input fields) sa funkcijom automatskog kompletiranja (eng. auto-completion) je veće nego bez.
- H2: Efikasnost Web aplikacije upotrebom polja za unos (eng. input fields) sa funkcijom automatskog kompletiranja (eng. auto-completion) je veće nego bez.
- H3: Zadovoljstvo korisnika sa tablom za poruke za prikaz novih komentara putem Ajax tehnologije je veće nego obično prenošenje komentara.
- H4: Efikasnost Web aplikacije sa tablom za poruke za prikaz novih komentara putem Ajax tehnologije je veće nego obično prenošenje komentara.

U radu [33], Pilgrim opisuje na koje usability koncepte je potrebno obratiti pažnju prilikom izrade Web 2.0 aplikacija i kako Web 2.0 aplikacije menjaju način na koji korisnik ima interakciju sa Web-om. Ovaj rad ukazuje na određene probleme koji se mogu javiti upotrebom Ajax tehnologije, od kojih je jedan svakako kognitivni zamor (eng. cognitive overload) ukoliko se ne koriste adekvatno. Holzinger i ostali autori [34], diskutuju o prednostima i nedostacima upotrebe Ajax tehnologije sa usability stanovišta. Analizira se kako Ajax tehnologija utiče na usability dimenzije. Pilgrim [35] istražuje usability aspekte sajtova za elektronsku trgovinu koji koriste Ajax tehnologiju. Uočavaju se dva glavna problema u radusa Ajax aplikacijama, a tiču se direktno usability-a, a to su upotreba Back dugmeta u pretraživačima i menadžment ažuriranja Web stranica (eng. update management). Sachdeva i Singh [36] opisuju kako je Ajax

promenio Web putem parcijalnog ažuriranja stranice i koji su efekti toga na usability aspekte. U istraživanju Hara i Kashimura [37], uočavaju usability probleme za grupu starijih punoletnih lica i predlažu određena rešenja u dizajnu koja mogu umanjiti te probleme.

3. ANALIZA USABILITY ASPEKTA U SPA

3.1. Ajax i usability

Sa Web 2.0 konceptom se promenio način na koji krajnji korisnici pristupaju i koriste sadržaj, kao i način komunikacije između samih korisnika putem socijalnih mreža kao što su npr. Facebook, Twitter, itd. Korisnici su promenili svoj pogled na Web od naprave za pregledanje sadržaja, do pogleda gde Web postaje platforma na kojoj se izvršavaju aplikacije i servisi [38]. Ova promena paradigme menja jedan od osnovnih principa arhitekture WWW-a u kome je osnovna jedinica stranica (eng. page) [39]. Sa jedne strane, svet u kojem živimo napreduje od evolucije u veličini i upotrebi Weba, dok sa druge strane žurba da se primene sve inovacije ima za posledicu da mnogo programera previdi dobro utvrđene principe dobrog dizajna i usability [40].

Nielsen [41] je 1995, predložio 10 heuristika za korisnički interfejs, kao što su npr. vidljivost sistemskog statusa, veza između sistema i realnog sveta, konzistentnost i standardi itd. Zasnovano na radovima Nilsena, Neil [42] je prilagodila heuristike i prikazala primenu heuristika na RIA aplikacijama razvijenih u tehnologijama Flash i Sliverlight. Usability je jedan od najvažnijih nefunkcionalnih softverskih zahteva i sastoji se iz određenih komponenti. Prema Nielsenu [27], to su:

1. *Lakoća učenja* (eng. *learnability*) – Koliko jednostavno korisnici izvršavaju jednostavne zadatke kada se prvi put susretnu sa dizajnom?
2. *Efikasnost* (eng. *efficiency*) – Jednom kada korisnik je naučio dizajn, koliko brzo može da izvede zadatke?
3. *Memorisanje* (eng. *memorability*) – Kada se korisnik vrati dizajnu nakon nekog perioda u kojem ga nije koristio, koliko jednostavno može da vrati svoju profesionalnost?
4. *Greške* (eng. *errors*) – Koliko grešaka korisnik pravi, koliko su ove greške ozbiljne i koliko jednostavno se oporavljaju od ovih grešaka?
5. *Zadovoljstvo* (eng. *satisfaction*) – Koliko je ugodno kada se dizajn koristi?

Prema istraživanju Pilgrima [35], konzistentnost, lakoća učenja i povratne informacije su relevantne kategorije, kada se procenjuje usability komercijalnih Web aplikacija, zasnovanih na Ajax tehnologiji. Kognitivni psiholozi objašnjavaju da prilikom interakcije sa svetom naš um kreira mentalne modele o tome kako stvari funkcionišu [43]. Prema Normanu [44], interfejsi koji zadovoljavaju mentalni model korisnika će omogućiti da korisnik jednostavno razume sistem i uspostavi potrebnu interakciju. Pilgrim [35] objašnjava da možemo očekivati da klasična interakcija sa stranice na stranicu (eng. page-to-page) bude ukorenjena u korisnikov mentalni model Weba, obzirom na dužinu upotrebe iste.

Prema Shneidermanu i Plaisantu [45], osnovni princip interakcije čovek-računar, jeste da interfejsi budu jednostavni za upotrebu i predvidivi. Korisnici na Webu su opšte poznato nes-

trpljivi, željni trenutnog zadovoljenja i napustiće Web sajt ukoliko odmah ne shvate kako treba da ga koriste [41]. Prema Normanu, interfejs treba da obezbedi jasno koje su akcije dostupne u trenutku, rezultat akcija i trenutno stanje sistema tako da korisnik zna šta treba da uradi instiktivno. Nedostatak konzistentnosti, lakoće učenja i povratnih informacija može proizvesti dezorijentaciju i kognitivni zamor u korisnicima Web aplikacija.

Prema Conklinu [46], kognitivni zamor nastaje tako što se ulaže dodatni napor i koncentracija potrebna za izvršenje nekoliko zadataka u uređenom redosledu. Kognitivni zamor je jedna od najvažnijih stvari koju ljudi koji razvijaju Web aplikacije treba da zaobiđu. Ima slučajeva kada su dizajn i način upotrebe tehnologije presudni za uklanjanje kognitivnog zamora, što se može videti u radu [37]. Hara i Kashimura eksperimentalno utvrđuju usability 3 različita dizajna Web aplikacije na starosnoj grupi ljudi između 60 i 70 godina. Kognitivni zamor je jedan od osnovnih problema koji se može javiti kada grupa starijih građana krene da koristi aplikacije zasnovane na Ajax tehnologiji. Prema njihovom istraživanju, dizajniranje Web aplikacija za starije korisnike ima 3 osnovna problema koji se javljaju u radu sa Ajax tehnologijama:

- nesposobnost da se primeti rezultat koji je prikazan
- nesposobnost da se podesi parametar za pretragu preko upotrebe tabova
- nesposobnost da se razume koji parametar pretrage je trenutno selektovan

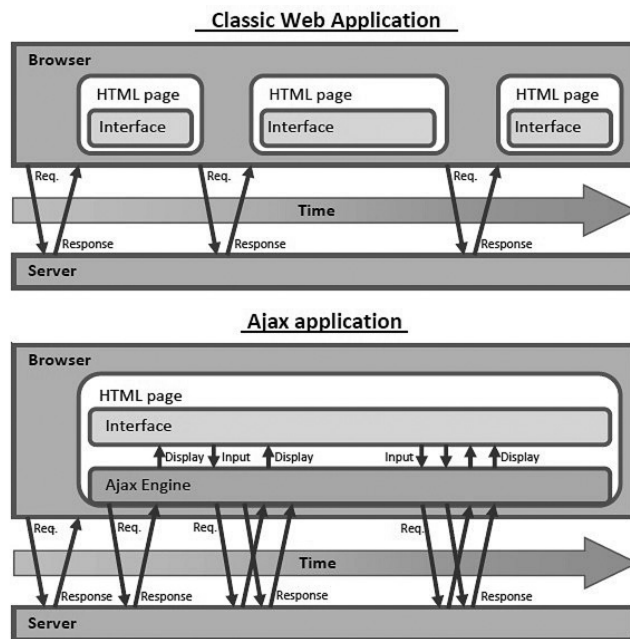
Ovi problemi se prevazilaze tako što se za grupu starijih korisnika primenjuju sledeće strategije:

- a) povećava se perceptualna isturenost elemenata grafičkog interfesja ili oblasti gde se taj element učitava
- b) ističe se status trenutnog podešavanja
- c) izbegavaju se tabovi, a koriste se liste da omoguće korisnicima da vide sve kategorije parametara odjednom

Prema Nielsenu [41], jednostavnost originalnog modela interakcije korisnika sa Web aplikacijom zasnovanog na stranicama doprinosi da se Web jednostavno koristi i upravno zbog toga porast upotrebe Web-a je velika. Isti autor [40], ističe da veliki broj Web sajtova koji koriste Ajax zanemaruju osnovne principe dobrog dizajna i usability-a.

3.2. SPA i usability

Na svojem početku Ajax tehnologija se upotrebljavala kao izvesna dopuna klasičnog modela interakcije korisnika i Web aplikacije. Kao rezultat masovne popularizacije pretraživača Google-a, veliki broj Web sajtova je krenuo da koristi Ajax funkcionalnosti kao što su automatska sugestija (eng. auto suggest) i automatsko kompletiranje (eng. auto completion). Iako je primarni model interakcije tih Web aplikacija bio sa stranice na stranicu (eng. page-to-page), neke od funkcionalnosti su uključivale Ajax i njegove tada najpopularnije funkcije. Ajax obzirom da obavlja asinhronu komunikaciju sa Web serverom, **slika 1.**, potpuno drugačije uspostavlja interakciju korisnika sa sadržajem Web stranice i to može da zbuni korisnika u potpunosti i da prouzrokuje kognitivni zamor. To se dešava jer korisnik ne zna šta da očekuje ponašanje određenog elementa HTML stranice usled kombinacije različitih modela interakcije i ne može da razume šta sistem zapravo radi.



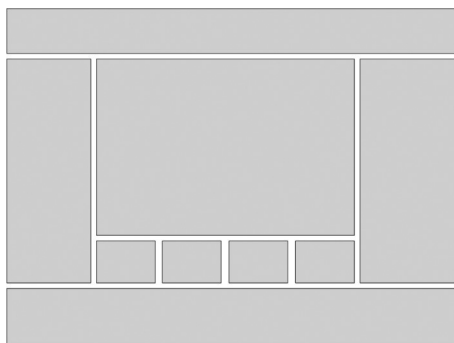
slika 1. Poređenje klasičnih Web aplikacija i Ajax aplikacija[32]

Sa druge strane SPA, koje koriste Ajax tehnologiju u potpunosti kao jedini način razmene podataka između klijenta i servera, primenjuju potpuno drugačiji model interakcije i kao što Mesbah [47] naglašava „razbija se metafora Web stranice“.

Jedna od osnovnih prednosti SPA aplikacija jeste njihova brzina. Prema vodećem ekspertu za usability pitanja Nielsen-u [48], postoje 3 važne granice vremena odgovora nekog sistema:

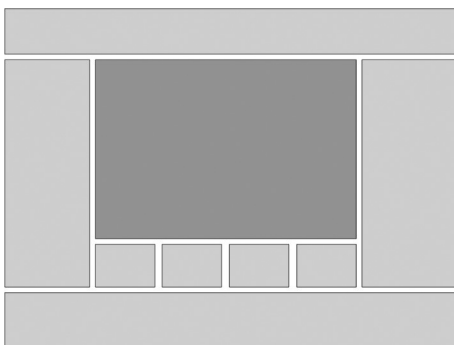
- 0.1 sekunda – je granica u kojoj korisnik oseća da sistem reaguje trenutno, što znači da nije potrebna specijalna povratna informacija osim prikaza samog rezultata
- 1 sekunda – je granica kada korisnik primećuje malo kašnjenje u odgovoru sistema na njegov ulaz i počinje svoj tok misli. Zapravo to je granica kada korisnik se odvaja od njegovog iskustva [17] i osećaja da radi direktno nad podacima.
- 10 sekundi – je krajnja granica za zadržavanje korisnikove pažnje. Nakon toga, on kreće da radi druge aktivnosti. Na primeru Web aplikacije on jednostavno odlazi na drugi sajt.

Neke studije pokazuju da čak i najmanje poboljšanja u kašnjenju mogu da imaju merljiv uticaj na korišćenje aplikacije [49]. Iz tog razloga, Web sajtovi za cilj imaju da poboljšaju stvarno kašnjenje, kao i ono koje percipira sam korisnik. Tradicionalne Web aplikacije funkcionišu po principu zahtev-odgovor (eng. request-response). Kada krajnji korisnik uputi zahtev za određenom Web stranicom, Web server prosleđuje tu stranicu pretraživaču. Pretraživač kreće da parsira tu stranicu i ukoliko postoje dodatni sadržaji kao što su CSS (eng. Cascading-Style-Sheet), slike, video i dodatne skripte, pretraživač pravi ponovni zahtev ka Web serveru i zahteva resurse. Razmotrimo scenario kada krajnji korisnik razgleda slike na Web sajtu i prikaz same slike se nalazi u centralnom delu Web stranice, a ispod centralne slike su prikazane umanjene slike (eng. thumbnail), kao na **slici 2.**



Slika 2. skica Web 1.0 galerije slika [17]

Kada korisnik pritisne thumbnail, kompletna Web stranica će se ponovo učitati sa svim ostalim elementima: header, footer i sekcije sa leve i desne strance. Kada korisnik nakon toga, ponovo zatraži neke podatke ili ode na drugu stranicu isti postupak - roundtrip, se ponavlja. Kod SPA aplikacija prvi put se učitaju svi fajlovi u pretraživač, gotovo isto kao i kod tradicionalnih Web aplikacija, mada praksa je pokazala da je to ipak malo duže, ali nakon toga kada korisnik uputi zahtev za novim sadržajem, samo taj element stranice se osvežava, dok ostatak stranice nije promenjen, *slika 3*.



Slika 3. skica SPA galerije slika [17]

Na taj način se štedi vreme koje je potrebno da se iskoristi za ponovno učitanje kompletne stranice (eng. reload). Iako tehnike keširanja mogu u mnogome ubrzati stvari, postoje situacije u kojima nije to u potpunosti primenjivo. Sa druge strane, prema istraživanju koje navodi Powell [17] najveći deo vremena, do 80% se troši na učitanje Web stranice i njen prikaz u pretraživaču, a samo 20% na odgovor od Web servera. S toga kada se eliminiše veličina fajlova koje pretraživač treba da renderuje i prikaže korisniku, smanjiće se i vreme učitanja stranice. SPA osvežava samo deo stranice u kojem je nastupila promena i time značajno ubrzava odgovor ka krajnjem korisniku i na taj način korisnik ima utisak da sistem reaguje trenutno. Veličina HTML stringova koja se razmenjuje na relaciji klijent-server je značajno manja, kao proizvod ekstenzivne upotrebe Ajax tehnologije.

Kao posledica veće brzine odgovora krajnjem korisniku i parcijalnog ažuriranja dela Web stranice, ostvaruje se veća *interaktivnost i fluidnost* [32] same Web aplikacije. Prema studiji [52], u kojoj se poredi nekoliko aplikacija sa različitim stepenima interaktivnosti, one aplikacije koje su interaktivnije direktno utiču na krajnje zadovoljstvo korisnika.

SPA aplikacije takođe utiču na *efikasnost*, kao aspekt usability-a. Usled toga što se primenjuju Ajax funkcionalnosti, kao što su npr. automatsko kompletiranje i provera podatka uživo, korisnici mogu da izvedu svoje zadatke mnogo brže. Kod tradicionalnih Web aplikacija, korisnik šalje podatke na Web server da bi se obavila validacija podatka u bazi i ukoliko nešto nije ispravno stranica se ponovo učitava i greška se prikazuje krajnjem korisniku i time se ponovo razmenjuju ogromne količine HTML stringova između klijenta i servera. Kod SPA mogućnost takve greške se smanjuje, jer se korisnik usmerava da unese validnu vrednost iz baze podatka. Automatskim kompletiranjem se postiže da se duže liste vrednosti ili padajuće liste efikasnije koriste, tako što će korisnik npr. unositi početna slova željenog naziva grada. Studija [32], upravo to i pokazuje.

U poslednje vreme na velikom broju portfolio Web stranica je prisutan novi trend u dizajniranju Web stranica, Single-Page Web Design [18][19][20][21][22][23][24][25][26]. Neke od prednosti ovog pristupa u izradi Web interfesja sa aspekta usability su [50][51]:

- *intuitivnost* – sve što je potrebno krajnji korisnik da zna jeste da skroluje, moguće je uključiti strelice ili druge navigacione oznake. Nije potrebno brinuti o tome da li će se korisnici izgubiti u navigaciji, beskrajno tražeći ono što im je potrebno.
- mogućnost *prepričavanja priče* (eng. storytelling) – single page Web sajtovi koriste mogućnost prepričavanje priče, pri čemu sajtovi sa više strana nisu toliko dobri u tome. Ovo može značajno da inspiriše korisnike da preduzmu akciju. Ljudi su naučili da prate priču, bila ona online[18] ili ne, tako da je ovo jasan benefit za korisničko iskustvo (eng. user experience). Veliki deo ovog iskustva je ukorenjen u ljudima dok su još bili deca, tako da SPA dizajn je nešto potpuno prirodno za korisnike.
- *lakši su organizovani* – ne postoji više ogromna lista stranica ili podstranica koje je potrebno organizovati. Nema više zamaranja oko toga da li neka stranica treba da bude otac ili dete u nekoj hijerarhiji. Nema više ogromne navigacije menija ili pod-menija. I kada navigacija postoji veoma je diskretna, makar za same portfolio sajtove.
- *fokusiranost na sadržaj* – Single-Page Web Design, omogućava da korisnik bude fokusiran samo na jedan deo sadržaja koji mu se servira kroz pozadinsku sliku i tekst manjeg obima, time se definitivno otklanja i najmanja mogućnost da dođe do kognitivnog zamora. Nakon što korisnik završi sa čitanjem te stranice, putem skrolovanja ili strelica na tastaturi on prelazi na sledeći sadržaj i tako za svaku narednu stranicu. Većina ovako organizovanih Web sajtova ima različite pozadine pa se jasno vidi razgraničenje između 2 celine. Tehnika parallax scrolling [53] je jedna od onih koja pruža ovaj efekat. Ova tehnika zaravo omogućava da se pozadinska slika kreće brže od slike u prvom planu i time se postiže efekat dubine i sjedinjavanja.

Pored toga, što smo se osvrnuli na dizajn single-page-a, postoji potreba da se ukaže na jedan vizuelni element, a to je *chrome*, koji je prisutan kod najvećih Web mail aplikacija, kao i socijalnih mreža. Kao što je Nielsen [54], objasnio chrome se ne odnosi na popularni pretraživač kompanije Google, već na vizuelni element koji se može naći u Web, ali i u ostalim aplikacijama, koji okružuje podatke i sadržaj Web stranice. On

obično služi da korisnicima pruži informacije o komandama, putem kojih se pokreću neke akcije vezane najčešće za sam sadržaj, a najčešće nije deo tog istog sadržaja. U Web aplikacijama chrome obično uključuje: navigacione elemente, footer, logo, polja za pretragu itd. koristi od chrome-a su sledeće:

- chrome obezbeđuje stabilan skup komandi i opcija koje su uvek vidljive, uvek se nalazi na istom mestu i oslobađa korisnike od potrebe da ga traže. Iako on zauzima dosta piksela, veoma je značajan korisnicima, jer oni mogu veoma jednostavno, čak i upotrebom tastature sprovesti određene komande ili prečice.
- chrome nudi skup generičkih komandi koje rade na različitim tipom sadržaja i podacima koji se pojavljuju unutar njegovog okvira. Zbog toga što je uvek isti, korisnici manje moraju da uče.
- chrome promovira konzistentnost i standarde u korisničkom interfejsu, koji olakšava učenje i daje korisnicima osećaj da imaju više kontrole nad njihovim iskustvom.

Ono što je važno sa druge strane za chrome, jeste da se ne pretera sa njegovim dimenzijama. Najpoznatija SPA aplikacija je Gmail, najpoznatiji Web mail. Ukoliko pogledamo Gmail, Yahoo mail i Outlook, možemo videti sve navedene prednosti koje su obezbeđene chrome-om. U sklopu chroma se nalaze komandne opcije koje su dostupne korisniku, u vidu kreiranju novog email-a, odabir sekcija Inbox, Sent, Draft, opcija za pretragu, zatim sekcija za korisnička podešavanja, sa ikonicom u vidu zupčanika – gear, prečice ka ostalim servisima pomenutih platformi. Uočavamo da su ispoštovani dobri principi dizajna i analize koja je proistekla putem studija praćenja oka (eng. eye tracking) da se navigacija nalazi sa leve strane. Veoma sličan primer je sa Web aplikacijom Twitter, kod koje se takođe vidi chrome, a u sklopu njega navigacija sa leve strane. Selekcijom na neku od opcija unutar navigacije, centralni deo u kojem se nalazi sadržaj se ažurira putem Ajax tehnologije, tako da krajnji korisnik ima utisak da se to dešava momentalno, čime se obezbeđuje izražena interaktivnost SPA Web aplikacija, *slika 4*.



Slika 4. skica Web mail SPA aplikacija

Jedna od najozbiljnijih prepreka koja ugrožava usability direktno je tzv. „*pokvareno Back dugme pretraživača*“. Naime, za razliku od klasičnih Web aplikacija kod kojih možemo da se vratimo na Web stranicu pre i da idemo na stranicu posle, kod SPA aplikacije je korisnije govoriti o stanju. Kada se preko Ajax poziva parcijalno osveži deo stranice, URL u pretraživaču ostaje isti i u trenutku kada pritisnemo Back dugme mi zapravo idemo na lokaciju na Webu, na kojoj smo bili pre nego što smo došli na lokaciju na kojoj je postavljena SPA aplikacija.

Opisana situacija krajnje korisnike dovodi u zabunu i svakako je jedan od razloga zašto se neće opredeliti da posećuju Web lokaciju sa SPA aplikacijom, jer se upotrebom Back dugmeta ne postiže željeni efekat. Prema određenim istraživanjima, Back button je jedna od najznačajnijih funkcionalnosti, samog pretraživača i čini otprilike 40% korisnikovih akcija dok pretražuju sadržaj na Webu. Nielsen, u svom članku kritički se osvrće na ovaj nedostatak prouzrokovan Ajax tehnologijom. Pilgrim [35], ukazuje na to kako Google za svoju aplikaciju Google Maps rešava postojeći problem, ali i na to da ne postoji generalno rešenje ovog problema. Preko Back dugmeta se upravlja istorijom URL, tako da su sve promene URL evidentirane u istoriji, a to nije korisno za SPA aplikaciju koja ima promenu stanja. To znači da je potrebno dokodirati funkcije SPA aplikacije, tako da kada korisnik pritisne na Back dugme,

da se vrati u pređašnje stanje. Jedan od načina za rešavanje ovog problema, jeste da se implementira rutiranje URL-a, preko HTML hash (#) elementa i da se implementiraju odgovarajuće JavaScript funkcije, koje bi reagovale na promenu URL-a. Najčešće se postavljaju na onhashchange Event handler.

4. ZAKLJUČAK

SPA svakako imaju efekte na usability i potrebno ih je razmotriti kao opciju prilikom realizacije Web aplikacije. Ono što je potrebno jeste više istraživanja na temu uticaja usability-a na Single-Page Applications. Pored efekta na usability, SPA imaju i još neke prednosti. Npr. one smanjuju opterećenje servera, time što je kompletna poslovna logika prebačena na klijenta, tako da nije potrebno dodatno investirati u hardver, već su korisnici koji se povezuju sa sistemom odgovorni za izgradnju i održavanje svojeg hardvera, bez obzira koji uređaj koriste [55]. To naravno znači da SPA aplikacija je potrebno da imaju responzivan dizajn, tj. takav dizajn koji se prilagođava veličini ekrana. SPA omogućavaju da se samo jedan programski jezik koristi za njihovu realizaciju, npr. JavaScript. Primer je MEAN JavaScript stack, gde se koriste MongoDB, Express.js, Angular JS i Node.js tehnologije.

Tradicionalne Web aplikacije će svakako nastaviti da postoje, ali i SPA aplikacije imaju svoje mesto na Webu pogotovo tamo gde je potrebno uneti veću interaktivnost. Sa novijim verzijama pretraživača, dolaze i bolji JavaScript engine-i, tako da se može očekivati porast SPA trenda. Novi pretraživač kompanije Microsoft Spartan će imati bolji JavaScript engine, čak i od trenutno najbolje na Google Chrome pretraživaču. To nije samo značajno zbog brzine samih pretraživača u renderovanju JavaScript koda već i zbog relativno novog tipa aplikacija koje će se razvijati pod Windows 10 platformom, tzv. UniversalApps. Zapravo je reč o aplikacijama koje se jednom isprogramiraju, a mogu da se koriste na svim uređajima, a JavaScript tu može da ima presudnu ulogu.

LITERATURA

- [1] "Microsoft Forge Community" <http://msforge.net/> (15.09.2015)
- [2] "Tarabica" <http://tarabica.msforge.net/> (15.09.2015)
- [3] "DaFED" <http://dafed.org/> (15.09.2015)
- [4] "Startit" <http://www.startit.rs/fest/> (15.09.2015)
- [5] "MS Sinergija" <https://www.mssinergija.net/sr/sinergija15/Pages/Home.aspx>
- [6] Allaire, Jeremy, "Macromedia Flash MX-A next-generation rich client" (2002) <http://download.macromedia.com/pub/flash/whitepapers/richclient.pdf> (15.09.2015)

- [7] Mullet, Kevin, "The Essence of Effective Rich Internet Applications" (2003) http://download.macromedia.com/pub/solutions/downloads/business/essence_of_ria.pdf(15.09.2015)
- [8] Aghaei, S., Nematbakhsh M., A., Farsani, H., K. "Evolution of the World Wide Web: from Web 1.0 to Web 4.0". International Journal of Web & Semantic Technology (IJWesT) Vol.3, No.1, (2012)
- [9] O'Reilly, T. "Web 2.0: Compact Definition?" (2005) <http://radar.oreilly.com/2005/10/web-20-compact-definition.html>(15.09.2015)
- [10] O'Reilly, T. "Design Patterns and Business Models for the Next Generation of Software" (2005) <http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html>(15.09.2015)
- [11] Busch, M. and Koch, N. "Rich Internet Applications - State-of-the-Art" (2009) http://uwe.pst.fh-lmu.de/publications/maewa_rias_report.pdf (15.09.2015)
- [12] Chen, C. and Raman T. "Ajax: A talking translation bot using google IM: Bringing web-2.0 applications to life". In Proceedings of the 5th International Cross-Disciplinary Conference on Web Accessibility (W4A'08). 54-56. (2008).
- [13] Kosanović, M. i Stošović S. "Bogate Internet Aplikacije pregled stanja 4 godine kasnije" (2015). INFOTEH-JAHORINA Vol. 14, (2015)
- [14] Mesbah, A. and van Deursen, A. "An Architectural Style for Ajax.". Proceedings of the 6th Working IEEE/IFIP Conference on Software Architecture (WICSA'07). IEEE Computer Society. (2007)
- [15] Mesbah, A. and van Deursen, A. "Migrating Multi-page Web Applications to Single-page Ajax Interfaces", Software Maintenance and Reengineering, CSMR '07. 11th European Conference on Date of Conference. (2007)
- [16] Mesbah, A. "Analysis and Testing of Ajax-based Single-page Web Applications. Dissertation" (2009)
- [17] Powell, J. "Three Keys to Successful Single Page Web Applications", HTML5 Developer Conference, San Francisco, CA, April 1-2 (2013) <http://html5devconf.com/archives/april2013/videos.html>(15.09.2015)
- [18] "History of icons" <https://historyoficons.com/>(15.09.2015)
- [19] "New York Summer" <http://johnjacob.ca/newyork/>(15.09.2015)
- [20] "Fraud Force Summit" <http://www.fraudforcesummit.com/>(15.09.2015)
- [21] "Bear on Unicycle" <https://bearonunicycle.com/>(15.09.2015)
- [22] "Wild" <http://wild.as/hell/>(15.09.2015)
- [23] "Franz Sans" <http://www.franzsans.de/>(15.09.2015)
- [24] "Build in Amsterdam" <http://www.buildinamsterdam.com/>(15.09.2015)
- [25] "Braking Badly" <https://citipark.co.uk/citipark/braking-badly/>(15.09.2015)
- [26] "Robby Leonardi" <http://www.rleonardi.com/interactive-resume/>(15.09.2015)
- [27] Nielsen, J. "Usability 101: Introduction to Usability" (2012) <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>(15.09.2015)
- [28] "What is usability?" http://www.usabilitynet.org/management/b_what.htm (15.09.2015)
- [29] Casteleyn, S., Garrigos, I., Mazon J., N. "Ten Years of Rich Internet Applications: A Systematic Mapping Study, and Beyond" ACM Transactions on the Web, Vol. 8, No. 3, (2014)
- [30] W3C. "WAI-ARIA Overview" <http://www.w3.org/WAI/intro/aria> (15.09.2015)
- [31] Rosenberg, G. "A Look Into the Interaction Design of the New Yahoo! Mail ... and the Pros and Cons of AJAX", interactions - 25 years of CHI conferences, Volume 14 Issue 2, pp 33-34 (2007)
- [32] Kluge, J., Kargl, F., Weber, M. "The Effects of the AJAX Technology on Web Application Usability", In: WEBIST 2007 International Conference on Web Information Systems and Technologies, pp. 289-294 (2007)
- [33] Pilgrim, C. "Improving the Usability of Web 2.0 Applications", Proceedings of the 19th ACM conference on Hypertext and hypermedia, USA [doi: 10.1145/1379092.1379144] (2008)
- [34] Holzinger, A., Mayr, S., Slany, W., Debevc, M.: The influence of AJAX on Web Usability. In: ICE-B 2010 - ICETE The International Joint Conference on e-Business and Telecommunications, pp. 124-127. INSTIC IEEE (2010)
- [35] Pilgrim, C. "An Investigation of Usability Issues in AJAX based Web Sites", Conference: Proceedings of the Fourteenth Australasian User Interface Conference - Volume 139. (2013)
- [36] Sachdeva, K., Singh, S. "Use of AJAX to Improve Usability of Online Information Systems," Global Journal of Computer Science and Technology, vol. 13. (2013).
- [37] Hara, Y., Kashimura K. "Cognitive aging and rich internet applications: Usability problems of Ajax based on the empirical study of older adults", Japanese Psychological Research, Special Issue: Cognitive aging, safety, and quality of life, Volume 52, Issue 3, pages 216-226, (2010)
- [38] Dix A., Cowen L. „HCI 2.0?: usability meets web 2.0“, Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it, September 03-07, 2007, University of Lancaster, United Kingdom (2007)
- [39] Berners-Lee, T. "Information Management: A Proposal, World Wide Web Consortium" (1989) www.w3.org/History/1989/proposal.html(15.09.2015)
- [40] Nielsen, J. „Web 2.0 'Neglecting good design'“ (2007) news.bbc.co.uk/1/hi/technology/6653119.stm(15.09.2015)
- [41] Nielsen, J. "10 Usability Heuristics for User Interface Design" (2005) <http://www.nngroup.com/articles/ten-usability-heuristics/>(15.09.2015)
- [42] Neil, T. "Review Usability Best Practices" (2009) <http://designingwebinterfaces.com/6-tips-for-a-great-flex-ux-part-5> (15.09.2015)
- [43] Johnson-Laird, P.N. "Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness". Harvard University Press, Cambridge, MA, (1986)
- [44] Norman, D.A. "The Design of Everyday Things", Doubleday, N.Y. (1988)
- [45] Shneiderman, B. and Plaisant, C. „Designing the User Interface: Strategies for Effective Human-Computer Interaction“, 4th Ed. Addison-Wesley, Reading, MA. (2004)
- [46] Conklin, J. "Hypertext: An Introduction and Survey", IEEE Computer, 20, 9, 17-40. (1987)
- [47] Mesbah, A. and van Deursen, A. "Invariant-based automatic testing of AJAX user interfaces", Proc. IEEE 31st International Conference on Software Engineering, 210-220. (2009)
- [48] Nielsen J. "Response Times: The 3 Important Limits" (1993) <http://www.nngroup.com/articles/response-times-3-important-limits/>(15.09.2015)
- [49] Souders S. "Velocity and the Bottom Line" (2009) <http://radar.oreilly.com/2009/07/velocity-making-your-site-fast.html> (15.09.2015)
- [50] Chapman C. "How to design the perfect single page website"(15.09.2015) <http://www.webdesignerdepot.com/2014/12/how-to-design-the-perfect-single-page-website/>
- [51] Infographic design team. "Benefits of Single Page Infographic Style Websites" <http://www.infographicdesignteam.com/blog/benefits-of-single-page-infographic-style-websites/> (15.09.2015)
- [52] Lowry, P., Madariaga, S., Moffit, K., Moody, G., Spaulding, T., and Wells, T."A theoretical model and empirical results linking website interactivity and usability satisfaction". HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on System Sciences, (2006)
- [53] Stover, K. „Single-page VS Traditional Site: How many pages does your Website need ?“(2014) <http://www.karristover.com/blog/single-page-vs-traditional-site-how-many-pages-does-your-website-need> (15.09.2015)
- [54] Nielsen J. "Browser and GUI Chrome" (2012) <http://www.nngroup.com/articles/browser-and-gui-chrome/>(15.09.2015)
- [55] Powell J., Mikowski M. "Single Page Web Applications- JavaScript End-to-End (The Hard Stuff)", HTML5 Developer Conference, San Francisco, CA, October 15-16, 2012 <http://html5devconf.com/archives/october2012/index.html>(15.09.2015)



Slavimir Vesić – JKP „Beogradski vodovod i kanalizacija“

Kontakt: vesic.slavimir@gmail.com

Oblasti interesovanja: Web programiranje, interoperabilnost, softverske arhitekture, interakcija čovek računar



Miroslav Minović je docent na katedri za informacione tehnologije pri Fakultetu organizacionih nauka, kao i istraživač pri laboratoriji za multimedijalne komunikacije

Kontakt: miroslav.minovic@fon.bg.ac.rs

Oblasti interesovanja: interakcija čoveka i računara, multimediji i razvoj korisničkih interfejsa

CIP – Katalogizacija u publikaciji Narodna biblioteka Srbije, Beograd 659.25

INFO M : časopis za informacionu tehnologiju i multimedijalne sisteme = journal of information technology and multimedia systems /

glavni i odgovorni urednik Dejan Simić.

– Štampano izd. – God. 1, br. 1 (2002) – . – Beograd : Fakultet organizacionih nauka, 2002 – (Stara Pazova : SAVPO). – 30 cm

Tromesečno. – Je nastavak: Info Science = ISSN 1450-6254. – Drugo izdanje na drugom medijumu: Info M (CD-ROM izd.) = ISSN 1451-4435

ISSN 1451-4397 = Info M (Štampano izd.) COBISS.SR-ID 105690636