

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Bojan Bajc

# **Primerjava spletnih brskalnikov**

DIPLOMSKO DELO NA  
VISOKOŠOLSKEM STROKOVNEM ŠTUDIJU

Mentor: dr. Igor Rožanc

Ljubljana, 2009



Št. naloge: 00404/2008

Datum: 15.10.2008

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **BOJAN BAJC**

Naslov: **PRIMERJAVA SPLETNIH BRSKALNIKOV**  
**THE WEB BROWSERS COPARISON**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija

Tematika naloge:

Učinkovita uporaba svetovnega spleta zahteva primerno orodje – spletni brskalnik. Obstaja več uveljavljenih spletnih brskalnikov, ki na prvi pogled dobro služijo svojemu namenu, a imajo s stališča zahtevnejših uporabnikov tudi pomembne pomankljivosti.

V diplomski nalogi primerjajte zadnje verzije petih najbolj razširjenih spletnih brskalnikov v Sloveniji. Pri primerjavi ustrazno upoštevajte različne vidike uporabe brskalnikov, predvsem njihove zmogljivosti, funkcionalnost, varnost in podporo spletnim standardom. Slednje v nalogi tudi ustrezno predstavite, podporo brskalnikov standardom pa preverite tudi z ustreznimi spletnimi testi. Na koncu dobljene rezultate analizirajte s programom HiView3 in podajte celovite ocene vseh brskalnikov.

Mentor:

viš. pred. dr. Igor Rožanc



Dekan:

prof. dr. Franc Solina

# IZJAVA O AVTORSTVU

## diplomskega dela

Spodaj podpisani/-a Bajc Bojan ,

z vpisno številko 63020192 ,

sem avtor/-ica diplomskega dela z naslovom:

Primerjava spletnih brskalnikov

---

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal/-a samostojno pod mentorstvom (naziv, ime in priimek)

dr. Igor Rožanca

in somentorstvom (naziv, ime in priimek)

---

- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne \_\_\_\_\_ Podpis avtorja/-ice: \_\_\_\_\_

## **Zahvala**

Zahvaljujem se mentorju dr. Igorju Rožancu.

Zahvaljujem se še staršem in puncu, ki so me podpirali in bodrili pri izdelavi diplomskega dela.

# Kazalo

1. Uvod.....	1
2. Standardi.....	2
2.1. Spletni standardi.....	2
2.1.1. Predstavitev.....	2
2.1.2. World Wide Web Consortium (W3C).....	3
2.2. Nekaj internetnih standardov in tehnologij.....	3
2.2.1. Označevalni jezik HTML.....	3
2.2.2. Razširljiv označevalni jezik XML.....	4
2.2.3. Razširljiv označevalni jezik XHTML.....	5
2.2.4. Stilske predloge CSS.....	6
2.2.5. Tehnologija RSS.....	7
2.2.6. Spletni skriptni jezik ECMAScript.....	8
2.2.7. Model DOM.....	9
3. Spletni brskalniki.....	10
3.1. Kaj je spletni brskalnik.....	10
3.2. Aktualni spletni brskalniki.....	10
3.2.1. Internet Explorer 7.0.....	11
3.2.2. Mozilla Firefox 3.x.....	11
3.2.3. Opera 9.x.....	12
3.2.4. Apple Safari 3.x.....	12
3.2.5. Google Chrome 1.x.....	12
4. Primerjava spletnih brskalnikov.....	13
4.1. Razširjenost.....	13
4.1.1. Slovenija.....	13
4.1.2. Svet.....	14
4.2. Fizične zmogljivosti.....	15
4.2.1. Hitrost zagona brskalnikov.....	15
4.2.2. Hitrost odpiranja internetnih strani.....	15
4.2.3. Poraba sistemskih sredstev.....	17

4.3. Funkcije in druge zmogljivosti .....	19
4.3.1. Uporabniški vmesnik.....	19
4.3.2. Podpora razvijalcem .....	22
4.3.3. Varnost.....	28
4.3.4. Druge značilnosti.....	31
4.4. Podpora standardom .....	32
4.4.1. W3C.....	32
4.4.2. Acid2.....	33
4.4.3. Acid3.....	35
4.4.4. CSS3 .....	36
4.4.5. JavaScript.....	43
5. Analiza rezultatov s programom HiView3.....	46
5.1. Hierarhični model.....	47
5.2. Opis in ocenjevanje kriterijev .....	48
5.3. Utežitev kriterijev.....	52
5.4. Analiza rezultata.....	54
5.5. Prihajajoče verzije brskalnikov .....	57
6. Sklepne ugotovitve.....	59
Dodatek A .....	60
Seznam slik.....	60
Seznam tabel.....	61
Literatura .....	62

## **Povzetek**

Namen diplomske naloge je primerjava spletnih brskalnikov. Pri tem spoznamo, kaj je bil razlog za razvoj spletnih brskalnikov in kasneje spletnih standardov. Glavne spletne standarde podrobneje opišemo in predstavimo njihov prispevek k delovanju spleta in posredno informacijskih tehnologij. Skozi diplomsko nalogo ugotavljamo, kako pomembni so spletni standardi, ki jih brskalniki žal uporabljajo različno (slabše ali boljše) ali jih celo ne uporabljajo.

Diplomska naloga obsega primerjavo petih najbolj razširjenih brskalnikov na svetu in v Sloveniji. Za začetek opišemo, kako so razširjeni v Sloveniji in po svetu. Nato nas zanimajo njihove fizične zmogljivosti, predvsem hitrost in obremenitev računalnika. Nato si ogledamo brskalnike z uporabniškega vidika, ocenimo njihovo varnost in to, kakšno podporo nudijo razvijalcem spletnih strani. Zadnji testi so namenjeni preverjanju njihove podpore spletnim standardom. Tu nas zanima stopnja podpore spletnim standardom, katere standarde uporabljajo in kako hitri so pri tem.

V zadnjem poglavju predstavimo program za analizo in obdelavo podatkov HiView3, s katerim analiziramo spletne brskalnike in predstavimo dobljene rezultate. Za konec s pomočjo programa razglasimo zmagovalca in poraženca med brskalniki.

**Ključne besede:** splet, spletni standardi, spletni brskalniki, analiza podatkov, HiView3.

## **Abstract**

The purpose of diploma is web browser comparison. We recognize the reason for web browsers and web standards development. We describe in details the main web standards and present their contribution to web operating and also their indirect contribution to the information technologies. This diploma shows how important are web standards and how different browsers implement them differently or even not at all.

The main aim of diploma is comparison and testing of five most-used browsers in the world and in Slovenia. At first we take a look at how they are distributed and continue with their performance, especially their speed in page loading and usage of memory. Further we test browsers from a user point of view. We evaluate their safety and the level of support to web page developers. We also recognize the importance of web standards through latest browser tests for their web standards support. The tests reveal how strong the support is and how fast the browsers implement individual standards.

In the last we present the program for the analysis and data processing HiView3. We also used HiView3 to analyze and present the results obtained in the web browsers testing. To conclude, we declare a winner and loser among the browsers we tested.

Keywords: web, web standards, web browsers, data analysis, HiView3.



## 1. Uvod

Od kar je Tim Berners-Lee začel razvijati splet kot ga poznamo danes, se med izdelovalci brskalnikov bje nenehen boj za svoj delež med končnimi uporabniki. V zgodovini se spomnimo rivalstva med Netscape Navigatorjem in Internet Explorer-jem, danes pa je na trgu še veliko več brskalnikov in bistveno močnejša konkurenca kot v preteklosti. Spletni brskalniki morajo biti uporabnikom čim bolj všečni, hitri, stabilni, varni in pri tem ne smejo pozabiti na podporo standardom, saj le tako ne bodo odšli na smetišče zgodovine, kot se je to zgodilo že nekaterim.

Hitrost razvoja spletnih standardov je veliko hitrejša kot je razvoj podpore s strani izdelovalcev spletnih brskalnikov. Razvijalci spletnih strani in aplikacij imajo pri tem velike težave, saj pri razvoju ne morejo uporabiti vseh razpoložljivih tehnologij, ker jih spletni brskalniki ne podpirajo. Nekateri izdelovalci spletnih brskalnikov se tega problema zavedajo in hitijo z razvojem novih različic brskalnikov, ki bodo sledile razvoju spletnih standardov, spet drugi si pri tem vzamejo več časa in s tem tvegajo padec popularnosti med uporabniki.

V teh časih splet doživlja svoj razcvet. Spletne strani rastejo kot gobe po dežju. Vse večjo popularnost dosegajo spletne aplikacije, ki slonijo na tehnologiji JavaScript-a, čigar podporo smo preverjali v diplomski nalogi. Javanska tehnologija slovi po tem, da ni preveč hitra, zato je še posebej zanimivo opazovati, kako je posamezen brskalnik hiter pri njenem izvajanju. Pri tem izdelovalci brskalnikov izbirajo svojstvene poti pri implementaciji in s skritimi triki poskušajo pohitriti izvajanje JavaScript-a. Seveda se morajo pri tem držati standardov. Spletnih standardov je vedno več, njihova uporaba pa postaja nuja pri razvoju sodobnih, interaktivnih, večpredstavnostnih in hitrih spletnih aplikacij.

Z letom 2008 imamo na tržišču največ brskalnikov do sedaj, v diplomski nalogi si jih bomo ogledali le nekaj, in sicer: Internet Explorer verzije 7, Mozilla Firefox verzije 3, Opero verzije 9, Apple Safari verzije 3 in novost med brskalniki Google Chroma verzije 1.

Namen diplomske naloge je:

- orisati razvoj spletnih standardov in brskalnikov skozi zgodovino,
- predstaviti glavne spletne standarde, ki so osnova za delovanje spleta,
- predstaviti aktualne spletne brskalnike, poudariti njihove slabosti in prednosti,
- primerjati spletne brskalnike med seboj in
- analizirati spletne brskalnike s programom HiView3.

Cilj diplomske naloge je:

- bralca seznaniti s spletnimi tehnologijami in standardi,
- ugotoviti, katere so prednosti in slabosti posameznih brskalnikov,
- raziskati, kako posamezni brskalniki podpirajo najnovejše spletne standarde in
- s pomočjo programa HiView3 objektivno proglasiti, kateri brskalnik je najboljši na testu.

## 2. Standardi

Standardi so dokumentirani dogovori, ki vsebujejo tehnične specifikacije ali druge natančno določene kriterije, ki se konsistentno uporabljajo kot pravila, smernice ali definicije lastnosti, da bi materiali, izdelki ali storitve ustrezale svojim namenom. [1]

Standardizacija oziroma poenotenje je metoda, katere bistvo je odstranjevanje odvečne raznovrstnosti in določanje izenačenosti glede na: kakovost, obliko, mere (dimenzije), materiale, varnost, zanesljivost ter življenjsko dobo izdelka. [2]

Primer: Vsa Evropa uporablja napetost 230 V/50Hz (čeprav različne vtiče), enotno GSM telefonijo in (vsaj uradno) metrski sistem.

### 2.1. Spletni standardi

#### 2.1.1. Predstavitev

Svetovni splet (ang. World Wide Web ali WWW) je porazdeljen hipertekstni (nadbесedilni) sistem, ki deluje v medmrežju. Hipertekstne dokumente pregledujemo s programom imenovanim spletni brskalnik, ki s spletnega strežnika prenese dokument in ga prikaže na računalniškem zaslonu. Besedilnim spletnim dokumentom pravimo spletna stran, smiselno povezanim spletnim mestom pa spletišče. V spletnih straneh so lahko povezave, ki kažejo na druge spletne strani ali pošljejo povratno informacijo spletnemu strežniku. Za sprehajanje po spletnih straneh se uporablja izraz deskanje.

Začetki spleta segajo v leto 1989, ko sta Tim Berners-Lee in Rober Cailiau v Evropskem središču za jedrske raziskave CERN razvila sistem ENQUIRE (angleška kratica za »Enquire Within Upon Everything«). Tim Berners-Lee je želel ustvariti sistem, ki bi poenostavil dostop do tekstnih informacij, ki so bile porazdeljene po različnih strežnikih v CERN-u.

Predlog za splet, kot ga poznamo danes, je Berners-Lee predstavil 12. novembra 1990, prvo spletno stran pa le dan pozneje. Čez božične praznike je izdelal vsa potrebna orodja za izdelavo spletne strani, brskalnik za ogled spletne strani, ki je bil obenem tudi urejevalnik spletne strani ter spletni strežnik.

Njegove sanje so bile, da bi splet postal skupen prostor, kjer bi si uporabniki delili informacije, skupaj delali, se družili in igrali. S tem, ko spletni razvijalci razvijajo učne, informativne, poslovne in družabne spletne strani, postajajo njegove sanje tudi realnost.

Ob veliki rasti spletnih strani, spletni razvijalci potrebujejo smernice in navodila, da lahko razvijejo spletno stran v polni zmogljivosti. Tem smernicam ali navodilom pravimo spletni standard (ang. Web standard). Standardi omogočajo vsem razvijalcem enakovreden dostop do informacij, ki jih potrebujejo, in omogočajo hitrejši in enostavnejši razvoj spletne strani.

Oktobra leta 1994 je izumitelj spleta Tim Berners-Lee zapustil Evropsko organizacijo za jedrske raziskave (CERN) in ustanovil inštitut World Wide Web Consortium (W3C). [3, 5]

## 2.1.2. World Wide Web Consortium (W3C)

World Wide Web Consortium (W3C) je mednarodni inštitut, kjer člani organizacije, zaposleni in javnost sodelujejo ter skupaj razvijajo standarde za splet (WWW).

Namen inštituta je naslednji: Voditi razvoj polno zmogljivih spletnih strani s pomočjo protokolov, pravil in smernic, ki zagotavljajo dolgoročno rast za svetovni splet. [4]

Inštitut skupno upravljajo Computer Science and Artificial Intelligence Laboratory (CSAIL) na Massachusetts Institute of Technology (MIT) v Združenih državah Amerike, European Research Consortium for Informatics and Mathematics (ERCIM) v Sophia Antipolisu, Francija in Keio University na Japonskem.

W3C ima pisarne v petnajstih državah po celem svetu. V pisarnah W3C-ja se ukvarjajo s svojimi spletnimi skupnostmi, s katerimi promovirajo tehnologije W3C-ja v lokalnih jezikih in tako širijo mednarodno sodelovanje in aktivnosti W3C-ja. [5]

## 2.2. Nekaj internetnih standardov in tehnologij

### 2.2.1. Označevalni jezik HTML

Označevalni jezik HTML (ang. Hyper Text Markup Language) je spletni jezik za izdelavo spletnih strani. Predstavlja osnovo spletnega dokumenta. S pomočjo HTML-ja ustvarimo strukturo in semantično ureditev dokumenta. Pišemo ga lahko v vsakem urejevalniku besedil (beležnici idr.) in je dokaj preprost, zato ga lahko kombiniramo tudi s PHP-jem in drugimi programskimi jeziki. Razvil ga je že omenjeni Tim Berners-Lee na inštitutu CERN leta 1990. [6]

Prvi javni opis HTML-ja je bil dokument imenovan HTML značke (ang. HTML Tags) iz leta 1991. Opisoval je 22 začetnih elementov HTML dokumenta in je bil dokaj enostavno zasnovan. Od 22-ih elementov jih zadnja verzija HTML-ja vsebuje še 13. Sledilo je 5 verzij HTML-ja. Prvi HTML, ki ga upoštevamo kot standard je HTML verzije 2 iz leta 1995. Z letom 1997 dobimo verzijo HTML 3.2, ki postane priporočilo W3C organizacije. Z verzijo 3.2 HTML iz svoje dokumentacije izpusti matematične formule in postane samostojni standard za spletni jezik. Matematične formule se kasneje vključujejo v HTML preko standarda MathML. Konec leta 1997 dobimo HTML 4.0, ki ponuja 3 stopnje: strogi (strict), prehodni (transitional) in stopnja z okvirji (frameset). Leta 1998 različica 4 dobi nekaj popravkov, vendar se številka verzije ne poveča. Leta 1999 HTML preide v verzijo 4.1, ki jo uporabljamo še danes. [6]

Prvi brskalnik z grafičnim uporabniškim vmesnikom, ki je podpiral HTML, je bil Mosaic. Razvila sta ga Marc Andreessen in Eric Bina leta 1993. Zaradi enostavnega uporabniškega vmesnika, zanesljivosti in enostavne namestitve se je hitro razširil med ljudmi. Tako se je Mosaic hitro razširil tudi globalno in močno prispeval k razvoju interneta.

Moderni brskalniki, kot sta Internet Explorer in Mozilla Firefox, temeljijo na Mosaicu, kar priča o tem, da je bil Mosaic pravilno zasnovan. Sodobni brskalniki so obdržali veliko značilnosti njegovega uporabniškega vmesnika in delovanja. [7]

V začetku, ko je HTML vseboval samo nekaj osnovnih značk (ang. tag) se ljudje še niso zavedali pomena standarda. Spletne strani so izgledale lepo v vseh brskalnikih. Ko so se v HTML-ju pojavile še slike, različne pisave in razne opcije urejanje izgleda strani, so se avtorji brskalnikov odločili za drugačen pristop. S svojimi brskalniki so nudili drugačen, lepši in hitrejši, a ne nujno boljši prikaz HTML-ja. S tem so si želeli povečati tržni delež, saj so želeli, da njihov brskalnik prikaže stran za uporabnika lepše kot konkurenčen. Že takrat je obstajalo rivalstvo med brskalniki za prevlado na trgu. Razvijalci so se temu seveda uprli, saj so želeli, da bi njihova stran izgledala enako v vseh brskalnikih. Zahtevali so standard in ločitev izgleda strani od vsebine strani. Tako se je rodil HTML standard kot ga poznamo sedaj. [8]

Na sliki 1 je primer HTML kode, ki v brskalniku izpiše naslov »Primer HTML strani.«.

```

1<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN"
2  "http://www.w3.org/TR/html4/strict.dtd">
3<html>
4<head>
5<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
6<title>Naslov</title>
7</head>
8<body>
9  <h1>Primer HTML strani.</h1>
10</body>
11</html>

```

*Slika 1: Primer HTML zapisa.*

## 2.2.2. Razširljiv označevalni jezik XML

Razširljiv označevalni jezik XML (ang. Extensible Markup Language) je jezik, ki ga pogosto srečamo, če brskamo po spletu. XML je preprost računalniški jezik podoben HTML-ju, ki nam določa format za opisovanje strukturiranih podatkov, lahko pa tudi strukturo za prenos podatkov in njihovo izmenjavo med več omrežji.

XML na več načinov vpliva na računalništvo, še posebej na področju komuniciranja med aplikacijami in strežniki. Je razširljiv, saj ima to možnost, da lahko sami določimo imena značk. Zelo je uporaben za komunikacije, saj ima zelo preprosto in pregledno zgradbo.

XML torej zagotovi strukturo in tako omogoči prenosljivost dokumenta. Izgled oziroma obliko XML dokumenta določimo z uporabo standardov grafičnih predlog, kot sta CSS in XLS. Ponujata nam veliko paleto možnosti, kako lahko opišemo izgled XML elementov znotraj dokumenta. Podrobnosti o standardih grafičnega oblikovanja si bomo ogledali pozneje.

XML je razdeljen na 3 dele:

- podatkovni (vanj shranimo podatke v ustrezni obliki),
- deklarativni (skrbi za to, da lahko pri dodajanju novih podatkov vidimo kaj določena značka predstavlja) in
- predstavitevni (z njim oblikujemo izpis podatkov).

Razvijalci XML nenehno povečujejo vsebino tega jezika in s tem tudi njegovih standardov, da vsebujejo podatke, ki se jih da enostavno preoblikovati in zamenjati v neenakih sistemih.

Kot je razvidno iz slike 2, moramo pri XML zapisu najprej deklarirati dokument, nato sledijo naše značke. Vsako značko (prazno ali polno) moramo pravilno začeti in pravilno zaključiti. [9]

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <imenik>
3   <oseba>
4     <ime>Bojan</ime>
5     <priimek>Bajc</priimek>
6     <telefon>040 349 773</telefon>
7     <email>bojan.bajc@gmail.com</email>
8   </oseba>
9   <oseba>
10    <ime>Janez</ime>
11    <priimek>Novak</priimek>
12    <telefon> / </telefon>
13    <email>janez.novak@gmail.com</email>
14  </oseba>
15</imenik>

```

Slika 2: Primer XML zapisa.

### 2.2.3. Razširljiv označevalni jezik XHTML

Razširljiv označevalni jezik za označevanje nadbesedila XHTML (ang. Extensible Hyper Text Markup Language) uporabljamo za pisanje spletnih strani. Z razliko od HTML-ja se XHTML drži bolj striktnih pravil in je na videz bolj podoben XML-ju.

XHTML mora biti pravilno tvorjen (značka mora biti zaključena, npr: <h1>Naslov</h1>) zato, da ga lahko razčlenimo s preprostimi razčlenjevalniki XML, za razliko od HTML-ja, ki potrebuje prilagojene razčlenjevalnike.

Uporaba XHTML 1.0 je priporočilo inštituta W3C od 26. januarja 2000, XHTML 1.1 pa 31. maja leto kasneje.

Originalen XHTML W3C-standard, XHTML 1.0, je bil preprost prehod iz HTML 4.01 v XML. XHTML 1.0 se pojavlja v treh »različicah«, pri kateri se vsaka zgleduje po njeni vzporednici v HTML.

- XHTML 1.0 Strict je enak HTML 4.01 Strict, vendar sledi sintaksnim pravilom XML.
- HTML 1.0 Transitional dovoljuje uporabo nekaterih pogosto uporabljenih značk in elementov (ang. attribute), ki se v XHTML 1.0 Strict ne pojavljajo, npr. <center>, <u>, <strike> in <applet>. Podpira tudi vse drugo, kar podpira HTML Strict, vendar vključuje elemente in attribute za stil in je priporočen za brskalnike, ki so nastali v času okoli leta 1990, saj imajo ti brskalniki težave s prikazovanjem CSS predlog.
- XHTML 1.0 Frameset: Dovoljuje uporabo HTML-okvirjev (<frames>). [10]

Ko pri pisanju spletne strani upoštevamo XHTML standard, s tem pišemo čistejšo, pravilnejšo in prilagodljivejšo kodo.

XHTML nas zavezuje:

- dokument mora biti deklariran z uporabo značke DOCTYPE
- struktura dokumenta se mora začeti z deklaracijo DOCTYPE, značko HTML v kateri moramo določiti tudi XHTML naslov, sledita mu znački HEAD in TITLE in seveda značka BODY
- značke in imena atributov pišemo z malimi črkami
- da vsako odprto značko zapremo, to velja tudi za značke, ki jih navadno ne zapiramo, npr. `<br>` pišemo kot `<br />`
- da značke zaključujemo v pravilnem vrstnem redu, npr: `<li><strong>Beseda</strong></li>`
- pri slikah uporabljamo atribut alt, ki v primeru, ko brskalnik ne zna prikazati slike, prikaže nadomestno besedilo, npr. `<img src='url_do_slike.jpg' alt='Nazaj' />`
- izogibamo se pisanju besedila v telesu dokumenta (`<body>`). To naredimo raje tako: `<body><div>Moje besedilo</div> ...`
- vrednosti atributov vedno pišemo v narekovajih, npr: `<div style='float:left;'></div>`
- namesto zanka & uporabljamo `&amp;`; Npr. `<h1>Testi &amp; primeri</h1>` [11]

Stran, ki preveri, če določena spletna stran ustreza XHTML standardu, je <http://validator.w3.org/>.

Na sliki 3 je primer kode, ki v brskalniku izpiše »Primer XHTML strani.«.

```

1<?xml version="1.0" encoding="UTF-8" ?>
2<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4<html xmlns="http://www.w3.org/1999/xhtml">
5<head>
6<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
7<title>Naslov</title>
8</head>
9<body>
10  <h1>Primer XHTML strani</h1>
11</body>
12</html>

```

Slika 3: Primer XHTML zapisa.

## 2.2.4. Stilske predloge CSS

Stilske predloge CSS (ang. Cascading Style Sheets) so predloge, ki določajo izgled spletnih strani. Z njimi določamo pisavo, velikosti črk ter vizualno predstavitev spletne strani. HTML naj bi predstavljal semantično strukturo in smiselno hierarhijo dokumenta, CSS pa ima predstavitevno vlogo.

CSS podpirajo vsi novejši spletni brskalniki (Internet Explorer 4.0 in več, Netscape Navigator 4.72, Mozilla, Opera). Spletni brskalniki, ki standarda CSS ne podpirajo, bodo prikazali običajen HTML dokument brez oblikovanja.

CSS je bil izdelan z namenom, da loči vsebino dokumenta (napisanega v HTML-ju ali katerem izmed drugih označevalnih jezikov) od izgleda dokumenta (za katero skrbi CSS). Z uporabo CSS-ja lažje spreminjamo izgled strani, stran je bolj fleksibilna in prilagodljiva. Zmanjša se tudi zahtevnost strani, odpravijo se ponavljanja (npr. neprestano oblikovanje naslovov, saj lahko izgled vseh naslovov določimo v predlogi CSS). Za spremembo izgleda strani ni potrebno spreminjati HTML-ja, ampak samo dopolnimo ali popravimo CSS dokument.

S CSS-jem si pomagamo tudi pri predstavitvi iste strani v različnih oblikah (za branje, tiskanje, itd). Ko uporabnik prebira vsebino strani je stran prilagojena uporabniku, da jo ta lažje razume. Uporabnik pri branju strani uporablja določene povezave (gumbe), ki ga usmerjajo po dokumentu na strani. Določeni deli teksta in gumbi so posebej označeni, da jih uporabnik lažje najde in uporablja. Posebej označenih gumbov in teksta pa ne potrebujemo, ko želimo stran natisniti. Za odstranitev vseh barvnih gumbov, ki na papirju nebi bili v nikakršno pomoč, skrbi ravno CSS, ki računalniku narekuje, kako brskalnik prikaže stran uporabniku in kako tiskalniku.

Problemi, ki so jih povzročali brskalniki s svojstveno implementacijo HTML-ja, se sedaj čutijo tudi pri njihovem razumevanju CSS standarda. Tako imajo razvijalci spletnih strani dvojno delo, da njihova stran izgleda enako v dveh različnih brskalnikih. O tem bomo podrobneje govorili, ko bomo primerjali, kako s CSS-jem oblikovane HTML elemente prikazujejo različni brskalniki. [12]

Spletna stran, ki preveri, če določena spletna stran ustreza standardu CSS je: [http://jigsaw.w3.org/css-validator/#validate\\_by\\_uri](http://jigsaw.w3.org/css-validator/#validate_by_uri). Primer CSS predloge vidimo na sliki 4.

```

3/* vsi elementi na strani bodo imeli pisavo Tahoma */
4
5*{
6    font-family: Tahoma;
7}
8
9/* vsi naslovi h1 bodo rdeče barve in
10bodo imeli pisavo velikosti 15px */
11
12h1{
13    color: red;
14    font-size: 15px;
15}

```

Slika 4: Primer CSS-ja.

## 2.2.5. Tehnologija RSS

Tehnologija za preprosto objavljanje informacij RSS (ang. Really Simple Syndication), je tehnologija, ki omogoča naročanje na vire podatkov posameznih spletnih mest, ki jih uporabnik prebira in ureja z enega mesta. Pojavila se je v času porasta spletnih časopisnih in televizijskih strani. Te strani novice posodablja dnevno ali celo v urnih intervalih, zato bralec težko sledi novicam, ki ga zanimajo.

Za delovanje tehnologije RSS je potrebno dvoje. Najprej mora spletna stran omogočati RSS. Taka stran je označena z oranžnim kvadratom desno od URL (ang. Uniform Resource Locator) naslova strani (slika 5).



*Slika 5: Primer spletne strani, ki omogoča RSS.*

Drugi pogoj pa je odjemalec (program) za RSS datoteke. Ta program izpisuje informacije, ki jih pridobi iz različnih RSS kanalov (tematsko razdeljene informacije, npr. šport, zabava, itd). Uporabnik se odloči, katere RSS kanale želite pregledovati. Veliko spletnih strani omogoča, da lahko določimo, katere rubrike nas zanimajo. Tako nas RSS obvešča samo o naši izbrani temi, na primer športu.

Na spletu je na voljo veliko različnih RSS odjemalcev:

- samostojni programi za branje RSS novic,
- RSS odjemalci, vključeni v program za e-pošto,
- RSS odjemalec preko spletnega brskalnika in
- naročanje na prejemanje RSS novic preko e-pošte.

Vgrajen RSS bralnik omogoča branje novic kar v samem brskalniku. Seveda je od same spletne strani odvisno, ali RSS vsebino podpira. Dva največja slovenska obveščevalna portala, 24ur.com in rtvslo.si, podpirata RSS vsebino. Ta omogoča branje nam zanimivih novic brez obiskovanja neke spletne strani. Informacije se shranjujejo v RSS datoteki, ki jo dobimo od ponudnika takih storitev na spletni strani. [13]

## 2.2.6. Spletni skriptni jezik ECMAScript

ECMAScript je spletni skriptni jezik, ki ga je standardizirala Ecma International v specifikaciji ECMA-262. Ecma je mednarodna, zasebna, neprofitna organizacija, ki se ukvarja s standardi informacijskih in komunikacijskih sistemov. Jezik se široko uporablja na spletu, pogosto pa ga zamenjujemo z JavaScript (Netscape) in JScript (Microsoft). To sta le dva skriptna jezika na osnovi katerih je bil ECMAScript standardiziran. Osnovna lastnost spletnih skriptnih jezikov je ta, da se njihova koda izvaja na odjemalčevem (našem) računalniku. Prednost takšnega izvajanja je manjša obremenjenost strežnikov. Na sliki 6 vidimo funkcijo za izračun datuma, ki je napisana po Ecma standardu.

```

5 function getDateAndHour() {
6     var currentTime = new Date();
7     var month = currentTime.getMonth() + 1;
8     var day = currentTime.getDate();
9     var year = currentTime.getFullYear();
10    var currentTime = new Date();
11    var hours = currentTime.getHours();
12    var minutes = currentTime.getMinutes();
13    minutes = minutes.length < 2 ? "0" + minutes : minutes;
14    var mojDatum = day+"."+month+"."+year+" "+hours+":"+minutes;
15    return mojDatum;
16}

```

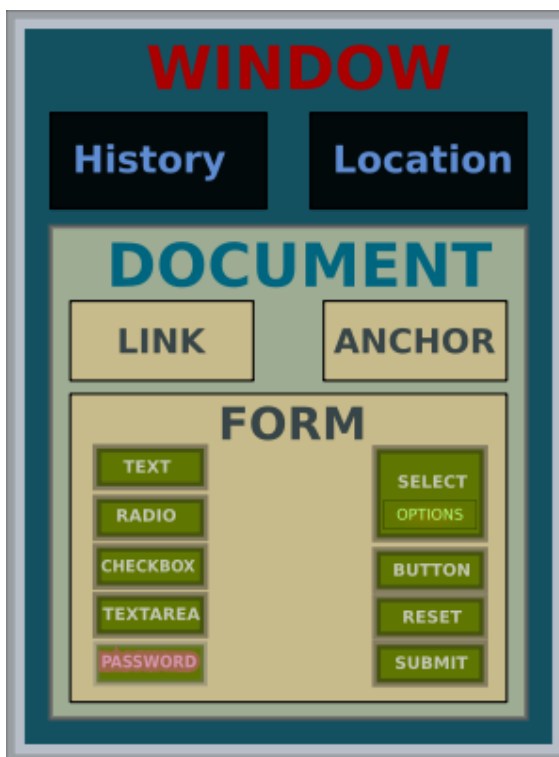
*Slika 6: Primer EcmaScript-e.*



Cilj izdelave ECMAScripta je izdelati industrijsko priporočilo za izdelovalce brskalnikov, kako naj bo v njihovih brskalnikih implementiran spletni skriptni jezik. Standard določa, kako so lahko obdelani podatkovni tipi, kako delujejo operatorji, kako izgleda posebna specifična podatkovna skladnost in ostale jezikovne lastnosti. Novejša verzija 3 doda veliko novosti k osnovnemu jeziku (verzija 2 je bila verzija 1 s popravljenimi napakami). ECMAScript sintaksa je podobna sintaksi C-ja. Je objektno usmerjena, večina vrednosti so objekti. [14]

### 2.2.7. Model DOM

Model DOM (ang. Document Object Model) je vmesnik, ki je neodvisen od platforme in jezika in omogoča programom in skriptam dinamičen dostop do vsebine, posodabljanje vsebine, strukture in sloga dokumenta. Ta dokument se lahko nadalje obdeluje in rezultati obdelave se lahko vključijo nazaj na predstavljeno stran. Vsi gradniki vsebine (tekstovna okenca, gumbi) so v modelu DOM predstavljeni v hierarhiji objektov (slika 7).



Slika 7: Hierarhija objektov v primeru modela DOM.

Dokument oblikujemo na samem odjemalcu, torej ne na strežniku ali katerim drugim oddaljenim računalnikom. Temu pravimo dinamični HTML (ang. Dynamic HTML), ki s pomočjo kombinacije HTML-ja, skriptnih jezikov (JavaScript) in stilskih predlog (CSS) omogoča dinamično oblikovanje dela ali celote dokumenta brez ponovnega nalaganja strani. Lahko rečemo, da se dokument s pomočjo DOM-a animira. [15]

Primer uporabe DHTML-ja: Vrednost okna z identifikatorjem »txtTestnoOkno« smo spremenili s pomočjo JavaScript-a in nismo kakorkoli zaposlili strežnik, ki gosti spletno stran. Vse smo opravili na odjemalcu (našem računalniku).

```
document.getElementById(»txtTestnoOkno«).value=»Nova vrednost.«;
```

## **3. Spletni brskalniki**

### ***3.1. Kaj je spletni brskalnik***

Spletni brskalnik je program, ki nam omogoča ogled večpredstavnostnih vsebin, kot je tekst, slika, video, glasba in igre, ki se tipično nahajajo na spletni strani, ta pa se nahaja na internetu ali na lokalnem omrežju. Spletna stran vsebuje tudi povezave na druge strani, preko katerih se lahko s pomočjo brskalnika hitro in enostavno premikamo.

Zgodovina brskalnikov se začne v poznih osemdesetih letih, ko so zaradi raznolikosti tehnologij postavili temelje za prvi spletni brskalnik. Imenoval se je WorldWideWeb in je deloval na računalniku NeXT. Napisal ga je Tim Berners-Lee leta 1990. Kasneje ga je preimenoval v Nexus, da bi preprečil mešanje imena brskalnika in imena samega spleta. Skozi leta so svoje brskalnike predstavila še druga podjetja kot so Mozilla, Netscape, Microsoft, Apple in Opera. Pred kratkim je predstavil svoj spletni brskalnik še internetni gigant Google.

Brskalnik je moral vso raznolikost spletnih tehnologij združiti na enem mestu. Brskalnik komunicira s spletnimi strežniki preko protokola HTTP (ang. Hypertext Transfer Protocol), kateri mu nato vrne spletno stran. Spletno stran dobi prek URL-ja, kateri je njen naslov. Spletna stran, ki jo brskalnik priskrbi za naš računalnik (odjemalec) od oddaljenega računalnika (strežnik), je ponavadi HTML datoteka. Razvijalci se HTML-ju vse bolj izogibajo in raje uporabljajo novejši XHTML standard, ki ga pravilno predstavi večina brskalnikov. [16]

### ***3.2. Aktualni spletni brskalniki***

Na trgu je veliko spletnih brskalnikov. Izbrali smo 5 aktualnih, in sicer v trenutno zadnji stabilni verziji, ker te navadno uporabljajo končni uporabniki. Na voljo so tudi nedokončane novejšje verzije, ki se odrežejo veliko bolje kot starejše dokončane. Omenimo naj samo Internet Explorer 8, Firefox 3.1, Opera 10, Safari 4 in Chrome 2, ki so po testih veliko boljši od aktualnih verzij. Izbrani spletni brskalniki so:

- Internet Explorer 7.0
- Mozilla Firefox 3.0.x
- Opera 9.x
- Apple Safari 3.x
- Google Chrome 1.x

### 3.2.1. Internet Explorer 7.0



Internet Explorer je brskalnik podjetja Microsoft. Deluje pod okoljem Windows in Macintosh. Razvili so ga leta 1995 na osnovi že obstoječega grafičnega brskalnika Mosaic. Najprej je bil na voljo kot dodatek operacijskemu sistemu Windows 95, v kasnejših verzija operacijskih sistemov Windows pa je bil že avtomatsko vključen v sistem. Ker je bil vključen v zelo razširjen operacijski sistem Windows, je njegova popularnost strmo narasla. Premagal je številne tekmece, kot je npr Netscape Navigator. V zadnjem času pa je dobil na področju priljubljenosti in razširjenosti močno konkurenco, predvsem v obliki Mozille Firefox-a. [17, 18]

Različice brskalnika so:

- Različica 1.0 – avgust 1995
- Različica 2.0 – november 1995
- Različica 3.0 – avgust 1996
- Različica 4.0 – oktober 1997
- Različica 5.0 – marec 1999
- Različica 5.5 – julij 2000
- Različica 6.0 – oktober 2001
- Različica 7.0 Beta – januar 2006
- Različica 7.0 – oktober 2006

### 3.2.2. Mozilla Firefox 3.x



Mozilla Firefox je razvilo podjetje Mozilla Corporation in prostovoljci. Je odprto kodni brskalnik, ki deluje v več operacijskih sistemih, kot so Linux, Windows, Mac in drugi. Mozilla so razvili na osnovi programa Netscape. V primerjavi z Internet Explorer-jem je precej mlad brskalnik. Začetki segajo v leto 2002 ko sta Dave Hyatt in Blake Ross začela s poskusnim projektom. Zanimivo je to, da je brskalnik večkrat zamenjal ime. Najprej se je imenoval Phoenix, na kar so ugotovili, da se tako imenuje Phoenix BIOS, zato ga preimenujejo v Firebird. Ker se je tako imenoval program za upravljanje baze podatkov, ga preimenujejo v Firebird Browser, nato Mozilla Firebird, na koncu pa Mozilla Firefox, kasneje krajše samo Firefox. [19]

### 3.2.3. Opera 9.x



Opera se je rodila kot projekt v največjem norveškem telekomunikacijskem podjetju Telenor leta 1994. V enem letu se je od projekta odcepila neodvisna razvojna družba Opera Software ASA. Danes lahko rečemo, da je Opera produkt podjetja Opera Software, ki je na voljo številnim operacijskim sistemom, kot so Linux, Windows, Mac. Poleg računalnikov lahko Opero uporabljajo tudi mobilne naprave kot so mobilni telefoni, dlančniki, najdemo jo celo v igralnih konzolah, kot je recimo Nintendo Wii. Prva verzija (2.0) brskalnika, ki je bila namenjena širši publiki, je bila izdana leta 1996. Verzija 3.0 je že podpirala različne operacijske sisteme. S kasnejšo verzijo 5.0 je bil brskalnik plačljiv, sledila ji je verzija 8.5, ki je bila zopet brezplačna. Trenutno je v uporabi brskalnik verzije 9.x. [20, 21]

### 3.2.4. Apple Safari 3.x



Safari je razvila družba Apple Inc. Prva beta verzija je bila javnosti predstavljena 7.1.2003 za operacijski sistem Mac OS X. Safari je postal tudi privzeti brskalnik tega operacijskega sistema, kasneje tudi za iPhone OS. Dne 11.6.2007 je Apple izdal različico za operacijski sistem Windows XP in Visto. Manjšo razširjenost brskalnika gre pripisati dejstvu, da dolgo ni bilo različice za operacijske sisteme Windows, ter nekoliko drugačnemu uporabniškemu vmesniku, ki je prilagojen operacijskemu sistemu Mac, ki ga širša množica ni vajena. [22]

### 3.2.5. Google Chrome 1.x



Google Chrome je najmlajši med brskalniki. Razvil ga je internetni velikan Google. Prvo verzijo so izdali 2.11.2008, uradno verzijo pa 11.12.2008. Za enkrat je Chrome namenjen samo uporabnikom operacijskega sistema Windows, v izgradnji pa sta verziji za Linux in Mac OS X. [23]

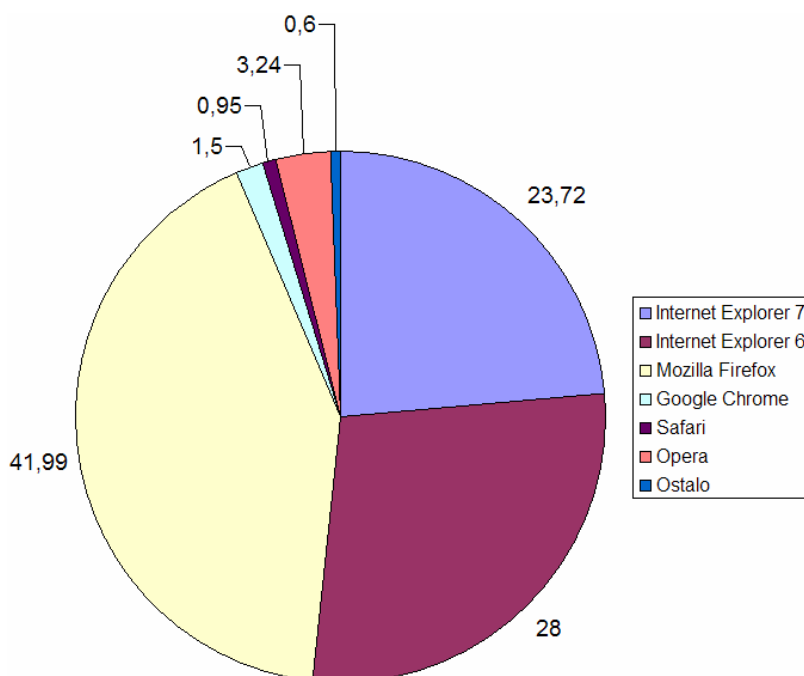
## 4. Primerjava spletnih brskalnikov

V poglavju bomo opisali kriterije, s katerimi lahko primerjamo brskalnike med seboj. Ugotovili bomo, kaj nam brskalniki lahko nudijo, katere tehnologije lahko podpirajo, kakšna je njihova uporabnost in podprtost. Po drugi strani bomo preverili kaj nudijo razvijalcem spletnih strani in spletnih aplikacij, kako si pri razvoju z določenim brskalnikom lahko pomagajo in seveda s čim olajšajo deskanje po spletnih straneh končnim uporabnikom.

### 4.1. Razširjenost

V tem poglavju si bomo ogledali kako so internetni brskalniki razširjeni med uporabniki.

#### 4.1.1. Slovenija



Slika 8: Razširjenost internetnih brskalnikov v Sloveniji.

Na sliki 8 so meritve, ki so bile objavljene na spletni strani podjetja iPROM dne 10.09.2008. Kot kažejo podatki, je v Sloveniji še vedno najbolj razširjen brskalnik podjetja Microsoft Internet Explorer. To gre pripisati predvsem temu, da je brskalnik integriran v operacijski sistem Windows, kar običajnim uporabnikom zadošča. Presenetljivo je to, da je precej razširjena starejša različica 6, katere sprejemljivost je vprašljiva. [24]

Na drugem mestu je brskalnik podjetja Mozilla Firefox. Predvsem v začetnih verzijah je imel Firefox nekaj izboljšav in novitet v primerjav z Explorer-jem, s čimer mu je že od samega začetka največji konkurent.

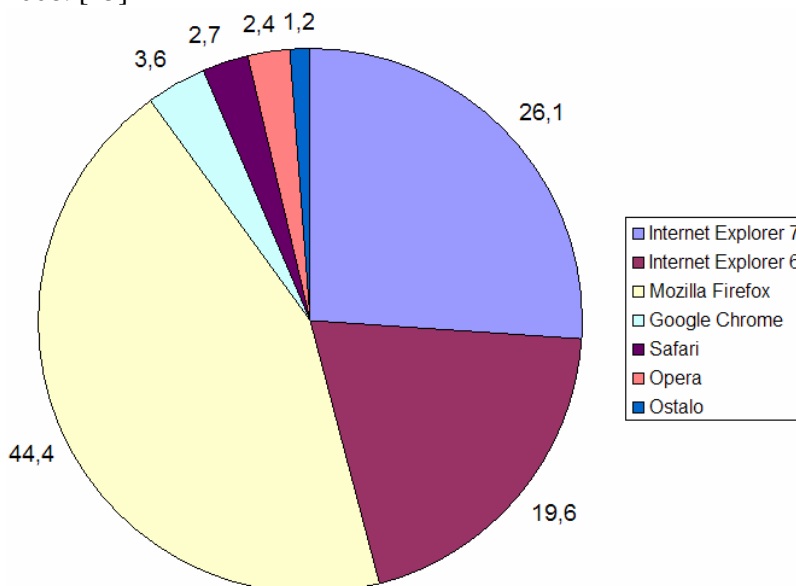
Opera je v Sloveniji očitno še precej neznano ime, vendar gre za enega boljših brskalnikov na trgu. V prihodnosti gre pričakovati, da se bo odstotek uporabe povečal.

Ko je podjetje Apple predstavilo verzijo svojega brskalnika Safari za operacijski sistem Windows, se je podalo v težak boj za odstotke. Zaradi nekoliko zastarelega uporabniškega vmesnika in za uporabnike Windowsov dokaj neznanega imena, si bo ta brskalnik težko izboril boljše mesto.

Nazadnje je tu še brskalnik podjetja Google Chrome. Gre za novinca med brskalniki, ki se poizkuša uveljaviti s pomočjo množičnega reklamiranja. Nedvomno mu bo zaradi dobrega imena podjetja tudi uspelo izboljšati sedanje stanje.

#### 4.1.2. Svet

Meritve na sliki 9 so bile objavljene na spletni strani podjetja w3schools in veljajo za zadnji mesec v letu 2008. [25]



Slika 9: Razširjenost internetnih brskalnikov v svetu.

Tudi po svetu je razširjenost brskalnikov med uporabniki precej podobna tisti v Sloveniji. Razlike vendarle so: Internet Explorer 6 izgublja odstotke, kar gre pripisati nekoliko večji internetni pismenosti uporabnikov po svetu, ki se nekoliko bolj zavedajo zastarelosti svojih brskalnikov.

Nekoliko bolj popularna sta tudi Chrome in Safari, slednji predvsem zaradi večje razširjenosti Apple računalniških sistemov v Ameriki.

## 4.2. Fizične zmogljivosti

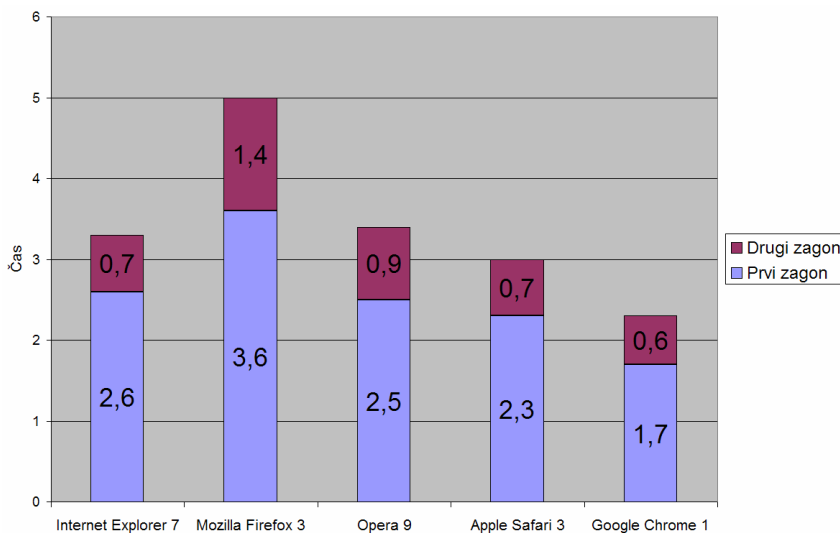
V podpoglavju si bomo ogledali, kako hitro se brskalniki zaženejo brez odprte strani in kako hitro izrišejo celo internetno stran. Nadalje bomo primerjali, koliko sistemskih virov porabijo med delovanjem z odprto internetno stranjo. Ker vsi brskalniki omogočajo več odprtih internetnih strani v enem oknu, bomo testirali kako se poveča poraba sistemskih virov z večanjem števila odprtih internetnih strani v enem oknu. Testni sistem je bil namizni računalnik z eno jedrnim procesorjem Pentium 4 630 s taktom 3.0 Ghz, 2 GB DDR2 pomnilnika, trdi disk SATA in operacijski sistem Windows XP sp3.

### 4.2.1. Hitrost zagona brskalnikov

Testirali smo, koliko časa potrebuje brskalnik, da se zažene prazen (brez odprte strani). Izmerjene čase vidimo na sliki 10. Čas v sekundah smo merili z ročno štoparico. Brskalnik smo pognali dvakrat, saj nas je zanimalo, koliko hitreje se zažene v drugem poizkusu.

Pogoji za izvedbo testa so bili:

- prazna stran,
- spraznjen pomnilnik brskalnika,
- izbrisani piškotki in
- ponoven zagon računalnika.



Slika 10: Hitrost zagona brskalnikov v sekundah.

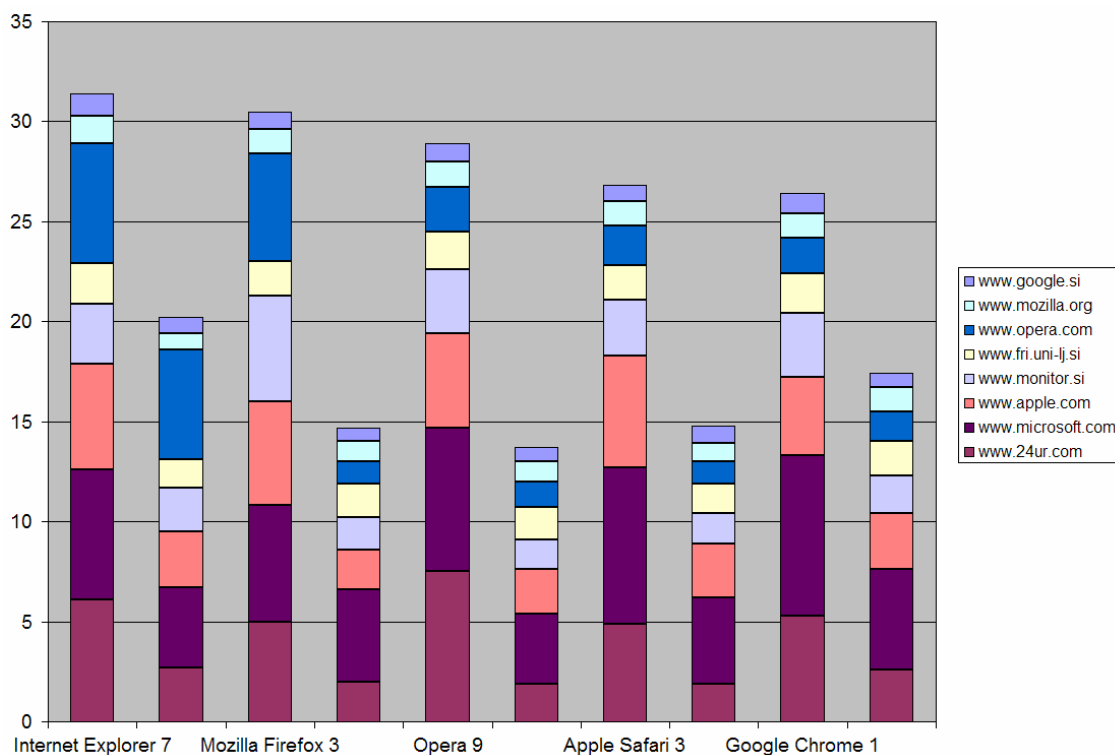
### 4.2.2. Hitrost odpiranja internetnih strani

Test smo pognali za različne internetne strani. Čas smo merili od pritiska na gumb za iskanje internetne strani dokler brskalnik ni sporočil, da je končal z izrisovanjem strani. Pred tem smo pobrisali predpomnilnik, saj si brskalniki določene stvari shranjujejo, da lahko naslednjič tudi za trikrat hitreje izrišejo stran.

Nato smo zaprli brskalnike, jih ponovno pognali in testirali, koliko hitreje so odprli internetno stran. Čase smo zapisali v tabelo 1. Vsak brskalnik v tabeli 1 ima dva stolpca. Prvi prikazuje prvi zagon internetne strani, drugi pa ponovni zagon iste internetne strani. Čas smo merili v sekundah z ročno štoparico.

Spletna stran	Internet Explorer 7		Mozilla Firefox 3		Opera 9		Safari 3		Google Chrome 1	
www.google.si	1,1	0,8	0,9	0,7	0,9	0,7	0,8	0,9	1	0,7
www.24ur.com	6,1	2,7	5	2	7,5	1,9	4,9	1,9	5,3	2,6
www.fri.uni-lj.si	2	1,4	1,7	1,7	1,9	1,6	1,7	1,5	2	1,7
www.mozilla.org	1,4	0,8	1,2	1	1,3	1	1,2	0,9	1,2	1,2
www.microsoft.com	6,5	4	5,8	4,6	7,2	3,5	7,8	4,3	8	5
www.apple.com	5,3	2,8	5,2	2	4,7	2,2	5,6	2,7	3,9	2,8
www.opera.com	6	5,5	5,4	1,1	2,2	1,3	2	1,1	1,8	1,5
www.monitor.si	3	2,2	5,3	1,6	3,2	1,5	2,8	1,5	3,2	1,9
Seštevek	31,4	20,2	30,5	14,7	28,9	13,7	26,8	14,8	26,4	17,4
Povprečna hitrost	25,8		22,6		21,3		20,8		21,9	

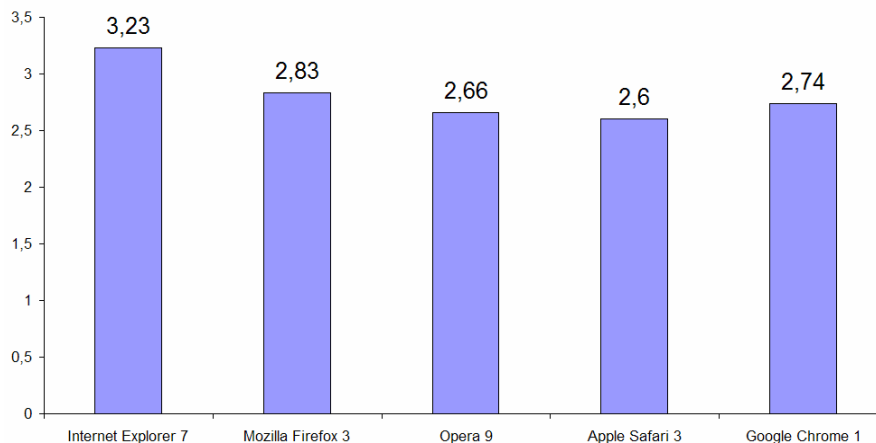
Tabela 1: Izmerjeni časi odpiranja spletnih strani za posamezen brskalniki v sekundah.



Slika 11: Hitrost zagona internetnih strani za posamezen brskalniki v sekundah.



Iz slike 11 je lepo razvidno, kateri brskalniki znajo najboljše optimizirati ponovni zagon iste internetne strani. Najbolje se odrežeta Firefox in Opera, sledi Safari, nekoliko slabše se odrežeta Explorer in Chrome.



Slika 12: Povprečna hitrost zagona internetne strani za posamezen brskalnik v sekundah.

Za povprečno hitrost zagona spletne strani smo določili, kot seštevek časov zagona posameznih internetnih strani, ki smo jih pognali prvič in nato še drugič. Rezultat delimo s 16, da dobimo povprečen čas odpiranja ene spletne strani, ki je prikazano na sliki 12. Izkaže se, da ni večjih razlik. To priča o tem, da so brskalniki v praksi podobno hitri. Če bi hoteli določiti, bi bil poraženec Explorer, zmagovalec pa Safari.

#### 4.2.3. Poraba sistemskih sredstev

Tukaj smo testirali, koliko pomnilnika posamezen brskalnik zaseda pri svojem delovanju z enim, dvema, štirimi ali desetimi odprtimi internetnimi stranmi v enem brskalniku. Za testno stran smo želeli zahtevno internetno stran polno večpredstavnih vsebin. Kot primerno stran smo si izbrali spletno stran slovenske nacionalne televizije RTV SLO (<http://www.rtv slo.si/>). Isto stran smo pognali tudi v več zavihkih. Porabo pomnilnika smo merili z upraviteljem opravil v sistemu Windows XP. Izmerjene rezultate smo zapisali v tabelo 2.

Brskalnik	Število odprtih zavihkov				
	0	1	2	4	10
Explorer 7	28.024	41.868	57.580	82.108	154.700
Firefox 3	30.680	46.396	50.632	63.096	93.520
Opera 9	20.544	36.288	43.864	61.404	92.380
Safari 3	34.780	57.932	64.260	76.104	105.400
Chrome 1	38.165	57.700	79.900	151.200	303.000

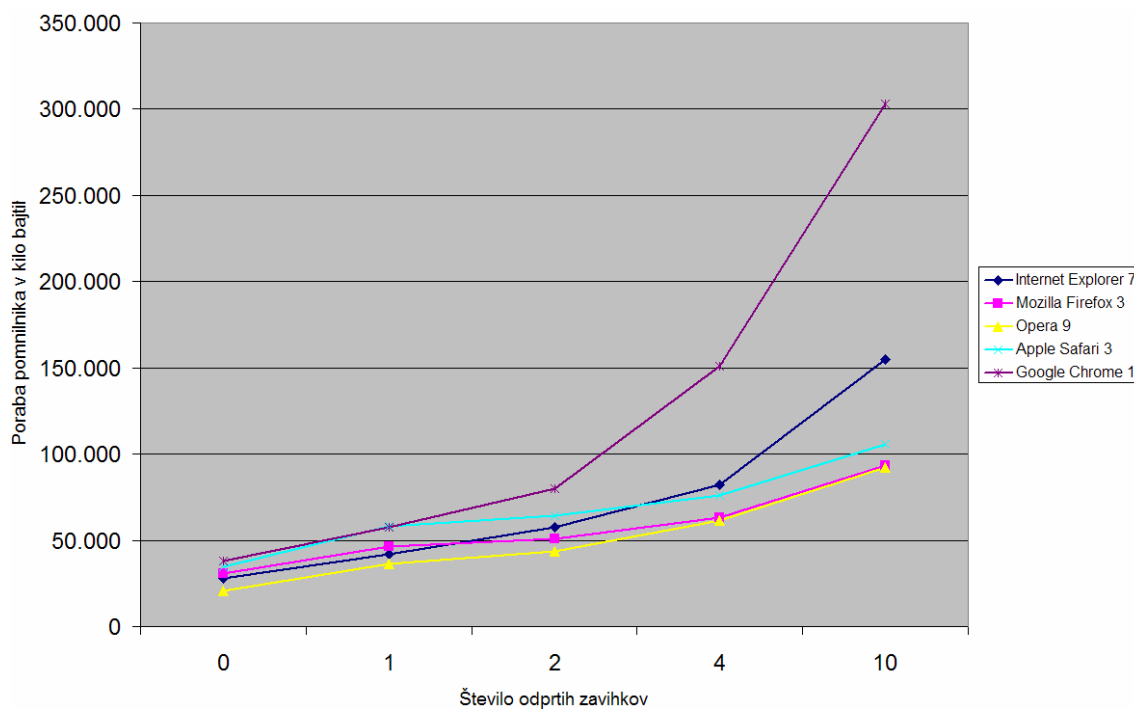
Tabela 2: Poraba pomnilnika v kilo bajtih za posamezen brskalnik.

Vsi brskalniki porabijo za svoje delovanje kar precej pomnilnika, še posebej, ko imajo odprtih več internetnih strani. Izdelovalci brskalnikov bi to najbrž opravičili z nizko ceno pomnilnika, saj imajo že najbolj osnovni računalniki vsaj gigabajt ali dva pomnilnika.

Na sliki 13 vidimo, kako narašča poraba pomnilnika z več odprtimi zavihki. Vsi brskalniki se odrežejo približno enako, ko nimajo odprte nobene strani ter eno ali dve odprti strani. Malenkostno odstopa le Chrome, ki porabi nekaj odstotkov več pomnilnika kot drugi.

Tudi pri štirih zavihkih močno odstopa le Chrome, pri desetih zavihkih pa Chrome postane zelo požrešen in se izkaže skorajda neprimeren za računalnike z manj pomnilnika. Nekoliko se poveča potreba pomnilnika tudi Explorer-ju.

Tu moramo poudariti, da je Explorer med vsemi brskalniki najstarejši, njegov naslednik Internet Explorer 8 se zna pri več odprtih zavihkih odrezati bolje. Chrom-ovo požrešnost po pomnilniku lahko opravičimo, saj je to praktično prva različica brskalnika, zato tudi pričakujemo, da bodo kasnejše verzije to pomanjkljivost odpravile.



Slika 13: Poraba pomnilnika v kilo bajtih za posamezen brskalnik.

### 4.3. Funkcije in druge zmogljivosti

V tem podpoglavju testiramo zmogljivosti brskalnikov, s katerimi se srečujemo med vsakodnevno uporabo. Testiramo naslednje zmogljivosti:

- uporabniški vmesnik,
- podporo razvijalcem,
- varnost in
- druge značilnosti, kot so razširjenost med operacijskimi sistemi.

#### 4.3.1. Uporabniški vmesnik

Pri uporabi brskalnika se najprej srečamo z njegovim uporabniškim vmesnikom, zato je zelo pomembno, da ta naredi dober vtis. V primeru, da uporabniški vmesnik ni dovolj pregleden, dostopen in funkcionalen, se lahko zgodi, da se uporabnik odloči za konkurenčen brskalnik. Izdelovalci brskalnikov se zato trudijo, da njihov brskalnik ponuja čim več novitet, saj lahko le tako pridobijo večje število uporabnikov njihovega brskalnika.



**Internet Explorer 7** se lahko pohvali z zelo preglednim in enostavnim slovenskim uporabniškim vmesnikom. Barve so nežne in prilagojene operacijskim sistemom, gumba pa hitro najdemo in prepoznamo njihovo funkcionalnost. Moti le, da si uporabnik ne mora prilagajati izgleda brskalnika, saj mu Internet Explorer 7 ne nudi zamenjave videza.

Dobro je gotovo to, da vsi njegovi gumbi in drugi gradniki zasedejo zelo malo prostora in s tem puščajo veliko območje zaslona za prikaz spletene strani. Brskanje z zavihki in okno za hitro iskanje je v tem času nuja, ki se ga zaveda tudi Explorer. V povezavi z zavihki ima Explorer zanimivo funkcijo, ki jo pri Microsoftu imenuje »hitri zavihki«. S pritiskom na ustrezen gumb, nam Explorer vse zavihke odpre kot majhna okna v enem oknu. Funkcija je uporabna, če imamo odprtih veliko zavihkov in ne moremo najti pravega, zato ga lažje najdemo kar med oprtimi okni. Explorer si zapomni vsako odprto stran, zato lahko najdemo po pomoti zaprte strani med zgodovino obiskanih strani.

Naša ocena, dobro:

- + enostaven, pregleden, hiter uporabniški vmesnik
- + slovenski jezik
- + deskanje z zavihki in hitrim iskalnikom
- + dodatne možnosti za zavihke
- + RSS prikazovalnik
- + beleženje zgodovine brskanja

Slabo:

- ne nudi zamenjave tem
- zavihka ne moremo povleči v novo okno



**Mozilla Firefox 3** že od začetka slovi po dobrem uporabniškem vmesniku. Vsebuje tako zavihke kot hitro iskanje. Na voljo je v več jezikih, tudi slovenska. Na internetu lahko dobimo več tem, ki prilagodijo izgled brskalnika našim željam. Malo moti le to, da moramo ob vsaki spremembi brskalnik ponovno pognati.

Dobrodošla je tudi možnost ogleda pred kratkim zaprtih zavihkov, saj lahko med njimi najdemo stran, ki smo jo zaprli nenamerno. Firefox nudi veliko dodatkov, ki jih lahko naložimo s spleta. Med drugimi lahko najdemo tudi nadomestek funkcije »hitri zavihki« Explorer-ja, ki jo pri Firefoxu imenujejo showcase. Posamezen zavihek lahko povlečemo tudi v novo okno brskalnika, ki pa more biti aktivno, saj se brskalnik v primeru, da povlečemo odprt zavihek na namizje ne zažene avtomatsko. Tako kot Explorer si tudi Firefox zapomni vsako odprto stran.

#### Naša ocena, dobro:

- + enostaven, pregleden, hiter uporabniški vmesnik
- + slovenski jezik
- + deskanje z zavihki in hitrim iskalnikom
- + dodatne možnosti za zavihke
- + RSS prikazovalnik
- + zamenjava teme
- + beleženje zgodovine brskanja

#### Slabo:

- nekoliko okorna namestitvev dodatkov, saj je potrebno pred uporabo novitet ponovno pognati brskalnik
- ne najboljša izvedba povleci-spusti zavihek med okni brskalnika



**Apple Safari 3** ima med vsemi brskalniki najbolj okoren uporabniški vmesnik. Sivina, ki ga obdajam, ugaja najbolj operacijskim sistemom Mac OS X, katerim je Safari v prvi vrsti tudi namenjen. Apple trdi, da se zaradi sivine uporabniškega vmesnika lažje osredotočimo na samo stran, vendar se zaradi enake barve celotnega uporabniškega vmesnika težje znajdemo. Pogrešamo tudi slovenski vmesnik.


Zavihki in hitro iskanje sicer sta prisotna, vendar brez podobnih funkcij za lažje iskanje strani med zavihki, kot jih imata Firefox in Explorer. Zanimiva je možnost povleci-spusti zavihek iz trenutnega okna Safari v novo okno in tudi obratno. S funkcijo »Customize Toolbar« si lahko zelo enostavno prilagodimo uporabniški vmesnik. Omogoča nam, da gumbke, ki jih želimo imeti v vrstici z gumbki, enostavno povlečemo iz okna z možnimi gumbi v vrstico gumbkov na način povleči-spusti. Safari tudi pomni obiskane strani.

#### Naša ocena, dobro:

- + deskanje z zavihki in hitrim iskalnikom
- + RSS prikazovalnik
- + enostavno urejanje orodnih vrstic
- + možnost povleci-spusti zavihek v novo okno brskalnika ali obratno
- + beleženje zgodovine brskanja

Slabo:

- nekoliko nestandarden uporabniški vmesnik, dolgočasne barve
- ni dodatnih možnosti za delo z zavihki
- ni slovenskega vmesnika
- ne nudi zamenjave tem

 **Opera 9** ima dober uporabniški vmesnik, ki pa žal ni preveden v Slovenščino. Seveda nam ponuja brskanje z zavihki in hitro iskanje. Namesto funkcije »hitri zavihki«, nam pomaga pri iskanju po odprtih straneh funkcija, ki ob zdrsu miške nad zavihke odpre pod njim minimalno okno z vsebino zavihka. Opera ima tako kot Safari tudi možnosti povleci-spusti zavihke med okni.


Pri Operi pogrešamo odpiranje spletnih strani iz mape priljubljenih v novem zavihku s pomočjo srednjega miškega gumba. Opera 9 ima možnost hitre in enostavne zamenjave izgleda brskalnika. Na spletu je veliko tem, ki nam spremenijo izgled brskalnika po naši volji. Teme brskalnik poišče celo sam in jih uredi po priljubljenosti. Samo klik na temo je dovolj, da zamenjamo izgled brskalnika. Brskalnika pri tem ni potrebno ponovno pognati. Že vsakemu se je kdaj zgodilo, da je po pomoti zaprl zavihke, ki ga ni hotel. Opera 9 ima za to rešitev. Vsak zavihke, ki ga zapremo, ga začasno shrani v koš z zavihki. Če smo ga po pomoti zaprli, ga lahko najdemo v košu, ki začasno shranjuje zaprte zavihke. V košu so shranjeni zavihki tako dolgo, dokler ne zapremo brskalnika. Vse zaprte strani beleži v mapi zgodovine.

Naša ocena, dobro:

- + enostaven, pregleden, hiter uporabniški vmesnik
- + deskanje z zavihki in hitrim iskanikom
- + dodatne možnosti za zavihke
- + RSS prikazovalnik
- + zamenjava teme
- + možnost povleci-spusti zavihke v novo okno brskalnika ali obratno
- + beleženje zgodovine brskanja
- + koš za zaprte zavihke

Slabo:

- ni slovenskega vmesnika
- stran iz zaznamkov ne moremo odpreti s srednjim miškinim gumbom v novem zavihku

 **Google Chrome 1** je med vsemi testiranimi najnovejši brskalnik, zato ima po pričakovanjih največ novitet med uporabniškimi funkcijami, ki pa se jih uporabnik težko navadi.

Prva noviteta, ki jo srečamo, je združeno okno za vpis naslova spletne strani z oknom za hitro iskanje. Če imamo kot privzeto stran nastavljen iskalnik Google, se lahko zaradi združitve zgornjih dveh oken odredimo tako gumbu za domačo stran kot tudi sami privzeti strani, saj je zaradi združitve vedno pri roki.

Vmesnik je bil že v začetku v Slovenščini, kar je pohvalno. Je zelo minimiziran in omogoča največje okno za prikaz spletne strani med vsemi testiranimi brskalniki. Zavihki so postavljeni v zgornjo vrstico, katera je pri ostalih brskalnikih prazna. S tem so se pri Googlu znebili ene odvečne vrstice za zavihke. Brskalnik je predvsem za začetnike nekoliko okoren za uporabo, saj ima nekoliko nestandardno porazdeljene gumbe in okna. Zelo zanimiva noviteta je ta, da je vsak odprt zavihkec svoj proces v sistemu, kar v praksi pomeni, da odzivnost določene spletne strani ne vpliva na celoten brskalnik, ampak samo na zavihkec, ki hrani odprto stran. Tako zgubimo samo eno odprto stran in ne vseh odprtih strani. Tudi Chrom si zapomni vsako stran, ki smo jo obiskali.

#### Naša ocena, dobro:

- + deskanje z zavihki in hitrim iskalnikom
- + RSS prikazovalnik
- + možnost povleci-spusti zavihkec v novo okno brskalnika ali obratno
- + slovenski vmesnik
- + zelo robusten brskalnik
- + beleženje zgodovine brskanja

#### Slabo:


- ni dodatnih možnosti za delo z zavihki
- za enkrat še nekoliko okoren uporabniški vmesnik

#### Sklep:

Za konec lahko ugotovimo, da imajo vsi brskalniki dovolj kvaliteten uporabniški vmesnik. Nekoliko bolje se odrežeta Opera in Firefox, pri katerima smo našli največ dobrih ocen, slabše pa Safari in Chrome, ki sta imela najslabše razmerje med dobrimi in slabimi ocenami.

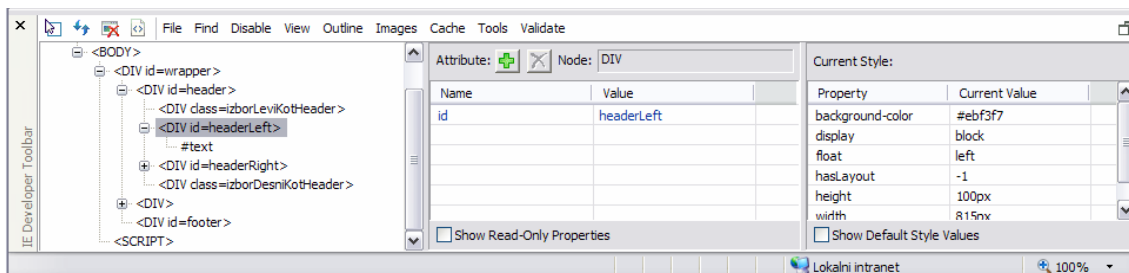
### 4.3.2. Podpora razvijalcem

Pri podpori razvijalcem smo testirali, kaj brskalniki ponujajo razvijalcem v pomoč pri razvijanju spletnih aplikacij. Iskali smo dodatke in funkcionalnosti, ki razvijalcem olajšajo delo pri programiranju JavaScripta, pisanju HTML-ja in stilskim oblikovanjem strani s pomočjo CSS-ja.

 **Internet Explorer 7** ima podporo za razvijalce. Najdemo jo v obliki dodatka z imenom Developer Toolbar. Internet Explorer Developer Toolbar predvideva več možnosti za raziskovanje in razumevanje spletnih strani.

Del Developer Toolbara, DOM Explorer, omogoča popoln vizualen pregled nad dokumentnim modelom objektov HTML strani. Drevesna struktura omogoča hierarhičen pregled nad elementi strani, trenutno izbran element pa je v oknu brskalnika ustrezno označen, prav tako so v tabeli prikazane njegove lastnosti. Objektu lahko spremenimo posamezen atribut ali stil, posledice spremembe pa so nemudoma vidne v oknu brskalnika. Tako ni več zamudnega popravljanja kode in nato ponovnega osveževanja strani. Spremenimo lahko lastnosti in nemudoma preverimo dejansko spremembo v prikazani strani in šele nato popravimo kodo. Ker je iskanje pravega elementa v bolj kompleksnih spletnih

strane težavno opravilo, nam DOM Explorer omogoči, da element, ki bi ga radi podrobneje pregledali oz. analizirali, preprosto označimo na prikazani spletni strani. Zanimiva je tudi tako imenovana Outline funkcija, ki nam na strani označi npr. vse tabele, slike, div elemente ali individualno določene elemente. Tu je tudi funkcija Validate, ki preveri veljavnost strukture HTML, CSS ali drugih deklaracij glede na potrjene standarde W3C organizacije. Na sliki 14 skrajno levo vidimo drevesno strukturo HTML dokumenta, na sredini so atributi elementa <div>, ki smo ga označili, skrajno desno pa njegova stilska predloga. [26]




Slika 14: Internet Explorer 7 Developer Toolbar.

#### Naša ocena, dobro:

- + pregledovanje DOM-a, JavaScript-a, CSS-ja
- + dinamično spreminjanje atributov in njihovih vrednosti
- + enostavno iskanje elementov v DOM-u
- + enostavno iskanje elementov na strani s pomočjo miške
- + ni opazno, da bi se zaradi dodatkov hitrost brskalnika zmanjšala

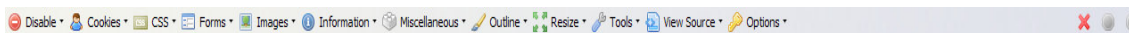
#### Slabo:

- ni dinamičnega urejanja CSS-ja
- slabo implementirana validacija DOM-a, JavaScript-a, CSS-ja
- ni bližnjic v orodni vrstici

 **Mozilla Firefox 3** skupaj z dodatki, ki so na voljo na njihovi strani, nudi zelo močno razvojno orodje. Lahko rečemo, da je Firefox na tem področju med najmočnejšimi, če ne celo najmočnejši brskalniki. Tu lahko omenimo tri dodatke, ki olajšajo delo razvijalcem:

- Web Developer

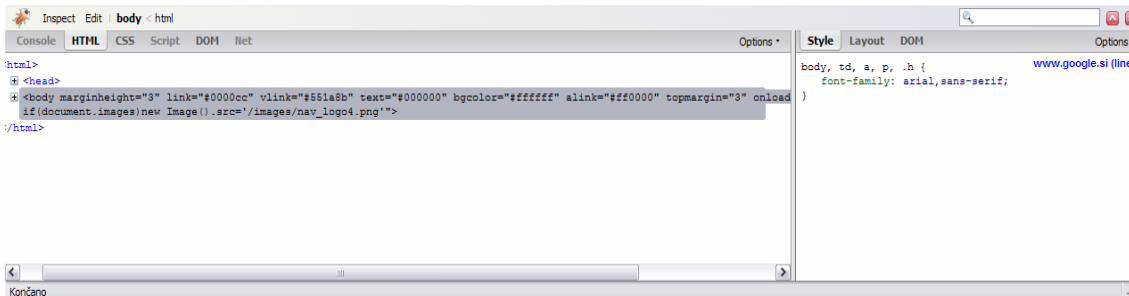
To je orodna vrstica (slika 15), s pomočjo katere si lahko ogledamo zgradbo spletne strani. Je zelo pregledna in lahko dostopna, kar olajša delo tudi novim uporabnikom. Z njo lahko zahtevamo natančen izris ali označbo HTML elementov, ki nas zanimajo (npr. slike). S pomočjo orodja, lahko prikazujemo stran v različnih zaslonskih resolucijah in s tem testiramo kako se bo naša spletna stran obnašala na različnih monitorjih. S klikom na HTML element vidimo natančen opis njegovih atributov, postavitev, vrsto pisav, velikosti itd. Omogočeno nam je izklapljanje in vklepanje pomnilnika in piškotkov, ki jih brskalnik in stran uporabljata pri svojem delu. Ogledamo si lahko tudi celoten CSS spletne strani. V desnem kotu vrstice Web Developer-ja je kljukica, ki je obarvana zeleno, če se je stran izrisala brez napake, oziroma rdeče, če ima spletna stran napako. V primeru napake nam Web Developer pokaže, kje in kakšno napako imamo. [27]



Slika 15: Mozilla Firefox 3 Web Developer Toolbar.

- Firebug

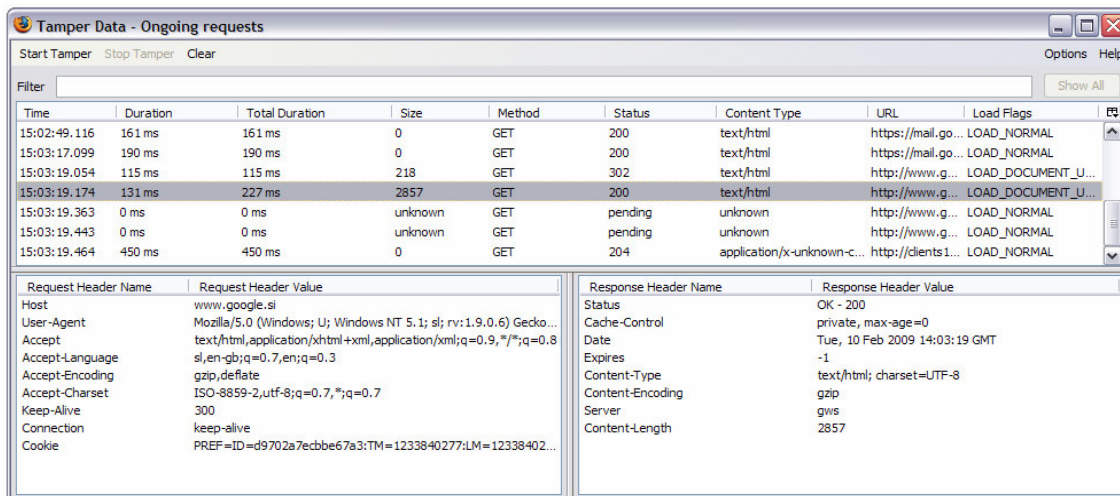
Firebug (slika 16) omogoča ogled, spremljanje in urejanje HTML-ja, CSS-ja in JavaScript-a v realnem času. Drugače povedano, s pomočjo Firebug-a lahko uredimo spletno aplikacijo direktno iz brskalnika. S tem ne spreminjamo fizične datoteke na disku, ampak samo stran. To nam pomaga hitreje poiskati napako ali dokončati izgled naše spletne strani. Tu je tudi razhroščevalnik za JavaScript in lovilec napak. [28]



Slika 16: Mozilla Firefox 3 Firebug.

- Tamper Data

S Tamper Data (slika 17) si lahko ogledamo in spreminjamo HTTP / HTTPS glave in parametre. Sledimo in merimo odzive HTTP zahtevkov in odgovorov. Omogoča tudi varnostni test spletne aplikacije s spreminjanjem POST parametrov. [29]



Slika 17: Mozilla Firefox 3 Tamper Data.



Naša ocena, dobro:

- + pregledovanje DOM-a, JavaScript-a, CSS-ja
- + dinamično spreminjanje atributov in njihovih vrednosti
- + enostavno iskanje elementov v DOM-u
- + enostavno iskanje elementov na strani s pomočjo miške
- + dinamično urejanje CSS-ja
- + dobro implementirana validacija DOM-a, JavaScript-a, CSS-ja
- + bližnjice v orodni vrstici

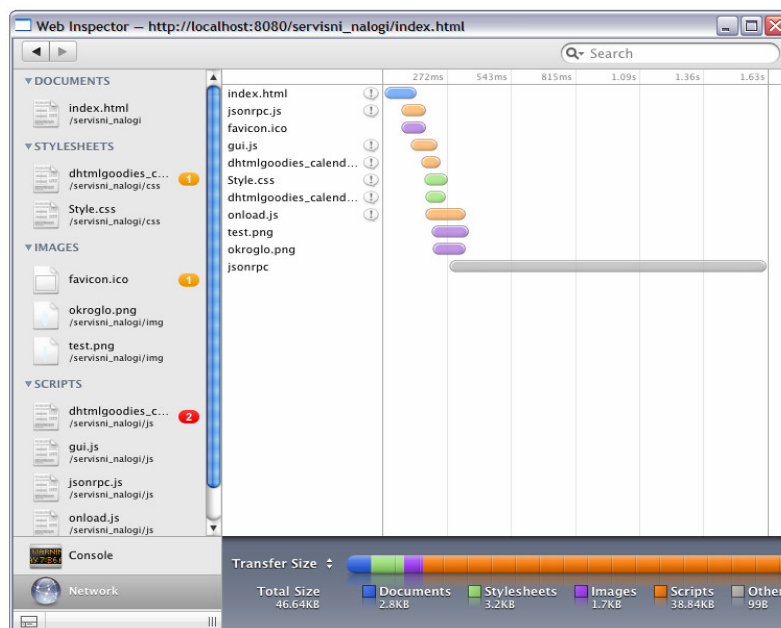
Slabo:

- zaradi dodatkov se zmanjšuje hitrost brskalnika



**Apple Safari 3** že vsebuje orodje za pomoč spletnim razvijalcem (slika 18), zato ga ni potrebno dodatno namestiti kot moramo to storiti pri Firefox-u in Explorer-ju. Pri Safari-ju ga je potrebno samo prikazati. To storimo tako, da gremo v meni »edit« in izberemo »preferences«. V zavihku »advance« obkljukamo »Show Develop menu in menu bar«.

Safari ponuja podobne možnosti kot jih ponujata Explorer in Firefox. Tako lahko tudi tu izklopimo prikaz določenih HTML elementov, izklopimo pomnilnik brskalnika itd. Zelo uporabno se zdi lovljenje napak, ki je izvedeno malo bolje kot pri Explorer-ju in Firefox-u, saj ulovljene napake združi po skupinah, npr. JavaScript napake, HTML napake itd. Zanimiv je prikaz izmerjenega časa pri nalaganju strani, saj nam čas predstavi v grafih za vsako naloženo datoteko posebej. Tako lahko vidimo, katere dele spletne aplikacije ali strani moramo dodatno optimizirati. [30]



Slika 18: Apple Safari 3 Developer.

Naša ocena, dobro:

- + pregledovanje DOM-a, JavaScript-a, CSS-ja
- + dinamično spreminjanje atributov in njihovih vrednosti
- + enostavno iskanje elementov v DOM-u
- + dinamično urejanje CSS-ja
- + dobro implementirana validacija DOM-a, JavaScript-a, CSS-ja
- + ni opazno, da bi se zaradi dodatkov hitrost brskalnika zmanjšala
- + nekatere dodatne funkcije kot so merjenje časa nalaganja posameznih delov strani

Slabo:

- ni bližnjic v orodni vrstici
- ni enostavnega iskanje elementov na strani s pomočjo miške

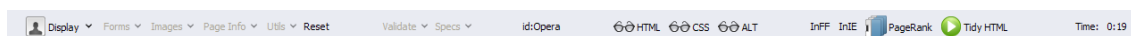


**Opera 9** že vsebuje nekaj orodij za razvijalce, na voljo pa so tudi dodatna orodja. Najdemo jih v zavihku »Tools« kjer izberemo »Advance«. Tu lahko najdemo Developer Tools (slika 20) in Error Console.

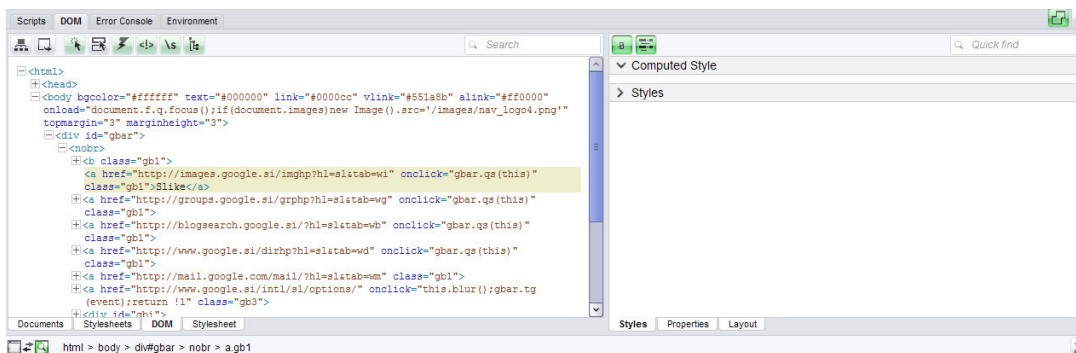
Z Developer Tools-i si pomagamo pri programiranju JavaScripta, saj lahko pregledujemo kodo po vrsticah, tu je še razhroščevalnik. Pregledujemo lahko DOM spletne strani, ga dinamično spreminjamo in urejamo ter iščemo elemente, ki jih najdemo v drevesnem prikazu elementov na strani. Tu je konzola, ki nas obvešča o napakah.

Dinamično spreminjamo CSS naše strani in jo s tem lažje in hitreje oblikujemo. Podrobnosti o napakah si pogledamo s pomočjo Error Console, kjer lahko napake tudi grupiramo po vrstah, npr HTML, JavaScript, CSS, itd. Grupiramo jih lahko tudi po težavnosti stopnji; obvestilo, opozorilo in napaka.

Tu je še orodna vrstica za razvijalce (slika 19), ki podobno kot pri Firefox-u nekoliko olajša delo, saj so opcije in funkcije, ki jih brskalnik nudi razvijalcem, logično grupirane in tako tudi lažje za uporabo.



Slika 19: Opera 9 Developer orodna vrstica.




Slika 20: Opera 9 Developer Tools.

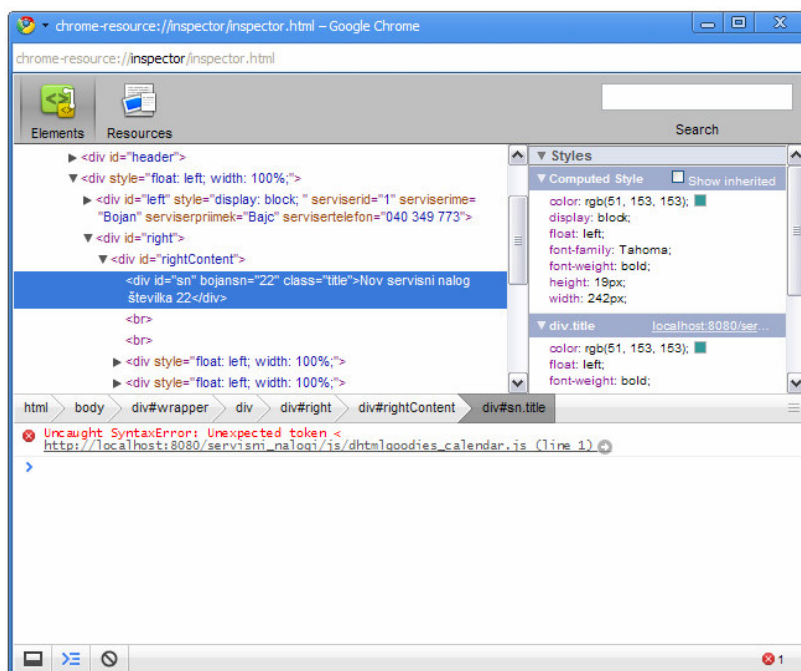
Naša ocena, dobro:

- + pregledovanje DOM-a, JavaScript-a, CSS-ja
- + dinamično spreminjanje atributov in njihovih vrednosti
- + enostavno iskanje elementov v DOM-u
- + dinamično urejanje CSS-ja
- + dobro implementirana validacija DOM-a, JavaScript-a, CSS-ja
- + nekatere dodatne funkcije kot so merjenje časa nalaganja celotne strani
- + bližnjice v orodni vrstici

Slabo:

- ni enostavnega iskanje elementov na strani s pomočjo miške
- zaradi dodatkov se hitrost brskalnika nekoliko zmanjšala

 **Google Chrome 1** ima že vgrajeno podporo za razvijalce (slika 21). Podpira praktično vse kar podpira konkurenca. Tukaj je lovljenje napak JavaScripta, HTML-ja, CSS-ja in drugih. Lahko iščemo elemente preko DOM drevesa, dinamično urejamo HTML, JavaScript in CSS.



Slika 21: Google Chrome Rzvijalec.

Naša ocena, dobro:

- + pregledovanje DOM-a, JavaScript-a, CSS-ja
- + dinamično spreminjanje atributov in njihovih vrednosti
- + enostavno iskanje elementov v DOM-u
- + dinamično urejanje CSS-ja
- + dobro implementirana validacija DOM-a, JavaScript-a, CSS-ja
- + nekatere dodatne funkcije kot so merjenje časa nalaganja celotne strani
- + ni opazno, da bi se zaradi dodatkov hitrost brskalnika zmanjšala

Slabo:


- bližnjice v orodni vrstici
- ni enostavnega iskanje elementov na strani s pomočjo miške
- Chrome Developer se odpira v novem oknu, kar je nekoliko moteče in zamudno za razvijalca

Sklep:

Firefox je zmagovalec tega pod testa, saj vsebuje največ pomoči spletnim razvijalcem, ob tem pa ga krasi še kvaliteten in pregleden uporabniški vmesnik. Dobro so izkažeta še Opera in Safari, ki vsebujeta nekaj uporabnik novitet, kot je npr. grupiranje ulovljenih napak glede na vrsto napake. Pri njima tako kot pri Chromu pogrešamo iskanje elementa na strani s pomočjo miške. Najslabše se odrežeta Explorer in Chrome, prvi zaradi nekoliko zastarelega orodja za pomoč razvijalcem, drugi pa zaradi neposrečene novitete odpiranja orodij za pomoč razvijalcem v novem oknu.

**4.3.3. Varnost**

V tem podpoglavju smo testirali, kakšno podporo nam brskalniki nudijo na področju varnosti. Testirali smo, kako se odrežejo pri blokiranju pojavnih oken in nezaželenih programov ter kakšna je njihova podpora elektronskemu poslovanju. Posamezni brskalniki so presenetili s samodejnim posodabljanjem in povezovanjem z naloženim protivirusnim programom, kar smo šteli kot plus.

 **Internet Explorer 7** se je glede na svojega predhodnika na področju varnosti izboljšal. V prvi vrsti vsebuje filtriranje lažnih spletnih mest. To so internetne strani, ki se izkazujejo za stran, ki to niso, da bi se lahko nepridipravi dokopali do naših zaupnih podatkov in jih izrabili proti nam. Sem sodijo lažna, goljufiva in ponarejena spletna mesta. Dalje Explorer preprečuje, da bi se na računalnik samodejno naložile določene datoteke ali programi, preprečuje tudi samodejni zagon programa. S tem nas reši samodejnega nalaganja raznih virusov in druge zlonamerne programske opreme.

Tu je tudi blokiranje pojavnih oken. To so okna, ki se odprejo ob vstopu na spletno stran, navadno še preden se stran naloži. Ponavadi nosijo reklamna sporočila, lahko pa tudi nevarne povezave do okuženih spletnih mest.

Med brskanjem po spletu si uporabnik velikokrat naloži razne dodatke za svoj brskalnik, ki služijo kot pomoč pri brskanju. To so navadno dodatne orodne vrstice z razno večpredstavnostno vsebino ali animacijo. Navadno to uporabniku izboljša kakovost deskanja po spletu, ne pa v vseh primerih. Nekateri dodatki lahko povzročajo ne odzivanje računalnika ali prikazujejo neprimerno vsebino. V takih primerih lahko z Explorer-jem odstranimo prikazovanje dodatkov, za katere ocenjujemo, da nas motijo ali so nam nevarni. Tu je tudi podpora spletnim certifikatom ali digitalnim potrdilom, ki jih pa žal ne zaščitijo pred uvozom ali izvozom.


128-bitna varna povezava (SSL) za uporabo varnih spletnih mest omogoča vzpostavljajanje šifrirane povezave s spletnimi mesti bank, spletnih prodajaln, zdravstvenih mest in drugih organizacij, ki ravnaajo z zaupnimi podatki strank. Žal ima med vsemi testiranimi največ odkritih varnostnih lukenj, glede na predhodnike pa na tem področju napreduje. [31, 32]

Naša ocena, prednosti:

- + filtriranje lažnih spletnih mest
- + blokiranje samodejnega nalaganja programov in virusov
- + blokiranje nezaželenih oken
- + blokiranje škodljivih dodatkov
- + podpora certifikatom
- + varne šifrirane povezave
- + samodejno posodabljanje

Slabosti:

- slaba zaščita certifikatov
- največ odkritih varnostnih lukenj

 **Mozilla Firefox 3** zna zaznati lažna spletna mesta in spletna mesta, ki nam želijo naložiti zlonamerno programsko opremo. Kot dokaz so izdelali dve strani, s katerima lahko preverimo delovanje brskalnikov na področju blokiranja lažnih in zlonamernih spletnih mest. Brskalnik lahko testiramo za lažna spletna mesta na naslovu <http://www.mozilla.com/firefox/its-a-trap.html>, za zlonamerna spletna mesta, ki nam želijo namestiti zlonamerno programsko opremo pa na naslovu <http://www.mozilla.com/firefox/its-an-attack.html>.

Zelo elegantno se lahko rešimo zasebnih podatkov. Počistimo jih z klikom na »počisti zasebne podatke« ali kar z kombinacijo tipk ctrl + shift + del. Pri Firefox-u je po privzetih nastavitvah nastavljeno samodejno nalaganje popravkov, kar zagotavlja, da ima naš brskalnik odpravljene vse napake, ki so jih odkrili. Tako kot Explorer blokira odpiranje nezaželenih, predvsem reklamnih oken. Po našem testu to počne nekoliko slabše kot Explorer. Podporo ima tudi za certifikate. Ob namestitvi brskalnik elegantno preišče računalnik za morebiten protivirusni program in se poveže z njim. Tako vsako datoteko, ki jo prenesemo na računalnik, pregleda protivirusni program. [31, 33]

Naša ocena, dobro:

- + filtriranje lažnih spletnih mest
- + blokiranje samodejnega nalaganja programov in virusov
- + podpora certifikatom
- + varne šifrirane povezave
- + dobra zaščita certifikatov z varnostnim ključem
- + brskalnik se poveže z nameščenim protivirusnim programom
- + ni odkritih varnostnih lukenj
- + samodejno posodabljanje

Slabo:

- slabše blokiranje nezaželenih oken



**Apple Safari 3** je tudi dobro poskrbel za zasebnost. En pritisk na gumb »Private browsing« in že bo brskalnik vsako geslo, stran ali iskalni niz, ki smo ga vpisali, pozabljal. Vsebuje podporo za blokiranje ponarejenih in zlonamernih mest, blokira pojavna reklamna okna, tu je tudi podpora za certifikate.

Pravijo, da je najbolj varen brskalnik. To je najbrž podedoval od operacijskega sistema Mac, katerega privzet brskalnik je. Vendar testi kažejo, da le ima nekaj varnostnih lukenj.

Podobno kot Firefox se tudi Safari samodejno posodablja in s tem sam skrbi, da smo redno deležni vseh varnostnih in drugih popravkov, ki skrbijo, da je naš brskalnik v koraku s časom in da je odporen na vse možne napade. [31]

#### Naša ocena, dobro:

- + filtriranje lažnih spletnih mest
- + blokiranje samodejnega nalaganja programov in virusov
- + podpora certifikatom
- + varne šifrirane povezave
- + samodejno posodabljanje
- + privatno deskanje

#### Slabo:

- nekaj odkritih varnostnih lukenj



**Opera 9** ve kaj je varnost in zasebnost. Tako kot Firefox vsebuje možnost brisanja vseh privatnih podatkov kot so piškotki, shranjena gesla, zgodovina obiskanih strani in drugo s pritiskom na gumb »Tools« in nato »Delete Private Data«.

Svoja shranjena gesla lahko dodatno zaščitimo še s tako imenovanim glavnim geslom. To prepreči dešifriranje naših gesel ali ključev s strani zlonamernih programov ali storilcev, saj morajo poznati glavno geslo, preden se lahko lotijo dešifriranja drugih ključev.

Zelo dobro deluje varnostni vmesnik za zaznavanje lažnih in zlonamernih spletnih mest, in sicer nekoliko boljše kot Explorer. V primerjavi z Firefox-om bolje blokira pojavna reklamna okna.

Za vzpostavitev varnih povezav uporablja SSL ali TSL protokol, ki sta varna protokola za šifriranje varnih povezav, podobno kot Explorer. [20, 21, 31]

#### Naša ocena, dobro:

- + filtriranje lažnih spletnih mest
- + blokiranje samodejnega nalaganja programov in virusov
- + podpora certifikatom
- + varne šifrirane povezave
- + samodejno posodabljanje
- + dobra zaščita certifikatov z varnostnim ključem
- + ni odkritih varnostnih lukenj



**Google Chrome** šifrira varne povezave s 168 bitno kodo. Očitati mu moramo nekoliko nadležno delo s certifikati. Podpira sicer jih, samo v praksi stvar nekako ne deluje najbolje. Pri blokiranju lažnih in goljufivih spletnih mest je najslabši med testiranimi. Poskrbljeno je le za blokiranje odpiranja reklamnih in drugih oken. Podobno kot pri porabi pomnilnika ga lahko tudi tu opravičujemo s tem, da gre za novinca med brskalniki. [31]

#### Naša ocena, dobro:

- + filtriranje lažnih spletnih mest
- + blokiranje samodejnega nalaganja programov in virusov
- + blokiranje škodljivih dodatkov
- + varne šifrirane povezave
- + samodejno posodabljanje

#### Slabo:

- nekaj odkritih varnostnih lukenj
- za enkrat podpora certifikatom še ne deluje najbolje

#### Sklep:

Absolutni zmagovalec tega testa je Opera, saj pri testu nismo našli nobene pomanjkljivosti. Sledita ji Firefox in Safari, ki nimata kritičnih pomanjkljivosti, najslabše pa se odrežeta Chrome in Explorer, predvsem zaradi slabe podpore certifikatom.

#### **4.3.4. Druge značilnosti**

Brskalnik	Operacijski sistem			Slovenščina	Vir
	Windows	Linux	Mac OS X		
Internet Explorer	da	ne	ne	da	[18]
Mozilla Firefox	da	da	da	da	[19]
Google Chrome	da	v izdelavi	v izdelavi	da	[23]
Opera	da	da	da	ne	[21]
Safari	da	neuradno	da	ne	[22]

*Tabela 3: Podprtost brskalnikov operacijskim sistemom in slovenskemu vmesniku.*

Trenutna verzija Internet Explorer-ja je dosegljiva le v okolju Windows. Za operacijska sistema Unix in Mac OS X pa je bil dosegljiv nazadnje v verziji 5.0. Vsi brskalniki so dosegljivi v okolju Windows. V sistemih Linux pa delujeta le Opera in Firefox, v Mac OS X pa Firefox, Opera in seveda Safari. [18, 22]

## 4.4. Podpora standardom

V podpoglavju smo testirali, kakšno podporo nudijo brskalniki spletnim standardom. Veliko testov je namenjeno testiranju CSS standarda in JavaScripta. Ogledamo si tudi podporo glavnih W3C standardov, na katerih je osnovan splet.

### 4.4.1. W3C

V drugem poglavju smo opisali nekaj standardov in tehnologij organizacije W3C, vendar ne vseh. Tu si bomo pogledali, katere izmed njih brskalniki podpirajo (ocena 1), delno podpirajo (ocena 0.5) ali ne podpirajo (ocena 0). Za veliko standardov ni znano, kako jih brskalniki podpirajo, zato zapišemo kot rezultat vprašaj in ga v končnem testu ne upoštevamo.

	Internet Explorer 7	Mozilla Firefox 3	Opera 9	Apple Safari 3	Google Chrome 1
CSS 2.1	0,5	1	1	1	1
CGI	?	?	?	?	?
DOM	0,5	1	1	1	1
GRDDL	?	?	?	?	?
HTML	1	1	1	1	1
OWL	?	?	?	?	?
RDF	?	?	?	?	?
SVG	0	0,5	0,5	0,5	0,5
SISR	?	?	?	?	?
SOAP	?	?	?	?	?
SPARQL	?	?	?	?	?
SMIL	?	?	?	?	?
SRGS	?	?	?	?	?
SSML	?	?	?	?	?
VoiceXML	0	0	1	0	0
XHTML+Voice	?	?	?	?	?
WCAG	?	?	?	?	?
WSDL	?	?	?	?	?
XACML	?	?	?	?	?
XHTML	0	1	1	1	1
XML	1	1	1	1	1
XML Events	?	?	?	?	?
XForms	0	1	0	0	0
XML Information Set	?	?	?	?	?
XML Schema	?	?	?	?	?
XPath	1	1	1	1	?
XQuery	?	?	?	?	?



XSLT	1	1	1	1	1
Skupaj podprtih	5	8,5	8,5	7,5	6,5

Tabela 4: Podpora W3C standardom posameznega brskalnika. [31]

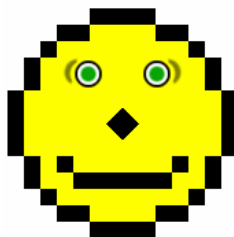
#### 4.4.2. Acid2

Acid2 je test v obliki spletne strani, ki testira spletne brskalnike. Izdelan je bil kot pomoč izdelovalcem spletnih brskalnikov. S pomočjo Acid2 testa si spletni razvijalci zagotovijo, da njihovi brskalniki pravilno podpirajo vse posebnosti in zmogljivosti, ki bi jih od brskalnikov lahko zahtevali spletni razvijalci.

Acid2 je zelo kompleksna stran, natrpana z veliko testi. Acid2 testira le obstoječe standarde, ki so bili napačno podprti od večine brskalnikov. Test na zelo enostaven način pokaže uporabnikom in razvijalcem spletnih brskalnikov, če njihov brskalnik pravilno podpira vse standarde. Če jih, se izriše »smeško« (slika 22) na levi strani zaslona. V primeru, da jih ne, se smeško ne izriše pravilno, oziroma se delno izriše v rdeči barvi.

Pogoj za začetek testa so privzete nastavitve brskalnika, kot so recimo privzeta velikost pisave, privzeta povečava strani itd.

Hello World!



Slika 22: Acid2 test.

Acid2 test uporablja štiri spletne standarde:

- **HTML4**  
Z razliko od predhodnikov, HTML4 podpira več večpredstavnostnih vsebin, skriptnih jezikov, stilskih predlog in dokumentov, ki so lažje dostopni uporabnikom s posebnimi potrebami, kot so npr invalidi.
- **CSS1**  
To so stilske predloge stopnje ena, ki predvidevajo enostavni stilski mehanizem, ki ga avtorji spletnih strani dodajo spletni strani. Avtor lahko predvidi privzet stil, medtem ko si ga lahko uporabnik prilagodi zaradi njegovih tehničnih ali drugih potreb.
- **PNG**  
Prenosljiva spletna grafika (ang. Portable Network Graphics) je rastrski slikovni format.

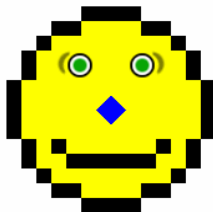
- **Data URLs**  
To je nova URL shema, ki ji pravimo »data«. Omogoča vključevanje manjših »data« podatkov kot takojšnje podatke, kot bi bili vključeni navzven. Nekatere aplikacije imajo potrebo po taki storitvi, kot je vključevanje slike v vrstici.  
Sintaksa: `data:[<mediatype>][;<base64>],<data>`  
<mediatype> zahteva internetno medijsko vrsto (tekst, slika, video), base64 pomeni, da so podatki kodirani. V <data> vpišemo URL internetnega medija. Podatki, ki se pretakajo po internetu s tem standardom, veliko manj obremenjujejo internetno povezavo.

Povzetek zmogljivosti, ki jih Acid2 testira:

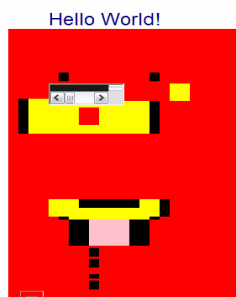
- prosojnost PNG-ja (oči so kodirane kod prosojni PNG-ji).
- uporaba <object> HTML elementa. Razvijalci so si že od nekdaj želeli element, ki ima lahko različno vsebino. Oči smeškota so pritrjene na ta element.
- absolutno, relativno in fiksno postavljanje elementov.
- »box model«. Gre za testiranje atributov »height«, »width«, »max-width«, »min-width«, »max-height« in »min-height«.
- nastavljanje pozicij in izgleda elementov na strani s pomočjo tabel (<table> element).
- lastnost HTML elementov »margin«. CSS natančno določa, kako se mora ta lastnost računati.
- CSS razčlenjevanje. Acid2 vsebuje vrsto neveljavnih CSS stavkov, ki jih morajo brskalniki ignorirati.
- testiranje večkrat pobarvane vsebine ali je pobarvana v pravem vrstnem redu. Ta test je narejene zaradi drugih zmogljivosti, ki rabijo za svoje delovanje pravilen vrstni red barvanja.
- »Hovering« efekt (ko se z miško premaknemo čez nos smeškota, se ta obarva modro). [34]

Vsi testirani brskalniki razen Internet Explorer-ja so test opravili z odliko. Pravilno opravljen test vidimo na sliki 23, neopravljen test pa na sliki 24. Acid2 lahko opravimo na naslovu: <http://www.webstandards.org/files/acid2/test.html>.

Hello World!



*Slika 23: Pravilno izrisan smeško Acid2.*

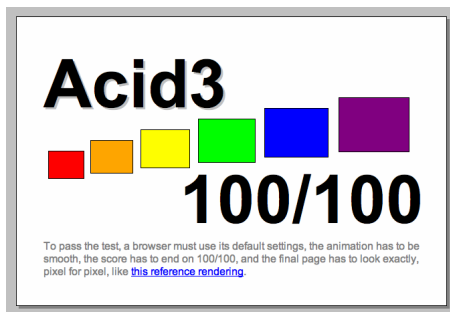


Slika 24: Nepravilno izrisan smeško Acid2.

### 4.4.3. Acid3

Acid3 (slika 25) je tretji test, ki je bil izdelan takoj, ko je večina brskalnikov opravila Acid2 test. Ravno tako kot predhodnik je izdelan kot pomoč razvijalcem in uporabnikom brskalnikov, da preverijo, če njihov brskalnik podpira zadnje tehnologije in standarde spleta. Testa v popolnosti ne opravi noben od preverjenih brskalnikov. To priča o tem, kako zelo težaven in zahteven je test.

Test prestane le brskalnik, ki izriše pravokotnike v pravih barvah do pike natančno in opravi vseh 100 podtestov. To še ni dovolj, animacija more bit gladka in ne sme porabiti več kot 33 milisekund za posamezen podtest. Tudi tu so pogoji za začetek testa: privzete nastavitve brskalnika, kot so privzeta velikost pisave, privzeta povečava strani itd.



Slika 25: Acid3 test.

Acid3 test je v večini napisan s pomočjo ECMAScript-a (JavaScript) in vsebuje 100 podtestov v šestih skupinah imenovanih »vedra«. Vedra imajo skupaj 96 testov, posamezno »vedro« pa se kaže kot pravokotnik na testu. Ostali testi 0, 97, 98 in 99 so posebni.

Posamezna vedra preverjajo:

- Vedro 1: DOM Traversal, DOM Range, HTTP
- Vedro 2: DOM2 in DOM2 dogodke
- Vedro 3: DOM2 poglede, DOM2 stil, CSS3 selektorje in medija poizvedbe
- Vedro 4: Obnašanje HTML-ja, ko urejamo tabele in forme s pomočjo skript in DOM2 HTML
- Vedro 5: Testi Acid3 konkurence (SVG, HTML, SMIL, Unicode, ...)
- Vedro 6: ECMAScript

Barva posameznega pravokotnika v testu določa koliko pod testov je brskalnik opravil v posameznem »vedru«:

- Če brskalnik ne opravi nobenega podtesta v vedru, se pravokotnik ne prikaže.
- 1-5 opravljenih pod testov: črn pravokotnik.
- 6-10 opravljenih pod testov: siv pravokotnik.
- 11-15 opravljenih pod testov: srebrn pravokotnik.
- Vseh 16 opravljenih pod testov: Pravokotniki se pobarvajo v vrstnem redu: rdeč, oranžen, rumen, zelen, moder in vijoličen. [35, 36]

V tabeli 5 so prikazani rezultati posameznih brskalnikov. Acid3 test lahko opravimo na naslovu <http://acid3.acidtests.org/>.

Za ogled testov, ki jih brskalnik ne opravi in razloge za to, je treba klikniti na testu izrisano črko »A«, ko se test konča. Podatki se nam izpišejo v obliki okna.







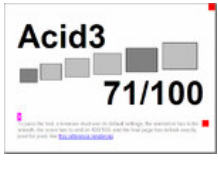


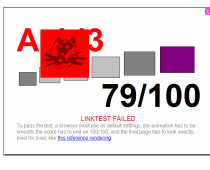
				
				

Tabela 5: Rezultati Acid3 testa za posamezne brskalnike.

Razvrstitev brskalnikov po Acid3 testu:

1. Opera
2. Chrome
3. Safari
4. Firefox
5. Explorer

#### 4.4.4. CSS3

CSS standard je doživel že več različnih verzij in je bil prilagojen različnim profilom. Vsaka naslednja verzija CSS standarda nadgrajuje prejšnjo. Tu gre predvsem za dodajanje novih funkcij in boljše izrabljanje prejšnjih. Označimo jih kot CSS1, CSS2 in CSS3. CSS3 je aktualna verzija, katero tudi testiramo v tem testu.

Profili so podmnožice ene ali več verzij CSS-jev, ki so zgrajeni za posebno napravo ali uporabniški vmesnik. Trenutno obstajajo profili za mobilne naprave, tiskalnike in televizijske sprejemnike.

Prva CSS specifikacija, ki je postala uradno W3C CSS priporočilo, je **CSS1**, objavljen pa je bila decembra 1996. Obsega:

- lastnosti pisave (npr. poudarki),
- barve besedil, okolja in drugih elementov,
- lastnosti teksta (npr. razmiki med besedami, črkami in vrsticami besedila),
- razvrstitev in postavitev teksta, slik, tabel in drugih elementov,
- razmiki med elementi, ki jih povzročajo lastnosti »margin«, »padding«, »border« in pozicioniranje elementov,
- enotna identifikacija in splošna klasifikacija skupin atributov.

**CSS2** so razvili pri W3C in ga objavili kot priporočilo v maju 1998. CSS2 je nadgradnja CSS1 in vključuje številne nove zmogljivosti kot so absolutno, fiksno in relativno pozicioniranje elementov, podporo vrstam medija, podpora zvočnim predlogam in številne nove lastnosti pisav, kot je npr »shadow«. W3C ohranja CSS2 priporočilo. CSS2 doživi še revizijo, ki odpravi nekatere napake, tako dobimo CSS2.1.

Že leta 1998 so začeli z razvojem **CSS3** ravni. W3C ga še vedno razvija in med razvojem izdaja poročila o napredku v razvoju. Zgrajen bo iz več ločenih priporočil. [37]

Razvoja so se lotili tako, da so ga razdelili na 3 stopnje prioritete in na postavko ostalo. O poteku razvoja in trenutnem stanju si lahko več preberemo na strani <http://www.w3.org/Style/CSS/current-work>.

CSS3 bo spremenil splet. Novosti je veliko, nekatere so nadvse zanimive in uporabne. Številne nove funkcije so implementirane in delujoče samo v nekaterih brskalnikih. Tu izstopajo predvsem Opera, Firefox, Safari/Webkit in Konqueror, ki ga v diplomski nalogi nismo obravnavali. Posamezni brskalniki lastnosti CSS3-ja berejo s svojo sintakso: tako brskalniki lastnost »background-size« poimenujejo različno, pri Operi se napiše »-o-background-size«, pri Safari-ju pa »-webkit-background-size«. Ravno tako pri Firefox-u zapis lastnosti »border-radius« poznamo kot: »-moz-border-radius«.

Nekatere funkcije iz zmogljivosti CSS3-ja:

## Obrobe:

- `border-color`, s katerim lahko naredimo obrobe v več barvah.

```
.mozBorder{
  border: 8px #000 solid;
  padding: 2px;
  -moz-border-bottom-colors: #555 #666 #777 #888 #999 #aaa #bbb #ccc;
  -moz-border-top-colors: #555 #666 #777 #888 #999 #aaa #bbb #ccc;
  -moz-border-left-colors: #555 #666 #777 #888 #999 #aaa #bbb #ccc;
  -moz-border-right-colors: #555 #666 #777 #888 #999 #aaa #bbb #ccc;
}
```

Firefox zna izrisati zelo zanimive obrobe.

- `border-image` ponuja obrobe iz slik.

```
.borderImg{
  border-image: url(border.png) 27 27 27 27 stretch stretch;
}
```

Lorem ipsum dolor sit amet.

- `border-radius` naredi obrobe okrogle.

Oglate obrobe v Firefoxu, Safariju in Chromu.

```
.borderRadius{
  background-color: #ccc;
  -moz-border-radius: 5px;
  -webkit-border-radius: 5px;
  border: 1px solid #000;
  padding: 10px;
}
```

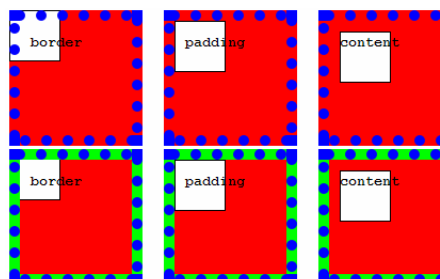
- Box-shadow obrobo osenči.

Osenčenje v Safariju in Chromu.

```
.boxShadow{
  background-color: #ccc;
  -webkit-box-shadow: 10px 10px 5px #888;
  padding: 5px 5px 5px 15px;
}
```


## Ozadja:

- Background-origin in background-clip lahko uporabljamo v Safariju, Firefoxu in Chromu. Imata lahko 3 vrednosti; border, padding in content.



Prvi definira, kako se postavi element v škatli, npr. glede na njegov border, njegov padding ali glede na njegov content, se pravi glede na element sam.

Drugi pa definira, če se element zlije z obrobo škatle v katerem se nahaja, ali s paddingom ali contentom.

- Background-size definira velikost ozadja v nekem elementu (npr velikost slike, ki je ozadje elementa <div>).  Slika, ki je postavljena kot ozadje je zmanjšana.

```
.backgroundSize{
  background-image: url("../img/tool_icon.png");
  background-repeat: no-repeat;
  -webkit-background-size: 50px 20px;
}
```

- Multiple backgrounds za več ozadij znotraj enea elementa.



```
.multiBackgrounds{
  background: url(body-top.gif) top left no-repeat,
  url(banner_fresco.jpg) top 11px no-repeat,
  url(body-bottom.gif) bottom left no-repeat,
  url(body-middle.gif) left repeat-y;
}
```

## Barve:

- HSL Colors. HSL ima tri vrednosti; Hue, Saturation, Lightness, slovensko: barvni odtenek, nasičenost, lahkost. Barvni odtenek je stopnja na barvnem kolesu: 0 (ali 360) rdeče, 120 zelene, 240 pa modre barve. Številke med temi



vrednostmi odražajo različne odtenke. Nasičenost je odstotek vrednosti: 100% je v celoti barva, 0% pa brez barve. Lahkost v odstotku: 0% je temna (črna), 100% svetla (bela), in 50% je povprečje. To je zelo širok spekter razpoložljivih barv in tonov.

```
<div style="background-color: hsl(0, 100%, 50%);"></div>
<div style="background-color: hsl(120, 100%, 50%);"></div>
<div style="background-color: hsl(240, 100%, 50%);"></div>
```

- HSLA Colors, pri čemur »A« določa motnost.



```
<div style="background-color: hsla(0, 100%, 50%, 0.2);"></div>
<div style="background-color: hsla(0, 100%, 50%, 0.4);"></div>
<div style="background-color: hsla(0, 100%, 50%, 0.6);"></div>
<div style="background-color: hsla(0, 100%, 50%, 0.8);"></div>
<div style="background-color: hsla(0, 100%, 50%, 1);"></div>
```

- Opacity kot dodaten atribut omogoča lažje prehajanje med barvami. Na spodnjem primeru vidimo, kako enostavneje je barvati elemente z različnimi rdečimi barvami s pomočjo atributa »opacity«.



```
<div style=" background: rgb(255, 0, 0) ; opacity: 0.2;"></div>
<div style=" background: rgb(255, 0, 0) ; opacity: 0.4;"></div>
<div style=" background: rgb(255, 0, 0) ; opacity: 0.6;"></div>
<div style=" background: rgb(255, 0, 0) ; opacity: 0.8;"></div>
<div style=" background: rgb(255, 0, 0) ; opacity: 1;"></div>

<div style=" background: rgb(243, 191, 189) ; "></div>
<div style=" background: rgb(246, 143, 142) ; "></div>
<div style=" background: rgb(249, 95 , 94) ; "></div>
<div style=" background: rgb(252, 47, 47) ; "></div>
<div style=" background: rgb(255, 0, 0) ; "></div>
```

- RGBA Colors, pri čemur »A« določa motnost.



```
<div style="background: rgba(255, 0, 0, 0.2);"></div>
<div style="background: rgba(255, 0, 0, 0.4);"></div>
<div style="background: rgba(255, 0, 0, 0.6);"></div>
<div style="background: rgba(255, 0, 0, 0.8);"></div>
<div style="background: rgba(255, 0, 0, 1) ;"></div>
```

## Posebni efekti za besedilo:

- Text-shadow osenči besedilo.

Osenčeno besedilo.

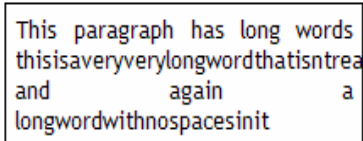
```
.textShadow{
  text-shadow: 4px 4px 4px #321;
  font-size: 30px;
}
```

- Text-overflow skrije predolge stavke. Na sliki smo na spodnji pravokotnik postavili miško.

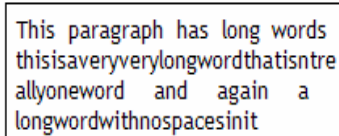
Predolg  
Predolg tekst drugič.

```
text-overflow: ellipsis;
```

- Word-wrap skrbi za lomljenje besed, ki so predolge.



```
word-wrap: normal;
```



```
word-wrap: break-word;
```

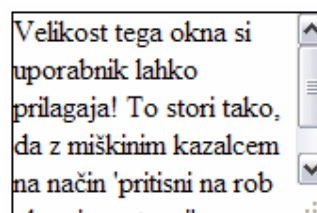
## Uporabniški vmesnik:

- Box-sizing skrbi za to, da lahko brskalnik prepričamo da padding, width in height lastnosti vključi v sam element, se pravi, jih ne prišteva k velikosti elementa.

```
box-sizing: border-box;
```

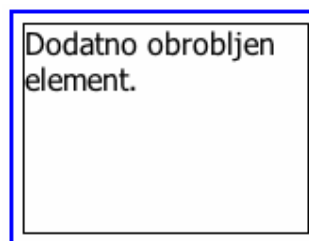
- Resize omogoča, da si uporabnik prilagodi velikost HTML elementa (npr <div>), da tako lažje prebere vsebino.

```
.resizeDiv{
  width: 100px;
  height: 50px;
  border: 1px solid;
  resize: both;
  overflow: auto;
}
```



- Outline omogoča dodatno obrobjanje elementa zunaj njegovega roba. S tem vstopimo v prostor drugega elementa.

```
.outlineDiv{
  width: 150px;
  height: 110px;
  border: 1px solid;
  outline-offset: 5px;
  outline: 2px solid blue;
}
```



## Drugi modeli:

- Media queries so prihodnost spleta. Z njihovo pomočjo, je stran lahko izpisana za več naprav (npr. zaslon, mobitel).
- Multi-column izgled omogoča razdelitev teksta na več stolpcev.

```
-webkit-column-count: 3;
-webkit-column-gap: 1em;
-webkit-column-rule: 1px solid black;
```

Cras urna metus, aliquam sed, condimentum eget, pellentesque scelerisque, massa. Nullam et est id augue blandit tincidunt. Ut consectetur, justo eleifend varius facilisis, tortor lorem pharetra nunc, ac sodales purus nunc semper tortor. Integer nec	urna. Praesent scelerisque, ipsum nec aliquet volutpat, sem ante sagittis risus, sed condimentum magna libero luctus elit. Donec pede purus, hendrerit non, laoreet vel, porta ut, neque. Cras eu lacus. Pellentesque tempus mattis magna. Nullam id nisl. Quisque dolor lorem, commodo	ac, pharetra sed, nonummy nec, nulla. Mauris purus. Suspendisse eget mauris nec justo eleifend vestibulum. Nunc ut eros sed pede pretium congue. Etiam in elit ut nisi ultrices hendrerit. Cras vulputate ultrices quam. Curabitur venenatis. Cras sed nisi.
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



- Web fonts omogoča, da brskalnik prikaže tekst v pisavi, ki jo najde nekje na spletu. Safari lahko prikaže pisavo, ki jo računalnik ne poseduje. Našel jo je kar na spletu.

```
@font-face {
  font-family: Delicious;
  font-weight: bold;
  src: url('http://www.css3.info/wp-content/uploads/2008/06/delicious-roman.otf');
}
h3 {
  font-family: Delicious, sans-serif;
}
```

- Speech podpira branje označenega besedila. Nosilec te funkcionalnosti, ki bo nekoč verjetno standard, je Opera. Branje označenega besedila omogočimo v Preferences > Advanced > Voice.

```
#voice-volume { -xv-voice-volume: x-soft; -xv-voice-balance: right; }
#voice-balance { -xv-voice-balance: left; }
#speech-cue { cue-after: url(ding.wav); }
#voice-rate { -xv-voice-rate: x-slow; }
#voice-family { voice-family: female; }
#voice-pitch { -xv-voice-pitch: x-low; }
#speech-speak { speak: spell-out; }
```

## Selektorji:

Test CSS3 selektorjev se nahaja na strani <http://www.css3.info/selectors-test/test.html>. Izvede se samodejno in ob koncu posreduje rezultate. Pri testiranju ni potrebno spreminjati nikakršnih nastavitev brskalnika, saj to na rezultat ne vpliva. Test vsebuje 578 preizkusov. Test smo pognali za vse testirane brskalnike in rezultate shranili v tabelo 6.

Brskalnik	Število opravljenih testov od 578
Internet Explorer 7.0	330
Mozilla Firefox 3.x	374
Opera 9.x	578
Apple Safari 3.x	578
Google Chrome 1.x	578

Tabela 6: Število opravljenih CSS3 selektor testov za posamezen brskalnik.

Podpora brskalnikov posameznim funkcionalnostim, ki jih navajajo v CSS3 standardu, smo zapisali v tabelo 7. [38, 39]

Zmogljivosti	Internet Explorer 7	Mozilla Firefox 3	Opera 9	Apple Safari 3	Google Chrome
Border-color	Ne	Da	Ne	Ne	Ne
Border-image	Ne	Ne	Ne	Da	Da
Border-radius	Ne	Da	Ne	Da	Da
Box-shadow	Ne	Ne	Ne	Da	Da
Background-origin	Ne	Da	Ne	Da	Da
Background-clip	Ne	Da	Ne	Da	Da
Background-size	Ne	Ne	Da	Da	Da
Multiple backgrounds	Ne	Ne	Ne	Da	Da
HSL Colors	Ne	Da	Da	Da	Da
HSLA Colors	Ne	Da	Ne	Da	Da
Opacity	Ne	Da	Da	Da	Da
RGBA Colors	Ne	Da	Ne	Da	Da
Text-shadow	Ne	Ne	Da	Da	Ne
Text-overflow	Da	Da	Da	Da	Da
Word-wrap	Da	Ne	Ne	Da	Da
Box-sizing	Ne	Da	Da	Da	Da
Resize	Ne	Ne	Ne	Da	Da
Outline	Ne	Da	Da	Da	Da
Media queries	Ne	Ne	Da	Da	Da
Multi-column layout	Ne	Da	Ne	Da	Da
Web fonts	Ne	Ne	Ne	Da	Ne
Speech	Ne	Ne	Da	Ne	Ne
Skupaj podprtih	2	12	9	20	18

*Tabela 7: Podpora potencialnim zmogljivostim CSS3-ja posameznega brskalnika.*

Ko teste iz tabele 6 in 7 združimo, vidimo, da se Safari in Chrome zelo dobro obneseta pri testu selektorju, kot tudi pri ostalih CSS3 testih. Opera se ravno tako zelo dobro obnese v testu selektorju, med tem ko ji pri drugih funkcionalnostih CSS3 standarda zmanjka moči. Sledi ji Firefox s povprečnim rezultatom. Najslabši pa je Explorer, ki se izkaže za nepripravljenega na nov standard.

Razvrstitev:

1. Safari
2. Chrome
3. Opera
4. Firefox
5. Explorer

#### 4.4.5. JavaScript

- **Podpora JavaScript-a**

V tabeli 8 je zapisano, kaj od izbranih JavaScript tehnologij in pa standardov posamezen brskalnik podpira. Številka 1 pomeni, da podpira, številka 0, da ne podpira, 0.5 pa da delno podpira. Če podpora standarda ni znana, jo označimo z vprašajem.

	Internet Explorer 7	Mozilla Firefox 3	Opera 9	Apple Safari 3	Google Chrome
JavaScript	1	1	1	1	1
ECMAScript 3	1	1	1	1	1
DOM 1	0,5	1	1	1	1
DOM 2	0	1	1	1	1
DOM 3	0	0,5	0,5	0,5	?
XPath	1	1	1	1	?
DHTML	1	1	1	1	1
XMLHttpRequest	1	1	1	1	1
Rich editing	1	1	1	1	1
Skupaj podprtih	6,5	7,5	7,5	7,5	7

*Tabela 8: Podpora JavaScript tehnologijam posameznega brskalnika. [31]*

- **Hitrost izvajanja JavaScript operacij**

Test preveri hitrost izvajanja različnih JavaScript operacij, ki se izvajajo na odjemalčevem računalniku v samem brskalniku. V tabeli 9 so rezultati testa. Test lahko opravimo na strani: <http://wd-testnet.world-direct.at/mozilla/dhtml/funo/jsTimeTest.htm>

Testi	Internet Explorer 7	Mozilla Firefox 3	Opera 9	Apple Safari 3	Google Chrome
For-zanka	170	75	125	110	8
Seštevanje	202	130	94	198	6
Odštevanje	683	124	79	190	6
Množenje	215	128	94	169	8
Deljenje	650	154	125	201	44
Deljenje 2	259	153	188	200	35
Iskanje vrednosti v tabeli	295	97	94	148	20
parseInt()	4980	1613	2984	2355	185
Deklariranje spremenljivk	326	138	109	143	27
Math.sin()	1708	367	656	545	178

Math.floor()	1632	292	671	450	126
if	203	96	78	121	4
Branje globalnih spremenljivk	281	90	110	219	15
Seštevanje nizov	1520	1255	781	574	68
Sortiranje tabele	47	12	219	14	19
Vsota časov	13171	4724	6407	5637	749
Povprečje	878,07	314,93	427,13	375,8	49,93

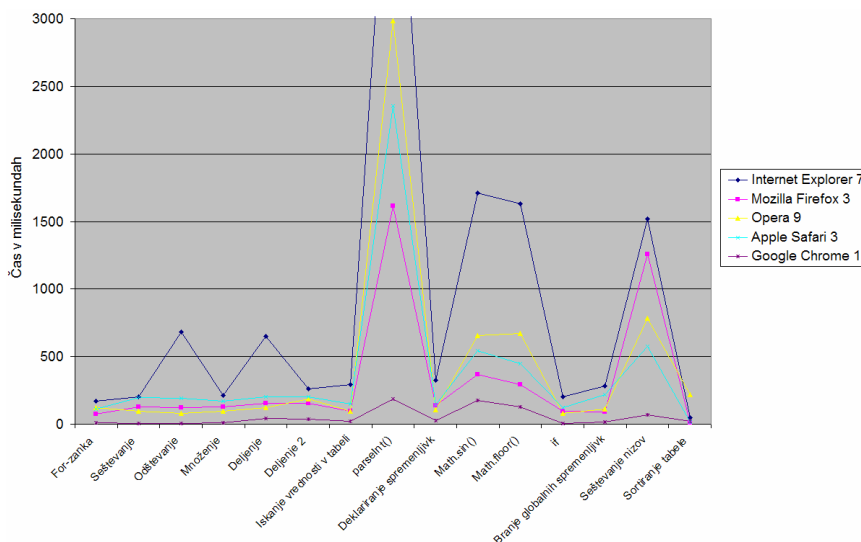
Tabela 9: Test delovanja JavaScript-a v brskalnikih.

Na sliki 26 je lepo vidno kateri, brskalnik je najboljši in kateri najslabši. Google Chrome je zmagovalec vseh testov, Internet Explorer pa poraženec. Viden je vrstni red brskalnikov, saj se v skoraj vseh testih razporedijo v enakem vrstnem redu, kar priča o tem, da so testi objektivni oziroma pokažejo dejansko moč brskalnika v izvajanju JavaScript-a. Vsak test razen zadnjega se izvede milijonkrat.

Vrstni red je sledeč:

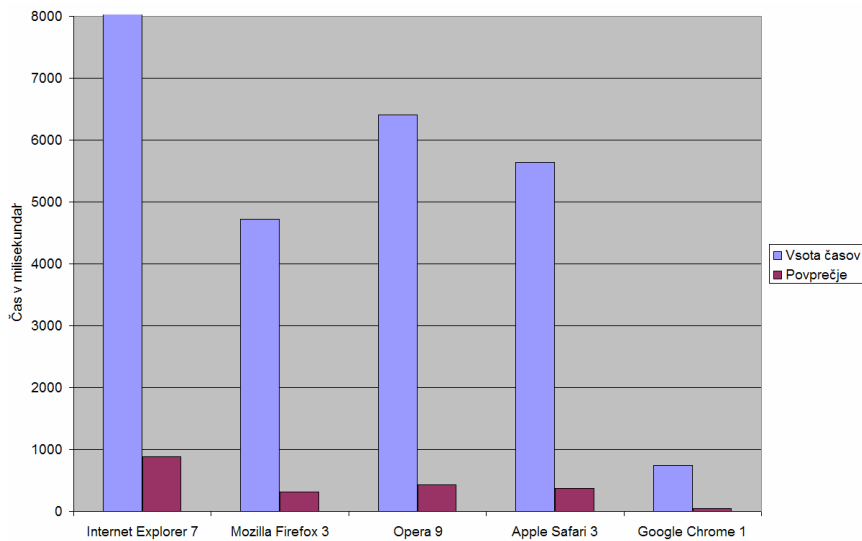
1. Chrome
2. Firefox
3. Safari
4. Opera
5. Explorerer

Vrh grafa smo zaradi boljšega pregleda odrezali, ker je Internet Explorer v testu parseInt() zabeležil kar 4980 milisekund, kar je veliko več kot ostali brskalniki. Če grafa nebi odrezali, bi bile v spodnjem delu vidne samo ravne črte.



Slika 26: Izmerjen čas posameznih JavaScript operacij.

Na sliki 27 je prikazan seštevek izmerjenih časov in povprečje izmerjenih časov za vse teste za posamezen brskalnik. Tudi spodnjemu grafu smo zaradi preglednosti odrezali vrh grafa, saj je imel Explorer v primerjavi z ostalimi brskalniki precej slabši rezultat. Na grafu je lepo vidna moč Chroma v izvajanju JavaScript operacij, saj je ostale pustil daleč za seboj. Nekako najbližji mu je bil Firefox, a tudi ta je daleč za njim. Internet Explorer je v tem testu gotovo veliki poraženec.



Slika 27: Vsota in povprečje časov izvajanja JavaScript operacij.

## 5. Analiza rezultatov s programom HiView3

HiView3 je orodje za analizo več kriterijskega odločanja. Obsega funkcije, s pomočjo katerih lahko izdelamo sorazmerno kvalitetne ocene za odločanje. HiView3 pri svojem delovanju uporablja odločanje na podlagi večkriterijske analize MCDA (Multi-Criteria Decision Analysis) in temelji na skladnosti splošnih načel teorije odločanja. S tem zagotavlja potrebno trdnost pri pomoči odločanja in strateško izbiranje.

HiView3 nam pomaga oceniti in ovrednotiti možnosti za lažje odločanje. Temelji na računalniški obdelavi podatkov, odločitvenemu modeliranju, te podatke ocenjuje in vrednoti možnosti, ki jih ti ponujajo. Je enako zmogljiv za skupinsko odločanje (npr odločitvene konference) kot tudi za samostojno odločanje. Z množico uporabniških funkcij je HiView3 mogoče nastaviti oz prilagajati za reševanje različnih problemskih domen.

Metoda HiView obsega:

- Definiranje okvirov odločitvenega problema
- Identificiranje alternativ
- Identificiranje ciljev in kriterijev
- Vrednotenje alternativ po kriterijih
- Kreiranje funkcij koristnosti
- Uteževanje kriterijev
- Analiza rezultatov
- Analiza občutljivosti

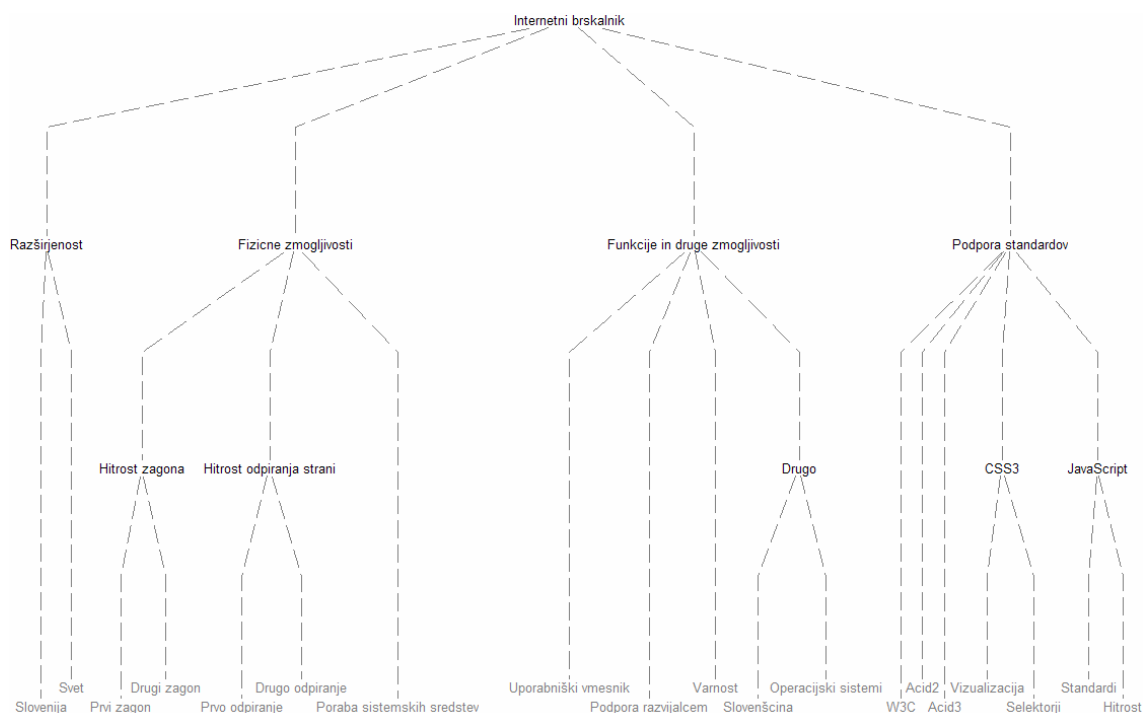
[40]

S programom smo naredili odločitveni model, ki nam je pomagal pri odločitvi, kateri brskalnik je trenutno najprimernejši na trgu, pri tem pa smo predpostavili, da se odločamo s stališča slovenskega študenta na Fakulteti za računalništvo in informatiko.

Model lahko prilagajamo tudi drugačnim uporabnikom in tako dobimo različne zmagovalce. Razvijalcu je recimo najbolj pomembna dobra podpora standardov in razvijanju, zato mogoče zmaga Firefox, med tem ko je ob osnovnih potrebah končnega uporabnika po enostavnosti in varnosti zadosten tudi Explorer.

## 5.1. Hierarhični model

Prva stopnja pri modeliranju s programom HiView3 je izgradnja hierarhičnega modela, kot ga vidimo na sliki 28. Vsako vozlišče, ki se nahaja na repu modela (list drevesa) je lastnost ali merilo, s katerim ocenjujemo brskalnike. Te lastnosti je potrebno izmeriti in zapisati. Posamezne lastnosti so združene v vozlišče, ki združujejo podobna merila med seboj v skupine. [41]



Slika 28: Hierarhični model v programu HiView3.

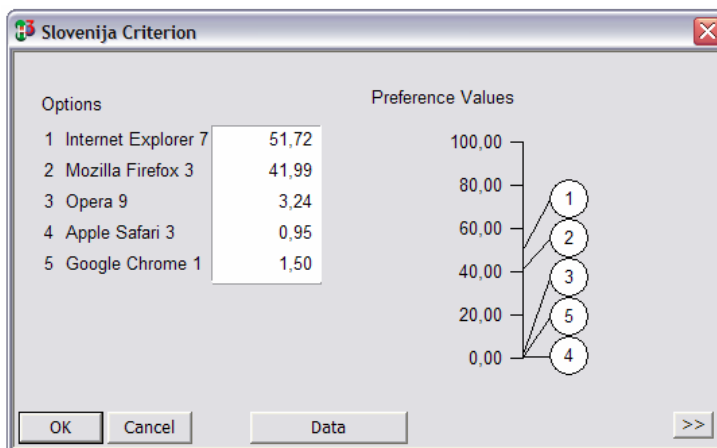
Kot je bilo zastavljeno v diplomski nalogi smo spletne brskalnike opazovali z vidika razširjenosti, fizičnih zmogljivosti, funkcij in drugih zmogljivosti ter podpore standardom. Posamezne kriterije smo razdelili na podkriterije, ki smo jih v diplomski nalogi spoznali in ocenili.

## 5.2. Opis in ocenjevanje kriterijev

Kriterije smo že predstavili in opisali v diplomu. V tej točki bomo opisali njihovo uporabo v programu HiView3.

Večino kriterijev v našem modelu smo izmerili z različnimi testi, kot so testi razširjenosti, hitrostni testi, testi podpore standardov itd. Nekatere kriterije se ni dalo izmeriti, zato smo jih ocenili in izmerili opisno. Pri tem smo se trudili, da smo bili čimbolj objektivni.

Na sliki 29 vidimo primer za ocenitev kriterija razširjenost brskalnikov v Sloveniji. Za ocenitev kriterija smo izbrali kar izmerjeno razširjenost brskalnikov med uporabniki, ki je bila opravljena na spletu. [41]



Slika 29: Okno za vnos izmerjenih vrednosti za merilo razširjenost brskalnika v Sloveniji.

Ker smo kriterije ocenjevali z različnimi lestvicami, je HiView3 le te normiral, tako da izbor obsega (0-10, 0-4 ali celo 0-100) ni vplival na pomembnost kriterija v modelu. Pri opisnih ocenah kriterijev smo se odločili za številske ocene, saj je tako preprosteje kvantitativno opisati in razumeti razmerja med njimi. Ocene po posameznih kriterijih kaže tabela 10. V nadaljevanju bomo pojasnili, kaj pomenijo posamezni kriteriji in utemeljili svoje opisne ocene.

Na koncu tabele smo zapisali še skupno število točk za posamezen brskalnik, ki pa nam pove veliko, saj obseg ocen ni povsod enak (npr. razširjenost 0 do 100, hitrost 0 do 5). Rezultat, ki nam ga bo vrnil program HiView3, bo drugačen, saj bo program ocene normiral, posameznim kriterijem pa bomo določili različno pomembnost. To bomo storili v naslednjem poglavju, ko bomo kriterije utežili.



	Internet Explorer 7	Mozilla Firefox 3	Opera 9	Apple Safari 3	Google Chrome 1
Razširjenost Slovenija	51,72	41,99	3,24	0,95	1,5
Razširjenost svet	45,7	44,4	2,4	2,7	3,6
Hitrost prvega zagona brskalnik.	2,6	3,6	2,5	2,3	1,7
Hitrost drugega zagona brskalnik.	0,7	1,4	0,9	0,7	0,6
Hitrost prvega odpiranja stran.	31,4	30,5	28,9	26,8	26,4
Hitrost drugega odpiranja stran.	20,2	14,7	13,7	14,8	17,4
Poraba sredstev	1,55	0,93	0,92	1,05	3,03
Uporabniški vmesnik	7,5	7,8	8	5,6	7,5
Podpora razvijalcem	6,2	8,7	7,8	7,8	7,0
Varnost	7,8	8,9	10	8,5	7,1
Slovenski jezik	1	1	0	0	1
Operacijski sistemi	1	3	3	2	1
W3C	5	8,5	8,5	7,5	6,5
Acid2	0	1	1	1	1
Acid3	14	71	85	75	79
Vizualizacija elementov	2	12	9	20	18
Selektorji	330	374	578	578	578
JavaScript standardi	6,5	7,5	7,5	7,5	7
JavaScript hitrost	4	8	6	7	10
Skupaj	538,87	648,92	776,36	769,2	777,33

*Tabela 10: Ocene kriterijev za posamezen brskalnik.*

#### Razširjenost brskalnika:

Za kriterij *razširjenost brskalnika v Sloveniji in svetu* smo izbrali kar izmerjene rezultate na spletu, ki so merili razširjenost brskalnika med uporabniki v Sloveniji in drugod. Ker so bili izmerjeni rezultati odstotki, je obseg ocene za ta kriterij med 0 in 100.

Hitrost zagona brskalnika:

Kriterij *hitrost prvega in drugega zagona brskalnika s prazno stranjo*, smo merili v obsegu med 0 in 5, pri čemur 0 pomeni, da je zelo hiter, 5 pa zelo počasen. Ocenili smo, da je zgornja meja, ki je še sprejemljiva 5. Kriterije smo ocenili z izmerjenimi časi v sekundah.

Hitrost odpiranja internetnih strani:

*Hitrost prvega in drugega odpiranja internetne strani* smo ocenili v obsegu med 0 in 50. Vpisali smo seštevke izmerjenih časov v sekundah, ki jih je brskalnik dosegel pri posameznih internetnih straneh. Seštevke smo uporabili zato, ker bi bilo testiranje ene internetne strani lahko zavajajoče, saj bi lahko bila prilagojena enemu izmed brskalnikov. Pri tem 0 pomeni, da strani odpira zelo hitro, 50 pa zelo počasi in je zgornja sprejemljiva meja.

Poraba sistemskih sredstev:

Pri *porabi sistemskih sredstev* smo se odločili da bomo upoštevali samo test, ki je bil opravljen na 10 zavihkih, saj so se vsi brskalniki pri več kot 2 zavihkih razvrstili v enak vrstni red, zato smo teste za 1, 2 ali 0 zavihkov zanemarili. Izmerjene čase smo delili s 100 in s tem dobili obseg ocen od 0 do 4, pri čemur 0 pomeni, da je njegova poraba zelo majhna, 4 pa zelo velika.

Uporabniški vmesnik:

Ocene *uporabniškega vmesnika* so bile precej subjektivne, vendar smo se vseeno potrudili in našli način kako bi bile kar se da objektivne. Pri posameznemu brskalniku smo našli določene pluse in minuse. Med vsemi plusi in minusi smo izračunali koliko odstotkov je plusov in rezultat pretvorili v številke od 0 do 10, pri čemur 0 pomeni, da ima brskalnik zelo slab uporabniški vmesnik, 10 pa zelo dober.

Podpora razvijalcem:

Tudi ocene za *podporo razvijalcem* so precej subjektivne, saj ni bilo mogoče izvesti podrobnega testa, ki bi dal natančnejši rezultat. Tudi tu delež plusov med vsemi znaki določa končno oceno, zopet z ocenami od 0 do 10, pri čemur 0 velja za slabo, 10 pa zelo dobro podporo razvijalcem. Podporo za razvijalce so imeli vsi brskalniki izdelano približno enako dobro. Razlike smo iskali predvsem v tem, kako je podpora implementirana s stališča uporabniškega vmesnika in koliko dodatki za razvijalce vplivajo na hitrosti brskalnika. Kot plus smo šteli vse dodatne funkcije pri posameznem brskalniku, katere drugi niso imeli implementirane. Firefox in Opera imata v orodni vrstici bližnjice za posamezne funkcije, prav tako pa tudi okno, ki nam prikazuje podrobnosti na strani, ki ga lahko tudi izklopimo. Safari in Explorer vsebujeta le okno za pomoč pri razvoju, ki je lahko večkrat v napoto. Chrome odpira okno za razvijalce v novem oknu, kar naredi delo nekoliko bolj moteče, saj se moramo z miško iskati med okni.

Varnost:

*Varnost* je bilo gotovo najtežje oceniti. Brskalnike smo testirali, kako blokirajo nezaželeno okna, njihovo podporo certifikatom, njihove dodatne varnostne funkcije in si ogledali na spletu kako je z varnostnimi luknjami. Zopet smo s plusi in minusi izpostavili glavne značilnosti in dobili kar se da objektivne rezultate. Zaradi nekoliko bolj občutljive zadeve smo ocene razdelili med 5 in 10, saj bi bil brskalnik, ki bi dobil oceno manj kot 5 prenevaren za uporabo. 5 pomeni, da je brskalnik dokaj nevaren za uporabo, 10 pa, da je zelo varen.

Slovenski vmesnik:

Kriterij o *obstoju slovenskega uporabniškega vmesnika* smo ocenili z ocenama 0 in 1, pri čemer 0 pomeni, da slovenskega vmesnika nima, 1 pa, da ga ima.

Operacijski sistemi:

Na kateri *operacijski sistem* je možno brskalnik naložiti smo ocenili z ocenami med 0 in 3. Številka določa na koliko operacijskih sistemov je brskalnik možno naložiti.

W3C:

Na spletu smo dobili rezultate za 10 *spletnih standardov*, ki jih priporoča W3C, zato smo izbrali oceno v obsegu med 0 in 10, glede na to, koliko standardov brskalnik podpira.

Acid2:

Pri testu *Acid2* nas je zanimalo samo ali brskalnik opravi test ali ga ne opravi. Zato smo ta kriterij ocenili z 0, ko brskalnik ni opravil testa in 1, ko ga je opravil.

Acid3:

Test *Acid3* testira, kako brskalnik podpira zadnje standarde. Rezultat testa je število opravljenih podtestov. Tako smo obseg ocen izbrali med 0 in 100 in kot rezultat meritve vpisali število opravljenih podtestov.

CSS3:

CSS3 test smo razdelili na dva dela. Prvi del se ukvarja s podporo oblikovanja posameznih HTML elementov in vsebuje 22 testov, zato smo tudi izbrali obseg ocen med 0 in 22. Drugi se ukvarja s selektorji in vsebuje 578 podtestov, zato smo izbrali obseg ocen med 0 in 578.

JavaScript:

Tudi *JavaScript* smo razdelili na dva dela. Prvi del vsebuje 9 testov podpore JavaScript funkcij in standardov, zato je obseg ocen med 0 in 9. Drugi meri hitrost izvajanja JavaScript operacij. Rezultat meritev je subjektivna ocena hitrost z ocenami od 0 do 10, pri čemer 0 pomeni da je zelo počasen, 10 pa zelo hiter.

### 5.3. Utežitvev kriterijev

Uteževanje kriterijev je ključna razlika med MCDA in drugimi odločitvenimi tehnikami. Na tej stopnji moramo imeti potrebno znanje za relativno presojo o pomembnosti posameznih kriterijev. Kriterije na nekem nivoju otežimo tako, da je vsota vseh uteži enaka 100. Navadno to delo opravlja skupina, ki vsebuje potrebne izkušnje in znanje. V našem primeru bomo model utežili kar sami. [41]

#### **Razširjenost 10**

Slovenija	70
Svet	30

#### **Fizične zmogljivosti 20**

Hitrost zagona brskalnika	30
Prvi zagon	20
Drugi zagon	80
Hitrost odpiranja strani	50
Prvo odpiranje	40
Drugo odpiranje	60
Poraba sistemskih sredstev	20

#### **Funkcije in druge zmogljivosti 30**

Uporabniški vmesnik	20
Podpora razvijalcem	30
Varnost	35
Druge značilnosti	15
Slovenski jezik	70
Podprtost sistemom	30

#### **Podpora standardom 40**

W3C	20
Acid2	10
Acid3	25
CSS3	25
Podpora vizualnosti	50
Podpora selektorjem	50
JavaScript	20
Podpora	50
Hitrost	50

Na sliki 30 je primer utežitve v programu HiView3 za kriterija razširjenost brskalnika v Sloveniji in svetu.

Razširjenost	Weight	Mozilla Firefox 3	Apple Safari 3	Opera 9	Google Chrome 1	Cumulative Weight	
Slovenija*	70	52	42	3	1	2	7,0
Svet*	30	46	44	2	3	4	3,0
TOTAL	100	50	43	3	1	2	10,0

Slika 30: Primer utežitve kriterija v programu HiView3.

Za najbolj pomemben kriterij smo si izbrali podpora standardom (**40**), saj menimo, da je najpomembneje to, da brskalnik pravilno izrisuje spletne strani, kot to določajo standardi. Kot drugi najpomembnejši kriterij smo izbrali funkcije in druge zmogljivosti (**30**), saj želimo brskalnik, ki bo omogočal udobno in varno delo in nam pri tem pomagal na različne načine. Nekoliko manj pomemben kriterij so fizične zmogljivosti (**20**), saj se ob obilici zmogljivih računalnikov bistveno ne pozna razlika pri normalni uporabi brskalnika. Najmanj pomemben kriterij je razširjenost (**10**), saj pogostost uporabe brskalnika bistveno ne vpliva na naše delo. Pri zelo razširjenih brskalnikih lahko lažje pridemo do pomoči drugih uporabnikov, ki uporabljajo enak brskalnik kot mi.

Kriterij razširjenost smo dalje razdelili na razširjenost v Sloveniji in v svetu, pri čemur smo dali razširjenosti v Sloveniji (**70**) nekoliko večji poudarek kot v svetu (**30**).

Fizične zmogljivosti smo razdelili na hitrost zagona brskalnika (**30**), hitrost odpiranja spletnih strani (**50**) in porabo sistemskih sredstev (**20**). Menimo da je hitrost odpiranja spletnih strani najbolj pomemben kriterij, saj to tudi največkrat počnemo z brskalnikom. Hitrost zagona brskalnika je nekoliko manj pomemben kriterij, vendar ne zanemarljiv, saj si želimo kar se da hitrega in udobnega dela. Najmanj nas skrbi poraba sistemskih sredstev, saj imajo v teh časih vsi računalniki zadostne količine pomnilnika.

Hitrost zagona brskalnika smo razdelili še na prvi zagon in drugi zagon v razmerju **20:80**, saj od brskalnika ne pričakujemo bliskovite hitrosti ob prvi uporabi, želimo pa si udobnega dela v nadalje.

Hitrost odpiranja spletnih strani smo podobno kot hitrost zagona brskalnika razdelili na prvo in drugo odpiranje, vendar z nekoliko manjšim razmerjem **40:60**, saj posamezno spletno stran odpremo dnevno enkrat do dvakrat in ne večkrat, kot je to običajno s samim brskalnikom.

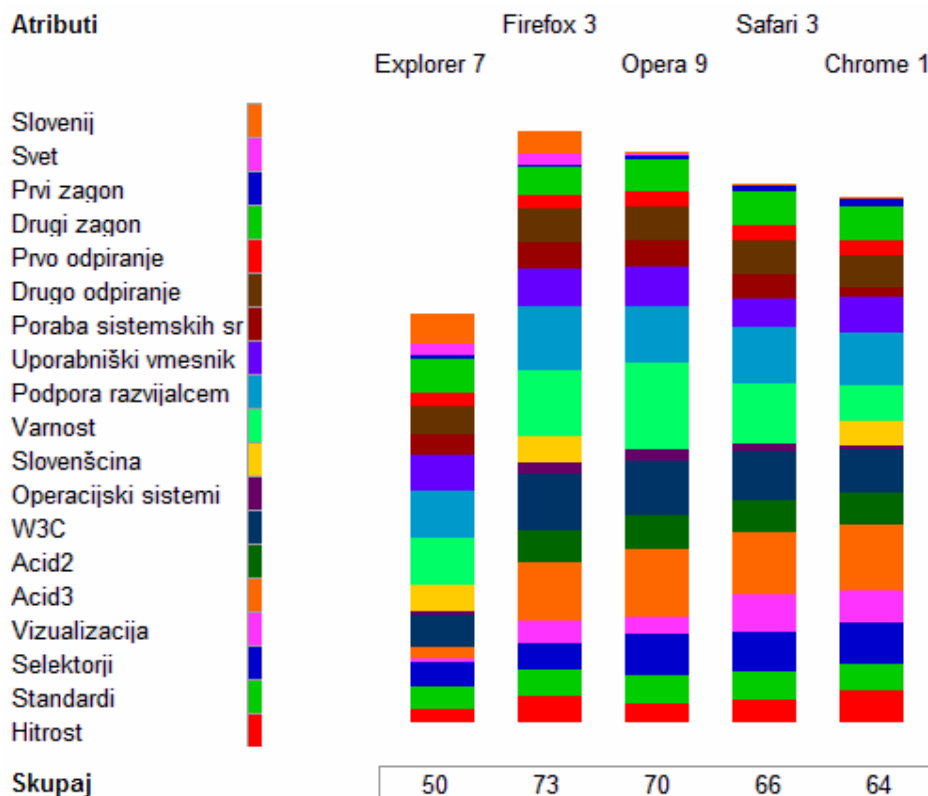
Funkcije in druge zmogljivosti smo razdelili na varnost (**35**), ki se nam zdi najbolj pomembna v obdobju elektronskega poslovanja, uporabniški vmesnik (**20**), ki se nam zdi malo manj pomemben, podporo razvijalcem (**30**), saj sodimo med razvijalce in je to za nas seveda pomembno. Kot zadnji kriterij funkcije in druge zmogljivosti smo izbrali kar druge zmogljivosti (**15**), ki smo jih dalje razdelili v razmerju **70:30** glede na to, ali brskalnik ima slovenski vmesnik, in za katere operacijske sisteme je na voljo.

Kriterij podpore standardom smo razdelili na to, katere W3C standarde podpira (**20**), kako je opravil Acid2 test (**10**), Acid3 test (**25**), CSS3 test (**25**) in JavaScript test (**20**). Najbolj pomembna se nam zdita Acid3 in CSS3 testa, saj sta to zadnja testa, ki testirata najbolj napredne funkcije in zmogljivosti brskalnikov in s tem napovedujeta njihovo prihodnost. Acid2 test smo ocenili kot najmanj pomemben, saj gre za starejši test, ki pa ga ne gre zanemarjati. W3C standardi in JavaScript sta seveda osnova za delovanje brskalnika, zato smo ju ocenili primerno visoko.

CSS3 test smo dalje razdelili na to, kako brskalnik podpira najnovejše stilske predloge (**50**) in selektorje (**50**). Tudi JavaScript smo v enakem razmerju kot CSS3 test razdelili na podporo JavaScript funkcijam in tehnologijam ter na hitrost izvajanja JavaScript operacij.

#### 5.4. Analiza rezultata

Ko smo vse brskalnike, izmerjene rezultate in utežene attribute vnesli v program HiView3, smo dobili HiView3 model (slika 28, ko vnesemo rezultate meritev in uteži se črtkane črte spremenijo v polne). Opomniti moramo, da smo kriterije utežili tako, da bi test ustrezal povprečno izobraženemu računalničarju. Taka oseba da največ poudarka podpori standardom, podpori razvijalcem in varnosti. Nekoliko manj ga zanima privlačen uporabniški vmesnik in sama hitrost brskalnika, čeprav tudi to dvoje ne zanemari.

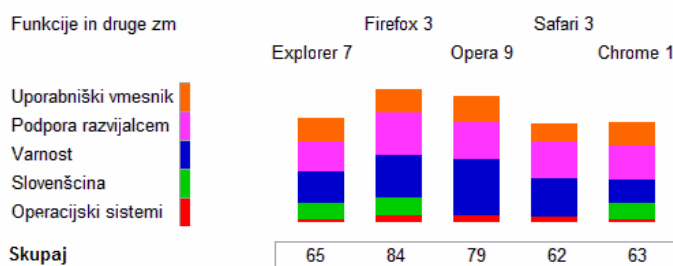


Slika 31: Grafični prikaz ocen spletnih brskalnikov.

Kot kaže slika 31 absolutnega zmagovalca ni. Zmaga sicer Firefox 3, vendar bi z nekoliko drugačnim modelom lahko zmagala tudi Opera 9 ali celo Safari 3. Po teh meritvah je za povprečnega računalničarja najbolj primeren brskalnik Firefox 3, zelo blizu mu sledi Opera 9. Safari 3 in Chrome 1 se odrežeta približno enako in si nekako delita tretje mesto, najslabše pa se odreže Explorer 7.

Za razlago in dodatno preverjanje rezultatov smo opazovali, koliko je k oceni posameznega brskalnika prispeval posamezen kriterij. Prispevek posameznega kriterija ponazarja legenda na levi strani slike 31.

Nekaj prednosti si Firefox 3 pred zasledovalcema Opero 9 in Safarijem 3 pridobi z dobro podporo razvijalcem spletnih aplikacij (slika 32), hitremu izvajanju JavaScript operacij, kvalitetnemu slovenskemu uporabniškemu vmesniku in pa popularnostjo med uporabniki. Ključ do zmage pa si Firefox 3 pridobi s svojo konstantnostjo, saj se je v vseh kriterijih izkazal za sprejemljivega. Tudi v kriterijih, kjer je sicer poraženec, nima izrazito slabih rezultatov. Internet Explorer 7 se žal izkaže za dokaj neprimerne, predvsem zaradi zelo slabe podpore spletnim standardom in tehnologijam. Nekoliko višje bi prišel v primeru, če bi ta test iskal najboljši brskalnik v kategoriji končnih uporabnikov, ki jih zanima predvsem hitrost, enostavnost in uporabniški vmesnik.

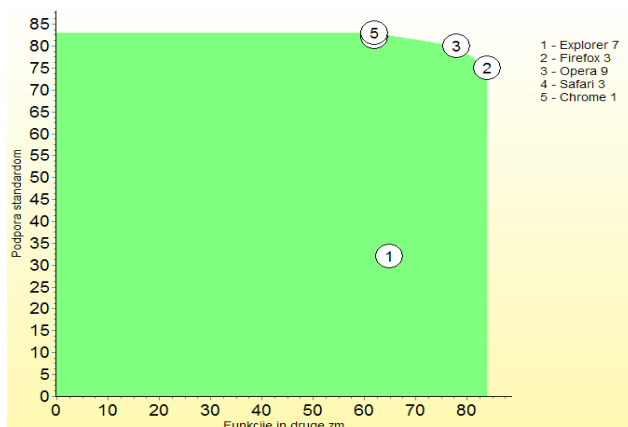


Slika 32: Razlog za zmago brskalnika - funkcije in druge zmogljivosti.

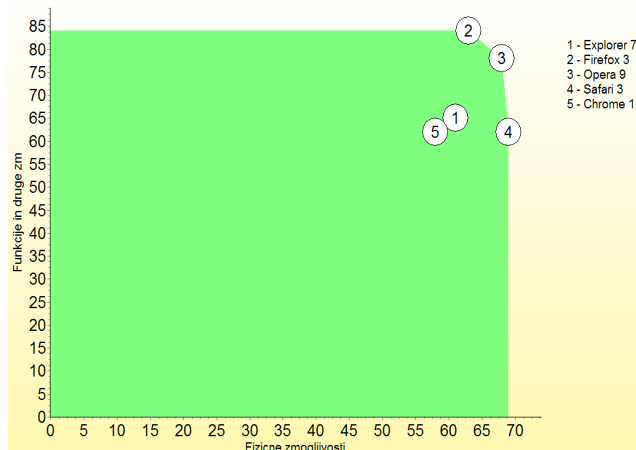
HiView3 omogoča pregled nad tem, koliko doprinese posamezen kriterij k dobri oziroma slabi uvrstitvi brskalnika. Primer tega lahko vidimo na sliki 33, ki kaže primerjavo brskalnikov po dveh najbolj pomembnih kriterijih, in sicer podpori standardom in funkcijam brskalnika. Tu lahko vidimo, da sta se Firefox 3 in Opera 9 odrežala podobno dobro. Firefox 3 se v primerjavi z Opero 9 nekoliko bolje odreže pri podpori uporabniku, Opera 9 pa pri podpori standardom.

Iz tega lahko razberemo, da bomo najlažje in najhitreje izdelovali ter pregledovali spletne aplikacije s Firefox-om 3, spletni standardi pa se bodo bolj kot Firefox-a 3 razveselili Opere 9. Kljub temu, da sta Safari 3 in Chrome 1 nesporna zmagovalca podpore standardom, skupno nista najboljša, saj jima nekoliko preveč zmanjka pri podpori uporabniku, podpori razvijalcem, kvalitetnemu uporabniškemu vmesniku in pa podpori varnosti. Iz slike je torej razvidno, da je najenostavneje izdelovali in pregledovali spletne aplikacije z brskalnikom Firefox 3, spletne standarde in tehnologije pa bo najbolje upošteval Chrome 1.

Na sliki 34 smo primerjali brskalnike med seboj še iz stališča povprečnega uporabnika spleta. To je uporabnik, ki ga zanima predvsem hitrost odpiranja strani, varnost in pa seveda dober in funkcionalen uporabniški vmesnik. Tukaj lahko vidimo, da sta se ravno tako najbolje odrezali Opera 9 in Firefox 3, nekoliko več zaostane Safari 3. Chrome 1 pa se tokrat odreže slabše kot Explorer 7, predvsem zaradi njegovega nenavadnega uporabniškega vmesnika, ki povprečnemu uporabniku spleta mogoče ne bo najbolj všeč. Veliko izgubi tudi zaradi velikih zahtev po pomnilniku.



Slika 33: Primerjava dveh najbolj pomembnih kriterijev.



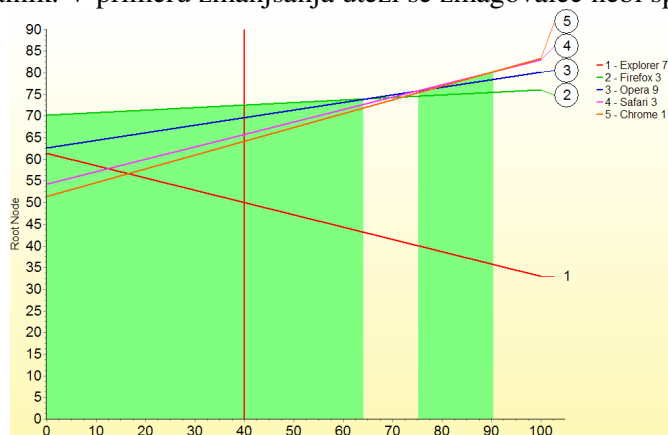
Slika 34: Primerjava dveh pomembnih kriterijev.

Program HiView3 omogoča tudi analizo občutljivosti za posamezne kriterije. S to metodo lahko analiziramo, kaj bi se dogajalo z razvrstitvami brskalnikov, če bi utež nekega kriterija zmanjšali oziroma jo povečali.

Na sliki 35 lahko vidimo analizo občutljivosti po najpomembnejšem atributu - podpori standardom. Presečišča poševnih črt brskalnikov z navpično rdečo črto označujejo oceno, ki jo je brskalniki dobili glede na uteženost atributa podpora standardom (40). S slike lahko razberemo, da bi se zmagovalca zamenjala, ko bi bila utež težka približno 64 točk. Na tem mestu bi vodstvo Firefox-u 3 prevzela Opera 9. Ob vrednosti uteži 75 pa bi vodstvo prevzela Chrome 1 in Safari 3.



To priča o tem, da je model dokaj neobčutljiv na uteži, saj bi morali pomembnost atributa kar precej povečati (povečati utežitev iz vrednosti 40 na vrednost 65), da bi zmagovalni prestol prevzel drugi brskalnik. V primeru zmanjšanja uteži se zmagovalec nebi spremenil.



Slika 35: Analiza občutljivosti za najpomembnejši atribut - podpora standardom.

## 5.5. Prihajajoče verzije brskalnikov

Avtorji v diplomski nalogi preverjenih brskalnikov so že nekaj časa nazaj dosegli kakovost svojega brskalnika, s katero je večina uporabnikov zadovoljna, zato je lahko povprečen uporabnik zadovoljen praktično z vsemi brskalniki, ki smo jih preverjali. Povprečen uporabnik namreč ne opazi bistvene razlike med brskalniki, ki se kaže predvsem pri podpori standardom, varnosti in podori razvijalcem spletnih strani.

Teh razlik pa se zavedajo avtorji brskalnikov, ki neprestano razvijajo popravke in dodatke za svoje brskalnike. S časoma dobivamo nove verzije brskalnikov, ki poskušajo zadovoljiti najbolj zahtevne uporabnike. Avtorji, katerih brskalniki niso deležni večje razširjenosti med uporabniki navadno izdajajo nove verzije pogosteje, kot tisti, katerih brskalniki so bolj razširjeni. To je razvidno tudi v našem testu, ko je najbolj razširjen brskalnik Internet Explorer najbolj redko posodobljen, kar se kaže v najslabši podpori standardom, varnosti in razvijalcem spletnih strani. Na drugi strani so brskalniki, ki so najmanj razširjeni, najbolj v tem delu našega testa, kar je razumljivo, če vemo, da so najmanj razširjeni brskalniki deležni največ novih verzij. Preden brskalnik preide v novo končno verzijo, je vmes deležen številnih pod verzij, navadno označenih kot »alfa« (dokaj nestabilna verzija), »beta« (dokaj stabilna verzija) in na koncu končna verzija.

Med pisanjem diplomske naloge, so avtorji preverjenih brskalnikov izdali naslednje nove verzije, ki smo jih na kratko tudi preizkusili:

- Internet Explorer 8 z oznako možnega kandidata številka ena za končno verzijo,
- Mozilla Firefox 3.1 z oznako beta 2,
- Opera 10 z oznako alfa,
- Apple Safari 4 z oznako beta in
- Google Chrome 2 z oznako alfa.

Nobena izmed novih verzij ni imela oznake končne verzije, zato kratkih testov, ki smo jih opravili ne gre jemati kot prave pokazatelje trenutnega stanja, lahko pa nam orišejo morebitno prihodnost brskalnikov.

Explorer 8 je končno opravil Acid2 test, ki je pogoj za pravilno izrisovanje standardnih spletnih strani. Kljub temu ni vidnega napredka v Acid3 testu, kjer opravi le 20 podtestov od 100, kar je za 6 več kot predhodnik. V izvajanju JavaScript-a je nekoliko hitrejši, in sicer za 26 odstotkov, na testu CSS3 selektorjev pa doseže 19 točk več kot verzija 7. Uporabniški vmesnik ni bil deležen bistvenih sprememb.

Firefox 3.1 je napredoval na vseh področjih podpore standardov. V Acid3 testu opravi 93 podtestov, kar je v primerjavi z 71 opravljenimi podtesti predhodnika dober napredek. V testu CSS3 selektorjev je napredek še večji, saj Firefox 3.1 opravi vse podteste. Tudi test hitrosti izvajanja JavaScript-a je podal bistveno boljše rezultate kot aktualna verzija in to za 200 odstotkov, s čimer se nekoliko bolj približala Chromu.

Opera 10 doseže v Acid3 testu 100 odstotni izkupiček, v testu hitrosti izvajanja JavaScript-a pa je boljša le za 20 odstotkov v primerjavi z Opero 9.

Tudi Safari 4 doseže v Acid3 testu 100 odstotni izkupiček, v testu hitrost izvajanja JavaScript-a pa doseže največjo pohitritev med vsemi, in sicer 270 odstotno, kar ga zelo približa Chromu. Safari 4 edini med novimi verzijami dobi novo preobleko. Nov uporabniški vmesnik ima izboljšave pri zavihkih (preselili so se v najvišjo vrstico in s tem še povečali okno, ki je namenjeno pregledovanju strani), okno z zgodovino obiskanih strani (sedaj je vizualno lepše in bolj funkcionalno) in pri samem izgledu, ki smo ga očitali pri verziji 3 (sedaj je bolj standarden).

Chrome 2 je pri testu hitrosti izvajanja JavaScript-a zopet presenetil. Kljub temu, da ni nobeden izmed novih testiranih brskalnikov dosegel rezultata Chroma 1, je Chrome 2 še hitrejši, in sicer za 40 odstotkov. Tudi Chrome 2 sedaj opravi Acid3 test 100 odstotno.

Končni rezultati so pokazali, da so se nove verzije brskalnikov odrezale bolje kot starejše. Če bi na osrednjem testu v diplomi testirali novejše nedokončane verzije namesto starejših končnih, se vrstni red nebi bistveno spremenil, saj novejše verzije nimajo bistvenih izboljšav drugje, kot na področju standardov, kjer pa so vse napredovale podobno, kot so napredovale že prejšnje verzije. Takšen test bi le nekoliko bolj stisnil rezultate, predvsem Explorer bi lahko dobil oceno namesto pogojno sprejemljiv, sprejemljiv, saj bi dosegel več točk kot 50.

## 6. Sklepne ugotovitve

V diplomski nalogi smo ugotovili, da si vsi brskalniki zaslužijo pozitivno oceno, saj so bili na koncu vsi ocenjeni z oceno 50 ali več (slika 31), kljub temu, da smo kriterije postavili strogo, kot da bi izbirali brskalnik za zahtevnega uporabnika in razvijalca spleta. Od brskalnikov smo zahtevali maksimalno upoštevanje standardov, odlično podporo razvijalcem spletnih aplikacij in dobro varnost. Prav tako nismo zanemarili zahtev povprečnih uporabnikov, ki si želijo predvsem hitrega, enostavnega in uporabniku prijaznega brskalnika.

Rezultat testa je bil zelo izenačen, predvsem na vrhu, kjer sta bila Firefox in Opera precej izenačena, sledila sta jima Safari in Chrome s prav tako zelo izenačenim rezultatom. Nekoliko več je izostal le Explorer, ki je dosegel komaj sprejemljiv rezultat, in sicer 50 točk. Tako bi lahko nekoliko drugačen model utežitve zamenjal zmagovalca ali celo Explorer označil za neprimernega. Če bi iskali brskalnik za povprečnega uporabnika spleta, bi se rezultati še nekoliko bolj izenačili, saj bi Explorer nekoliko manj izgubil pri standardih, ostali pa bi pri standardih nekoliko manj pridobili.

Za konec smo na kratko pogledali še zadnje verzije preverjenih brskalnikov, ki nimajo še oznake »končana« in naredili nekaj osnovnih testov. Tudi zadnje verzije brskalnikov so se razvrstile v podoben vrstni red, saj je posamezen brskalnik v primerjavi z njegovo aktualno verzijo napredoval podobno kot ostali.

Tako lahko v prihodnosti pričakujemo še naprej hud boj za zmagovalni prestol med brskalnik in vedno bolj popolne brskalnike. Določena podjetja bodo naprej ciljala na čim večjo razširjenost brskalnika med povprečnimi uporabniki, kot to delajo pri Microsoftu in njihovem Explorer-ju. Druga bodo ciljala na nekoliko bolj zahtevne uporabnike, kot to počne Mozilla s svojim Firefox-om, kateremu se bo najbrž pridružil še Google Chrome. Nekateri pa bodo ciljali na specifične uporabnike, kot to počnejo pri Applu z njihovim Safari-jem in pri Operi.

Če se bo število zahtevnejših uporabnikov povečalo, lahko pričakujemo zamenjavo v vodstvu priljubljenosti, kjer lahko prestol Explorer-ja prevzameta Firefox in čez nekaj let celo Chrome.

## Dodatek A

### Seznam slik

Slika 1: Primer HTML zapisa.....	4
Slika 2: Primer XML zapisa.....	5
Slika 3: Primer XHTML zapisa.....	6
Slika 4: Primer CSS-ja.....	7
Slika 5: Primer spletne strani, ki omogoča RSS.....	8
Slika 6: Primer EcmaScript-e.....	8
Slika 7: Hierarhija objektov v primeru modela DOM.....	9
Slika 8: Razširjenost internetnih brskalnikov v Sloveniji.....	13
Slika 9: Razširjenost internetnih brskalnikov v svetu.....	14
Slika 10: Hitrost zagona brskalnikov v sekundah.....	15
Slika 11: Hitrost zagona internetnih strani za posamezen brskalnik v sekundah.....	16
Slika 12: Povprečna hitrost zagona internetne strani za posamezen brskalnik v sekundah.....	17
Slika 13: Poraba pomnilnika v kilo bajtih za posamezen brskalnik.....	18
Slika 14: Internet Explorer 7 Developer Toolbar.....	23
Slika 15: Mozilla Firefox 3 Web Developer Toolbar.....	24
Slika 16: Mozilla Firefox 3 Firebug.....	24
Slika 17: Mozilla Firefox 3 Tamper Data.....	24
Slika 18: Apple Safari 3 Developer.....	25
Slika 19: Opera 9 Developer orodna vrstica.....	26
Slika 20: Opera 9 Developer Tools.....	26
Slika 21: Google Chrome Rzvijalec.....	27
Slika 22: Acid2 test.....	33
Slika 23: Pravilno izrisan smeško Acid2.....	34
Slika 24: Nepravilno izrisan smeško Acid2.....	35
Slika 25: Acid3 test.....	35
Slika 26: Izmerjen čas posameznih JavaScript operacij.....	44
Slika 27: Vsota in povprečje časov izvajanja JavaScript operacij.....	45
Slika 28: Hierarhični model v programu HiView3.....	47
Slika 29: Okno za vnos izmerjenih vrednosti za merilo razširjenost brskalnika v Sloveniji... 48	48
Slika 30: Primer utežitve kriterija v programu HiView3.....	53
Slika 31: Grafični prikaz ocen spletnih brskalnikov.....	54
Slika 32: Razlog za zmago brskalnika - funkcije in druge zmogljivosti.....	55

	61
Slika 33: Primerjava dveh najbolj pomembnih kriterijev. ....	56
Slika 34: Primerjava dveh pomembnih kriterijev. ....	56
Slika 35: Analiza občutljivosti za najpomembnejši atribut - podpora standardom. ....	57

## ***Seznam tabel***

Tabela 1: Izmerjeni časi odpiranja spletnih strani za posamezen brskalnik v sekundah. ....	16
Tabela 2: Poraba pomnilnika v kilo bajtih za posamezen brskalnik. ....	17
Tabela 3: Podprtost brskalnikov operacijskim sistemom in slovenskemu vmesniku. ....	31
Tabela 4: Podpora W3C standardom posameznega brskalnika. [31] ....	33
Tabela 5: Rezultati Acid3 testa za posamezne brskalnike. ....	36
Tabela 6: Število opravljenih CSS3 selektor testov za posamezen brskalnik. ....	41
Tabela 7: Podpora potencialnim zmogljivostim CSS3-ja posameznega brskalnika. ....	42
Tabela 8: Podpora JavaScript tehnologijam posameznega brskalnika. [31] ....	43
Tabela 9: Test delovanja JavaScript-a v brskalnikih. ....	44
Tabela 10: Ocene kriterijev za posamezen brskalnik. ....	49

## Literatura

- [1] Janez Demšar. Prosojnice predavanj pri predmetu Standardizacija in kakovost informacijskih sistemov: Splošno o standardih. Fakulteta za računalništvo in informatiko, Ljubljana, 2007.
- [2] Standardizacija. Vir na spletu:  
<http://sl.wikipedia.org/wiki/Standardizacija>
- [3] Začetek spleta. Vir na spletu:  
<http://sl.wikipedia.org/wiki/Splet>
- [4] Namen W3C-ja. Vir na spletu:  
<http://www.w3.org/Consortium/>
- [5] W3C. Vir na spletu:  
[http://sl.wikipedia.org/wiki/World\\_Wide\\_Web\\_Consortium](http://sl.wikipedia.org/wiki/World_Wide_Web_Consortium)
- [6] HTML. Vir na spletu:  
<http://en.wikipedia.org/wiki/Html>
- [7] Spletni brskalnik Mosaic. Vir na spletu:  
[http://en.wikipedia.org/wiki/Mosaic\\_\(web\\_browser\)](http://en.wikipedia.org/wiki/Mosaic_(web_browser))
- [8] Ryan A. MacMichael. Cleaning Up the Clutter: Why Web Standards Matter. Vir na spletu:  
<http://ieeexplore.ieee.org/Xplore/guesthome.jsp>
- [9] XML. Vir na spletu:  
<http://sl.wikipedia.org/wiki/Xml>
- [10] XHTML. Vir na spletu:  
<http://sl.wikipedia.org/wiki/XHTML>
- [11] Kako mora bit zgrajen XHTML dokument. Vir na spletu:  
<http://www.webstandards.org/learn/faq/>
- [12] CSS. Vir na spletu:  
[http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets)
- [13] RSS. Vir na spletu:  
<http://www.monitor.si/clanek/doba-peskovnikov-je-tu/>
- [14] ECMAScript. Vir na spletu:  
<http://en.wikipedia.org/wiki/ECMAScript>
- [15] Model DOM. Vir na spletu:  
<http://www.w3.org/DOM/>

- [16] Spletni brskalnik. Vir na spletu:  
[http://en.wikipedia.org/wiki/Web\\_browser](http://en.wikipedia.org/wiki/Web_browser)
- [17] Predstavitev Internet Explorer-ja. Vir na spletu:  
<http://www.microsoft.com/windows/WinHistoryIE.msp>
- [18] Predstavitev Internet Explorer-ja. Vir na spletu:  
[http://en.wikipedia.org/wiki/Windows\\_Internet\\_Explorer](http://en.wikipedia.org/wiki/Windows_Internet_Explorer)
- [19] Predstavitev Mozilla Firefox. Vir na spletu:  
[http://sl.wikipedia.org/wiki/Mozilla\\_Firefox](http://sl.wikipedia.org/wiki/Mozilla_Firefox)
- [20] Prestavitev Opere. Vir na spletu:  
<http://www.opera.com/company/>
- [21] Prestavitev Opere. Vir na spletu:  
[http://en.wikipedia.org/wiki/Opera\\_browser#cite\\_note-38](http://en.wikipedia.org/wiki/Opera_browser#cite_note-38)
- [22] Prestavitev Safarija. Vir na spletu:  
[http://en.wikipedia.org/wiki/Safari\\_\(web\\_browser\)](http://en.wikipedia.org/wiki/Safari_(web_browser))
- [23] Predstavitev Chroma. Vir na spletu:  
[http://en.wikipedia.org/wiki/Chrome\\_\(browser\)](http://en.wikipedia.org/wiki/Chrome_(browser))
- [24] Podatki o uporabi internetnih brskalnikov v Sloveniji. Vir na spletu:  
<http://www.iprom.si/press.html?id=141>
- [25] Podatki o uporabi internetnih brskalnikov v svetu. Vir na spletu:  
[http://www.w3schools.com/browsers/browsers\\_stats.asp](http://www.w3schools.com/browsers/browsers_stats.asp)
- [26] Internet Explorer in razvijanje aplikacij. Vir na spletu:  
<http://www.microsoft.com/downloadS/details.aspx?familyid=E59C3964-672D-4511-BB3E-2D5E1DB91038&displaylang=en>
- [27] Mozilla Firefox Web Developer. Vir na spletu:  
<https://addons.mozilla.org/en-US/firefox/addon/60>
- [28] Mozilla Firefox Firebug. Vir na spletu:  
<http://getfirebug.com/>
- [29] Mozilla Firefox Temper Data. Vir na spletu:  
<https://addons.mozilla.org/en-US/firefox/addon/966>
- [30] Apple Safari Develop meni. Vir na spletu:  
<http://developer.apple.com/internet/safari/faq.html#anchor2>
- [31] Veliko o brskalnikih. Vir na spletu:  
[http://en.wikipedia.org/wiki/Comparison\\_of\\_web\\_browsers](http://en.wikipedia.org/wiki/Comparison_of_web_browsers)

- [32] Varno deskanje z Internet Explorer-jem 7. Vir na spletu:  
<http://windowshelp.microsoft.com/Windows/sl-SI/help/771b599b-cabd-46f0-a0b4-b82c3bcd3d461060.mspx#EQD>
- [33] Varno deskanje z Mozillo Firefox 3. Vir na spletu:  
<http://www.mozilla.com/en-US/firefox/features/>
- [34] Acid2 test. Vir na spletu:  
<http://www.webstandards.org/action/acid2/>
- [35] Acid3 test. Vir na spletu:  
<http://www.webstandards.org/action/acid3>
- [36] Acid3 test. Vir na spletu:  
<http://en.wikipedia.org/wiki/Acid3>
- [37] Uvod v CSS3. Vir na spletu:  
[http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets)
- [38] Selektorji v CSS3. Vir na spletu:  
<http://www.css3.info/selectors-test/test.html>
- [39] Nekatere funkcije in zmogljivosti bodočega CSS3. Vir na spletu:  
<http://www.css3.info/preview/>
- [40] Program za analizo podatkov, HiView3. Vir na spletu:  
<http://www.catalyze.co.uk/?id=230>
- [41] Zaporedja pri delu s programom HiView3. Vir na spletu:  
<http://wd-testnet.world-direct.at/mozilla/dhtml/funo/jsTimeTest.htm>