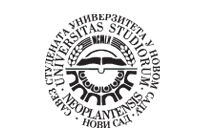
****

Tehnički fakultet “Mihajlo Pupin” Zrenjanin

**SEMINARSKI RAD**

iz predmeta Veb Dizajn

**TEMA: Web sajt “GameShop”**

Profesor: **prof. dr Branko Markoski** Student: **Dejan Nadlački**

Asistent: **dr. Predrag Pecev** Broj indeksa: 66/10-10

Smer: IT - Inženjerstvo

Zrenjanin Septembar 2018.

Contents

[Predmet 2](#_Toc525677279)

[Cilj 2](#_Toc525677280)

[Apstrakt 2](#_Toc525677281)

[UVOD 3](#_Toc525677282)

[PHOTOSHOP 4](#_Toc525677283)

[Alatke u Photoshop-u 4](#_Toc525677284)

[HYPERTEXT MARKUP LANGUAGE (HTML) 7](#_Toc525677285)

[Istorija HTML-a 7](#_Toc525677286)

[HTML dokument 8](#_Toc525677287)

[Sintaksa HTML elemenata 8](#_Toc525677288)

[Struktura HTML dokumenta 9](#_Toc525677289)

[HTML tagovi tela dokumenta i njihov opis 10](#_Toc525677290)

[HTML 5 14](#_Toc525677291)

[Novi Elementi i Karakteristike 15](#_Toc525677292)

[Multimedia 16](#_Toc525677293)

[CASCADING STYLE SHEETS (CSS) 17](#_Toc525677294)

[Sintaksa 18](#_Toc525677295)

[Boje 18](#_Toc525677296)

[Korisni alati za CSS 19](#_Toc525677297)

[Vrste selektora 20](#_Toc525677298)

[*Slika 4. Prikaz box modela* 21](#_Toc525677299)

[Načini implementacije stilova 21](#_Toc525677300)

[Prednosti upotrebe CSS-a u web dokumentima 23](#_Toc525677301)

[ŠTA JE JAVASCRIPT 23](#_Toc525677302)

[JAVASCRIPT UNUTAR WEB BROWSERA 24](#_Toc525677303)

[KLJUČNE REČI I TIPOVI PODATAKA 24](#_Toc525677304)

[VARIJABLE I LITERALI 25](#_Toc525677305)

[OPERATORI I IZRAZI 27](#_Toc525677306)

[MOGUĆNOST I UPOTREBA JAVASCRIPT-A 28](#_Toc525677307)

[IMPLEMENTACIJA JAVASCRIPTA 29](#_Toc525677308)

[PRIMERI 29](#_Toc525677309)

[KORIŠĆENJE SPOLJAŠNJEG JAVASRIPTA 31](#_Toc525677310)

[jQuery 31](#_Toc525677311)

[jQuery Jezgro 31](#_Toc525677312)

[jQuery Selektori 32](#_Toc525677313)

[Flash 34](#_Toc525677314)

[Scena 36](#_Toc525677315)

[Vremenska Linija 36](#_Toc525677316)

[Paneli 37](#_Toc525677317)

[Alatke u Flash-u 39](#_Toc525677318)

[Kreiranje Web Sajta 41](#_Toc525677319)

[Kreiranje početne strane 41](#_Toc525677320)

[Kreiranje stranice “Sve Igre” 41](#_Toc525677321)

[Kreiranje stranice “Žanrovi” 42](#_Toc525677322)

[Kreiranje stranice “Platforme” 43](#_Toc525677323)

[Kreiranje stranice “Kontakt” 43](#_Toc525677324)

[Zaključak 44](#_Toc525677325)

[Literatura 44](#_Toc525677326)

[Prilog 44](#_Toc525677327)

# Predmet

Da student ovlada osnovnim tehnologijama koje se koriste pri izradi Web sajtova . Web dizajn je predmet koji studentima omogućava da steknu neka osnovna načela koja se koriste pri dizajniranju Web sajtova.

# Cilj

Cilj seminarskog rada jeste upoznavanje i ovladavanje tehnikama koje se koriste pri dizajniranju Web sajtova, HTML, CSS, JAVASCRIPT. Cilj predmeta je da student čije je opredeljenje Web dizajn dalje upozna sa metodologijma i alatima koji se koriste .

# 

# Apstrakt

Ovaj rad nastao je kao rezultat seminarskog rada iz Web-Dizajna. U seminarskom radu predstavljen je sajt, koji je urađen korišćenjem različitih tehnologija (HTML, CSS, JAVASCRIPT, PHOTOSHOP ). O svakoj tehnologiji biće reči u daljem izlaganju , i na samom kraju biće predstavljen i sam sajt.

# UVOD

Web dizajn predstavlja grafičko osmišljavanje ili dizajniranje internet stranice koristeći slike, jezik za označavanje i formiranje „HTML”, standardizaciju i upotrebljivost stranice.Razumna upotreba teksta i multimedijalnih elemenata (slika, animacije, zvuka), tj. prilagodjenost sadrzaja nameni web strane.Korisnici se ne smeju nervirati prevelikim sadrzajem koji se dugotrajno ucitava, mora se voditi racuna o znacenju boja u kulturi naroda, vecu kolicinu sadrzaja raspodeliti na vise stranica.Prema nacinu izrade, stranice se dele na staticke I dinamicke. Staticke web strane uradjene su kao zasebni fajlovi i celine, od kojih svaka ima svoju oznaku (naziv) i linkovima (vezama) su povezane sa ostalim stranama u okviru sajta kao celine, a multimedijalnim objektima (slikama, audio datotekama) I sa stranama koje se nalaze na web-u. Dinamicke web strane pretpostavljaju programerska znanja I postojanje baze podataka u okviru memorijskog prostora web servera na koji je sajt postavljen (hostovan), Ovaj tip strane pored HTML koda, koji sluzi da bi tekst I multimedijalne sadrzaje koji se objavljuju na strani korisnika prikazivao na odredjenom mestu, sadrzi I skript jezik (npr. php ili asp) za povezivanje sa bazom podataka na serveru. Konekcija sa bazom podataka omogucava da se na strani korisnika generise sadrzaj u obliku web strane. Dinamicku web stranu prepoznajemo po sintaksi sadrzanoj u adresnom polju – “?” i “=”.

# PHOTOSHOP

Photoshop je grafički program koji se razvili stručnjaci Adobe-ovog kreatingo tima za Windows i Macintosh platform. To je program za dijaniranje i slika i fotografija i kao takav pretežno se koristi od strane grafičkig dizajnera. Dopušta manipulaciju sa slikama i stvaranje specijalnih efekata , a na naprednijoj veriji moguće je od mnogo pojedinačnih slika načiniti složene slike kojima se mogu promeniti boje i po želji im uređivati pozadinu. Premda se uglavnom koristi za retuširanje i manipulaciju fotogorafijama može se koristiti i za stvaranje slika od početka ili na temelju postojeće fotografije. Photoshop je vrlo moćan program za stvaranje uređivanje sa neverovatnim postavkama i upravo zbog velikih mogućnosti koje nudi vrlo je teško u kratkom vremenu temeljno ga proučiti pa ćemo se bazirati na opis rada s osnovnim alatima, a na primeima objasniti neke naprednije alate . Takođe treba imati na umu da često trebamo koristititi i nekoliko različitih alata da bismo stvorili željene efekte, a mnogi od tih alata da bi ih se uspešno koristili, zahtevaju vežbanje i upotrebu mašte.

## Alatke u Photoshop-u

|  |  |
| --- | --- |
| ***Rectangle Marguee Tool*** | Selektuje pravougaone oblasti na slici.  Shift+ pritisnut taster dobija se kvadrat |
| ***Elliptical Marquee Tool*** | Selektuje eliptične oblasti slike(elipsa).  Alt+ pritisnut taster rešava problem selekcije od ugla do ugla.  Shift+ pritisnut taster dobija se krug |
| ***Single Row Marquee Tool*** | Selektuje linju jedan piksel po visini. |
| ***Single Row Column Tool*** | Selektuje linju jedan piksel po širini. |
| ***Lasso Tool*** | Selekcija slobodnom rukom |
| ***Polygonal Lasso Tool*** | Selekciju poligona(geometriskih oblika sa više ravnih strana) |
| ***Magnetic Lasso Tool*** | Selekcija se lepi za objekat koji obeležavate |
| ***Quick Selection Tool*** | Selekcija se vrši automatskih pronalaženjem i praćenjem definisanih ivica. |
| **Crop Tool** | Selektovani deo ostaje nepromenjen, zatamljeni deo biće odsečen. |
| **Eyedropper Tool** | Postavlja boju prednjeg plana ili pozadinsku boju |
| **Color Sampler Tool** | Postavlja četiri tačke preko koje možemo pratiti boje i njihove promene dok radimo sa njima. |
| **Spot Healing Brush Tool** | Uklanja nedostatke. Prelepljuje piksele i automatski podešava tastaturu, osvetljenje i senčenje sa onim koje prelepljuje. |
| **Healing Brush Tool** | Klonira piksele koje sami izaberemo |
| **Patch Tool** | Automatski prilagođava tastaturu, osvetljenje i senčenje uzrokovanih piksela onima koje prilepljujemo |
| **Red Eye Tool** | Rešava problem crvenih očenja tako što ih potamljuju. |
| **Brush Tool i Pencil Tool** | Klasični alati za crtanje. razlika između ova dva alata je da četkica (brush) ocrtava to jest boji sa mekanim ivicama dok Pencil to jest olovka  boji sa tvrdim ivicama |
| **Mixer Brush Tool** | simulira realistične tehnike crtanja kao što je mešanje boja na platnu i u četkici kao i vlažnost boja. Rezervoar sadrži boju kojom bojite dok preuzimač boje preuzima boju sa platna i meša ih sa onom iz rezervoara |
| ***Clone stamp tool*** | duplira (klonira) deo slike ili uklonja ono što ne želite da se vidi na slikama. |
| ***Pattern Stamp Tool*** | boji delove slike (Pattern) koji odaberete na traci sa opcijama. |
| ***Eraser Tool*** | Služi za brisanje piksela |
| **Background Eraser** | Briše piksele sa slike i ima dodatne opcije za podešavanje koje omogućavaju dodatne kvalitete za pravljenje kompozicija slika. |
| **Magic Eraser Tool** | Ovaj alat je automatizovan i na jedan klik briše sve piksele koji su slični. |
| **Gradient Tool** | Postavlja odabrani gradiant na deo koji odaberemo. |
| **Paint Bucket Tool** | služi za popunjavanje bojom ili obrascem (pattern). |
| **Blur Tool** | omekšava ivice a kada se primenjuje u većoj meri dovodi do postepenog gubljenja detalja na tim delovima slike. |
| **Sharpen Tool** | Radi suprotnu stvar od Blur tool. |
| **Smudge Tool** | Pravi efekat. Uzima boju sa slike i gura je u pravcu u kome vi gurate taster miša ili olovku na tabli za crtanje. |
| **Dodge i Burn tool** | Ova dva alata koriste četkicu tako da prvo treba podesiti parametre četkice kao što su tip, veličina, tvrdoća... |
| **Sponge Tool** | Menja intezitet boje. |
| **Horizontal Type Tool** | Unos teksta po horizontali |
| **Vertical Type Tool** | Unos teksta po vertikali |
| **Horizontal Type Mask Tool** | Pravi horizontalnu privremenu masku koja će biti pretvorena u privremenu selekciju kada završite sa  kucanjem |
| **Vertical Type Mask Tool** | Pravi vertikalnu privremenu masku koja će biti pretvorena u privremenu selekciju kada završite sa  kucanjem |
| **Rectangle Tool** | Crta pravougaonik ili kvadrat. |
| **Rounded Rectangle Tool** | Crta pravougaonik ili kvadrat sa zaobljenim ivicama. |
| **Ellipse Tool** | Crta elipsaste oblike ili krug. |
| **Polygon Tool** | Crta figure sa najmanje 3 a najviše 100 strana. Pomoću ovog alata možete nacrtati i zvezdu. |
| **Line tool** | Crta liniju. |
| **Custom shape tool** | Crta najrazličitije oblike. možete kreirati svoje ili preuzeti sa interneta. |

# HYPERTEXT MARKUP LANGUAGE (HTML)

HTML nije programski jezik već jednostavan meta-jezik za standardizovano strukturiranje web dokumenata. Web dokument je, dakle, u potpunosti tekstualni dokument koji se može kreirati u bilo kojem tekst editoru. Sastoji se od označnih elemenata - tagova, unutar kojih se smešta sadržaj. Tipom taga određuje se (označava) vrsta sadržaja taga, tj. da li je nositelj informacije unutar elementa paragraf, zaglavlje, tabela itd.

Internet browser nakon učitavanja web dokumenta parsira dobijeni tekst, interpretira tagove i njihov sadržaj prema specifikaciji HTML standarda, te ga na kraju prikazuje kao celinu, to jest web stranicu.

Web dokumente karakteriše jedinstvena osobina koja ih fundamentalno razlikuje od ostalih vrsta elektronskih dokumenata, a to je njihova lokacija na internetu, odnosno jedinsveni identifikator te lokacije - URL (Uniform Resource Locator). Time web dokumenti donose mnoge prednosti u odnosu na klasične elektronske dokumente. Neke od tih prednosti su:

* mobilnost pristupa, s obzirom da se web dokumentima može pristupati s bilo koje lokacije i bilo kojeg uređaja
* mogućnost da se unutar dokumenta definiše veze prema drugim dokumentima jednostavnim navođenjem njihovih URL-ova
* razmena i kolaboracija
* mogućnost centraliziranog, serverski kontrolisango arhiviranja

## Istorija HTML-a

Prva ideja iz koje se razvio World Wide Web nastala je 1989. godine kao posledica potrebe za razmenom informacija. U to vreme rad na mreži koristili su uglavnom naučnici koji su za korištenje mrežnih usluga morali poznavati veliki broj Unix naredbi. Tim Berners-Lee u CERN-u je predložio stvaranje hipertekstualnog sistema koji će omogućiti jednostavnu razmenu informacija među stručnjacima iz cijelog svijeta.

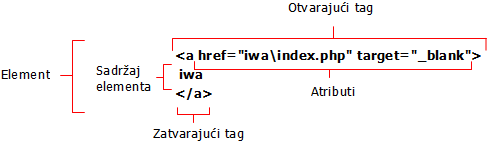
# HTML dokument

HTML dokument je tekstualna datoteka, te ju je moguće kreirati bilo kojim tekst editorom. Osim tekstualnog sadržaja dokumenta, u HTML datoteci koriste se specijalne sekvence karaktera kojima se opisuje struktura i značenje pojedinih delova dokumenta.

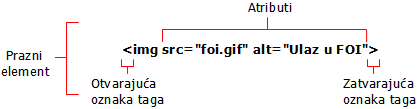
## Sintaksa HTML elemenata

Osnovna strukturna jedinica HTML dokumenta jest HTML element, koji se sintaksno opisuje na sljedeći način:

* počinje otvarajućim tagom (opening tag)
* završava zatvarajućim tagom (closing tag)
* sadržaj elementa je sve što se nalazi između otvarajućeg i zatvarajućeg taga
* neki elementi imaju prazan sadržaj
* prazni elementi (bez sadržaja) zatvoreni su u otvarajućem tagu
* većina HTML elemenata može imati atribute koji dodatno opisuju specifičnosti pojedinih tagova



Neki tagovi ne mogu imati sadržaj, to su tzv. "prazni elementi" i kao takvi oni nemaju zatvarajući tag.



HTML elementi mogu imati više različitih atributa; svaki atribut sastoji se od imena i vrednosti (ime="vrednost"), a specificira se u otvarajućem tagu elementa. Atributi daju browser-u dodatne informacije, specifične za pojedinu vrstu elementa.

Osim tagova za strukturiranje tekstualnih ili grafičkih informacija, pojedini tagovi služe grupisanje drugih tagova, tako nastaje hijerarhijska struktura HTML dokumenta.

Primer fragmenta HTML koda:

<div id="login">

<h4> Izgradnje web aplikacija</h4>

<form method=post action='index.php'>

<p>

<label for="email">e-mail:</label>

<input type="text" name="email" id="email"><br>

<label for="password">password:</label>

<input type="password" name="password" id="password"><br>

<input class="button" type="submit" value="Login" name="btnEnter" id="btnEnter">

</p>

</form>

</div>

Iako HTML dopušta nedoslednosti pri ugnježđivanju tagova, pa čak i izostavljanje zatvarajućih tagova, preporučljivo je držati se specifikacije pune sintakse.

## Struktura HTML dokumenta

Osnovna struktura HTML dokumenta takođe je jednostavna: započinje tagom deklaracije vrste dokumenta <!DOCTYPE>. Iza deklaracije sledi <html> tag unutar kojeg je smešten celi dokument, a završava sa zatvarajućom oznakom </html>. Dokument se sastoji od dva dela: <head> sekcije, koja sadrži opšte, nevizualne informacije, iza koje sledi <body> sekcija s vizualnim komponentama dokumenta.

Elementi HTML stranice ne zahtevaju razdvajanje nekim posebnim karakterom . Svaki tag ima definisanu oznaku za svoj početak i kraj, pa je sasvim moguće da se datoteka koja definiša sadržaj i izgled kompletne web stranice sastoji od jedne dugačke linije, bez ijednog karaktera razmaka. Naravno, u praksi je dobro težiti što boljoj čitkosti koda, pa je preporuka, naročito za duže tagove, izvorni kod stranice vizualno formatirati prateći model standardnog formatiranja blokova kod klasičnih proceduralnih programskih jezika.

<html>

<head>

<title>Naslov</title>

</head>

<body>

<h3>foi</h3>

<img src="slika1.png">

</body>

</html>

Unutar HTML dokumenta moguće je napisati komentare koji nemaju nikakav uticaj na izgled web stranice, ali mogu poslužiti za lakše snalaženje u kodu. HTML komentari, kao i tagovi, imaju oznaku za početak (<!--) i kraj (-->), mogu sadržavati razmake i tabove, pa se mogu protezati kroz više redaka:

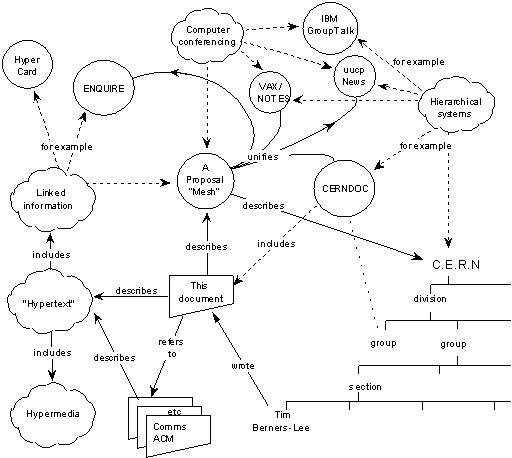
<html>

<body>

<!—Ovo je komentar

unutar HTML koda-->

</body>

</html>

*Slika 1. Originalni predlog za WorldWidwWeb kojeg je izradio Tim Berners-Lee (u samom početku imao je naziv 'Mesh')*

Berners-Lee je sa svojim timom kreirao HyperText Transfer Protocol (HTTP), protokol za standardizaciju komunikacije između klijenta i servera. 1990. godine tim je počeo razvijati projekt World Wide Web-a i specificirao označni jezik za kreiranje strukturiranih dokumenata koji trebaju biti prikazani na internetu: "HyperText Markup Language" ili skraćeno HTML. Tada je napisan i prvi tekstualni pretraživački program za web (web browser) jednostavnog naziva 'WorldWideWeb'. Smatra se da je prvi grafički web browser MOSAIC koji je nastao početkom 1993. godine, iako su mu prethodili neki manje poznati browser-i npr. Erwise i ViolaWWW.

## HTML tagovi tela dokumenta i njihov opis

HTML tagovi se mogu podeliti u dve glavne celine: tzv. 'blok-level' tagove i 'inline' tagove. Blok tagovi osim svoga sadržaja mogu imati unutar sebe ugnježdene druge blok ili inline tagove, dok inline tagovi mogu sadržavati isključivo tekst ili druge inline tagove.

U HTML-u, blok tagovi su:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| <address> | <blockquote> | <div> | <dd> | <dl> | <dt> | <fieldset> | <form> | <h1..h6> | <li> | <ol> |
| <p> | <pre> | <table> | <tbody> | <td> | <tfoot> | <th> | <thead> | <tr> | <ul> |  |

Svi ostali tagovi su inline.

Sledi puna lista svih dozvoljenih tagova u HTML 4 Strict definiciji, grupisano prema nameni odnosno vrsti sadržaja koju predstavljaju:

|  |  |
| --- | --- |
| **Tag** | **Značenje** |
| FORMATIRANJE TEKSTA | |
| <h1> do <h6> | Naslovi u dokumentu |
| <p> | Paragraf teksta |
| <br> | Novi red |
| <hr> | Horizontalna linija |
| tip teksta | |
| <b> | Podebljani tekst (bold) |
| <i> | Ukošeni tekst (italic) |
| <big> | Povećani tekst |
| <small> | Smanjeni tekst |
| <sub> | Smanjeni i spušteni tekst (indeks) |
| <sup> | Smanjeni i povišeni tekst (eksponent) |
| <blockquote> | Citirani blok teksta |
| <q> | Kratki tekst u navodnicima |
| <abbr> | Definiše kraticu |
| <acronym> | Definiše akronim |
| <bdo> | Menja predefinisani smer teksta u browser-u s "dir" atributom |
| <cite> | Definiše citat |
| <code> | Tekst kompjuterskog koda |
| <dfn> | Termin za definiciju |
| <em> | Istaknuti tekst |
| <strong> | Naglašeni tekst |
| <samp> | Primer kompjuterskog koda |
| <var> | Varijabla unutar teksta |
| <kbd> | Tekst tipke |
| <pre> | Preformatirani tekst (preslikava tekst tačno kako je napisan u kodu, zajedno s uzastopnim razmacima i prelomima linije; ima fiksnu širinu fonta) |
| <tt> | Tekst teleprintera |
| <address> | Informacije o autoru/vlasniku dokumenta |
| <del> | Izbačeni tekst |
| <ins> | Umetnuti tekst |
| BITMAPE | |
| <img> | Definiše sliku |
| <map> | Mapa klikabilnih područja na slici |
| <area> | Pojedino klikabilno područje na slici |
| LISTE | |
| <ul>, <ol>, <dl> | Liste (ne uređena, uređena i lista definicija) |
| <li> | Pojedina stavka numerisane ili ne numerisane liste |
| <dt> | Termin u listi definicija |
| <dd> | Opis termina u listi definicija |
| TABELE | |
| <table> | Tabela |
| <caption> | Naslov tabele |
| <thead> | Grupiše sadržaj zaglavlja u tabeli |
| <tbody> | Grupiše sadržaj tijela tabele |
| <tfoot> | Grupiše sadržaj podnožja u tabeli |
| <tr> | Red tabele |
| <td> | Ćelija tabele |
| <th> | Ćelija naslova |
| <col> | Definira vrednosti atributa za jednu ili više kolona |
| <colgroup> | Grupiše kolone tabele |
| HYPERLINK | |
| <a> | Hyperlink na drugi dokument ili područje unutar istog |
| FORME | |
| <form> | Forma za unose korisnika |
| <fieldset> | Okvir za elemente forme |
| <legend> | Naslov za uokvirene elemente forme |
| <label> | Oznaka (natpis) za pojedinu kontrolu |
| <input> | Područje za unos informacija od strane korisnika |
| <select> | Padajući meni |
| <option> | Pojedine opcije u padajućem meniju |
| <optgroup> | Grupisanje sličnih opcija u padajućem imeniju |
| <button> | Dugmena koji se može staviti sadržaj (za opis značenja dugmeta) |
| <textarea> | Područje za unos teksta u više linija |
| GENERIČKI TAGOVI | |
| <div> | Divizija u dokumentu |
| <span> | Sekcija u dokumentu |
| UGRAĐENI OBJEKTI (Java applet, ActiveX, PDF, Flash) | |
| <object> | Definiše ugrađeni objekat |
| <param> | Pojedini parametar ili varijabla ugrađenog objekta |

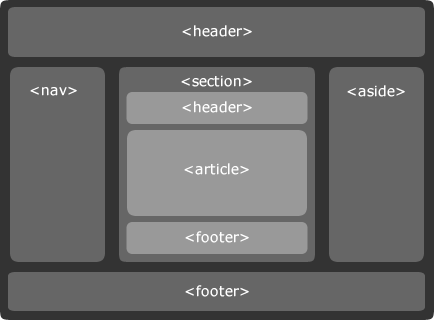
# HTML 5

Aktuelna verzija HTML-a 4.01 bez izmena je opstala punih 10 godina, a u tom periodu kvalitativni razvoj web sadržaja oslanjao se na dve raspoložive mogućnosti:

* maksimalno korištenje komplementarnih tehnologija za implementaciju rešenja koja unutar samog HTML 4.01 nije moguće izvesti: CSS, JavaScript, DOM, itd.
* upotreba vlasničkih dodataka browser-ima za bogate web sadržaje: Flash, Java, Silverlight, Google Gears itd.

Sredinom 2004. godine formirana je grupacija Web Hypertext Application Technology Working Group (WHATWG), kao svojevrstan odgovor na relativno spor razvoj standarda od strane W3C konzorcijuma. Okosnicu grupacije čine uticajne osobe iz kompanija Apple, Mozilla Foundation i Opera Software, uz značajnu potporu šire web zajednice, te naknadno zajedničku saradnju s W3C-om. Nakon nekoliko godina definisanja standarda uz konzultiranje svih zainteresiranih strana, u aprilu 2010., dakle u vreme nastajanja ovog rada, predlog specifikacije standarda HTML5 nalazi se pred konačnom objavom.

HTML5 je doneo značajan napredak s aspekta strukture web dokumenata, definiše nove specijalne tagove kojima se dokument može organizovati jasnije i jednostavnije u odnosu na postojeći potpuno generički <div> tag:



*Slika2. Specijalni HTML5 tagovi za organizaciju dokumenta*

Takođe, neki često korišćeni elementi modernih web dokumenta koji su morali biti implementirani pomoću JavaScripta, su podržani specijalnim tagovima uz potpunu kotrolu izlgleda kroz listu stilova. Radi se o poljima za različite korisničke unose s automatskom validacijom (numeričke/email/datumske/vremenske), te često korištenim kontrolama kao što su progressbar, interaktivni grid, meni, itd.

Stardadiziran je i <canvas> tag, do sada donekle podržan od nekih browser-a, kojim se specificira dvodimenzionalno područje za crtanje upotrebom skriptnog jezika (grafikoni, prezentacije itd).

Tagovi koji su u već u HTML verziji 4.01 bili proglašeni zastarelima, sada su potpuno izbačeni, primeri: <applet>, <font>, <frame>, <s>, <strike>, <u> itd.

Možda najznačajniji korak napravljen je u standardizaciji podrške za bogate web sadržaje koji su do sada bili rezervisani isključivo za različite vlasničke dodatke browser-u (plugins):

* <audio> i <video> tagovi za multimedijalne sadržaje
* podrška za geolokaciju korisnika
* offline storage
* offline application caching

## Novi Elementi i Karakteristike

HTML 5 je definitivno standard koji će prilično brzo promeniti pogled na Internet, doprineti lakšem i bržem razvoju ozbiljnih aplikacija, omogućiti apsolutnim početnicima da jednostavno kreiraju sve aktualne komponente na web stranicama, pa čak i one malo zahtevnije. Kada govorim o HTML 5 standardu, ne govorimo samo o HTML-u, već u o nizu tehnologija implementiranih pod navedenim imenom. Tu su pre svega CSS i JS (JavaScript), ali i sve ono novo i zanimljivo što donosi HTML 5.

## Multimedia

* Audio

Do sada nije postojao standard za preslušavanje audio fajlova na web strani. Danas se većina audio fajlova preslušava preko plugin-ova (Flash), tako da svaki od pretraživača može imati različite. HTML5 kao jednu od novina sadrži element <audio>, kojim je definisan standard za audio fajlove na web stranama.

|  |
| --- |
| <!DOCTYPE html>  <html>  <body>  <audio controls=”controls”>  <source src=”song.ogg” type=”audio/ogg” />  <source src=”song.mp3” type=”audio/mpeg” />  Your browser does not support the audio element.  </audio>  </body>  </html> |

Atribut control dodaje nam audio kontrole kao što su play, pause, kao i mogućnost regulisanja jačine zvuka. Između krajnjih tagova nije na odmet staviti tekst kojim će korisnici biti obavešteni ukoliko njihov pretraživač ne podržava <audio> element. <audio> element omogućava elemente sa različitih izvora, <source> element nam omogućava da naši audio fajlovi budu na različitim lokacijama, a pretraživač će otvoriti prvi format koji podržava. Na slici 1. je prikazana realizacija audio elementa.

* Video

Kao i sa audio elementom, isti je slučaj i sa video elementom, do sada nije postojao standard za postavljanja videa na web stranama. Do sada su vam bili potrebni razni dodaci (plug-ins), Apple QuickTime, RealPlayer ili Adobe Flash, koji su ugradjeni u pretraživače. Podrška za ovaj element je u razvoju, tako da ga ne podržavaju svi pretraživači još uvek, ali se radi na tome. Najveći problem je u usaglašavanju formata. HTML 5 podržava .ogg, .mpeg4 i .webM, dok je ostale formate potrebno konvertovati.

|  |
| --- |
| <!DOCTYPE html>  <html>  <body>  <video width="320" height="240" controls="controls">  <source src="movie.mp4" type="video/mp4" />  <source src="movie.ogg" type="video/ogg" />  Your browser does not support the video tag.  </video>  </body>  </html> |

# CASCADING STYLE SHEETS (CSS)

CSS je meta-jezik kojim se definiše prezentacija dokumenata kreiranih bilo kojim markup jezikom, pa time i HTML/XHTML dokumenata. Dodjeljivanjem stilova (fontova, boja, pozadina, margina, pozicija...) pojedinim elementima ili grupama elemenata definiše se grafička prezentacija web dokumenta. Korišćenjem CSS-a nad strukturiranim web dokumentima moguće je uticati na prezentaciju dokumenata bez potrebe za dodavanjem novih HTML tagova, te bez žrtvovanja nezavisnosti uređaja. Doslednom upotrebom CSS-a, postiže se potpuna separacija prezentacije web dokumenta od njegovog sadržaja.

Kada korisnik učita web stranicu, browser šalje dodatne upite na server za sve povezane i ugrađene objekte (kao što su slike, video, itd.) među kojima su i datoteke sa CSS kodom..

W3C je otišao dosta duboko u specifikaciji načina na koji bi browser-i trebali renderovati CSS-om stilizovani HTML kod, ali kod većine browser-a još uvek postoje greške i nedoslednosti u implementaciji. Rezultat toga je da ne postoje dva browser-a koja će potpuno jednako renderovati web stranicu. U većini slučajeva ove razlike korisnici neće ni primetiti ali u nekim slučajevima su te razlike jako uočljive i zahtevaju dodatnu pozornost pri razvoju kako bi stranica izgledala konzistentno u širokom spektru browser-a.

Kao primer može se navesti prikaz objekta sa sledećim svojstvima:

#objekt1 {

border: 5px solid #000;

padding: 10px;

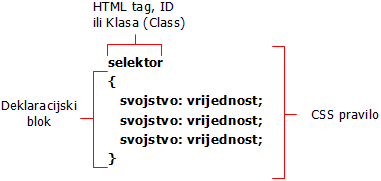
width: 100px;

}

Većina browser-a će ovom objektu dati širinu prostora za sadržaj 100px na što će još dodati debljinu okvira i debljinu padding-a, pa će cijeli objekt (njegovi spoljni rubovi) biti širok 130px. Internet Explorer 5.5 i ranije verzije dati će ukupnu širinu objekta 100px, što znači da će prostor za sadržaj biti 70px. Srećom, u svim novijim verzijama browser-a takve banalne nedoslednosti su ispravljene, ali i dalje su prisutne neke sitne nedoslednosti. Na primer, stil width primenjen nad inline-tagovima većina browser-a ignoriše (kako je i predviđeno standardnom), no čak i novije verzije Internet Explorera nepravilno predstavljajutaj stil kao da se radi o blok-tagovima, to jest dodeljuju inline elemnetu specificiranu fiksnu širinu.

Prvu draft CSS specifikaciju od World Wide Web Consortium-a koncipirali su Håkon Wium Lie i Bert Bos 1996. godine. Dve godine kasnije objavljena je verzija broj 2, a tek 8.9.2009. CSS-2.1., ali paralelno s razvojem CSS-2 radi se i na razvoju CSS-3 standarda.

## Sintaksa

Sintaksa CSS-a sastoji se od samo dva elementa: selektora i deklaracijskog bloka. Selektor označava HTML element ili skup elemenata nad kojima će biti primenjen deklaracijski blok, a deklaracijski blok se sastoji od grupe svojstava i njihovih vrijednosti.

*Slika 3. Osnovna sintaksa CSS-a*

## Boje

Boje možemo definisati na dva načina:

* preko imena,
* kombinacijom slova i brojeva (brojevi imaju širu paletu boja, ali ih je teže zapamtiti).

Heksadecimalni zapis koristi 6 znakova. To je kombinaciju slova i brojeva kako bi se definisala boja. Ako znamo hex kod neke boje, to je sve što nam treba da je postavimo na stranicu. Dok koristimo hex kod kako bi definisali znak, uvek ispred koda treba stajati znak #.

Također možemo koristiti dve decimalne RGB vrednosti:

* Rgb (r, g, b): r, g i b su brojevi između 0 i 255 koji predstavljaju crvenu (red),

zelenu(green) i plavu(blue) boju.

* Rgb(r%, g%, b%): za svaku boju određujemo procenat.

CSS koristi sljedeća svojsva kako bi definirao boju:

* Color (definira boju fonta i boje za linkove)
* Background ili background-color definiše boju pozadine za celu stranicu ili za pojedine elemente

Selector (color: value;)

Najčešća upotreba pseudo klasa u CSS-u je definisanje stilskog pravila za zadani element u "stablu dokumenta". Ovo je tehnički termin koji znači da internet browser gradi hijerarhijsku reprezentaciju svih elemenata u dokumentu.

Primer. Link je pseudo klasa koja definiše stilska pravila za bilo koji još neposećeni link. Postoji 5 najčešćih pseudo klasa koje se mogu koristiti s hyperlinkovima (vezama):

* link(formatiranje izgleda još neposjećenih linkova)
* :visited(formatiranje izgleda posjećenih linkova)
* :focus(formatiranje izgleda likova trenutno selektiranih pomoću tipke)
* :hover (formatiranje izgleda linkova na kojima je trenutačno strelica miša)
* :active(formatiranje izgleda linkova koji su selektovani)

Ispred imena pseudo klase dolazi dvotočka. Mogu se koristiti sa:

* Elementima
* Klasama
* ID-ovima

Link može biti u više stanja od jednom. Stilska pravila za linkove uvek se trebaju napisati

* :link,
* :visited,
* :visible,
* :focus,
* :hover,
* :active.

## Korisni alati za CSS

CSS dokumenti mogu se napisati koristeći jednostavni tekstualni editor poput notepada. No danas postoji puno programa koji služe baš za pisanje CSS-a. Korištenje Notepad-a najbolje će vas naučiti kako pravilno pisati CSS kod. Macromedia Dreamweaver ima jednostavno rešenje za izradu CSS stilova. Stilovi su već definisani od strane programa, a vi samo morate odrediti njihovu vrednost odabirom iz padajućih menija. Umetanje CSS klasa je takođe vrlo jednostavno i izvodi se s dva-tri klika mišem. Nedostatak Dreamweaver-a (kao i svih ostalih vizualnih web editora) jeste taj što CSS kod neće biti tako čitljiv kao da ste ga ručno napisali u Notepadu. Postoji nekoliko alata koji su namenjeni isključivo pisanju CSS-a. Jedan od boljih je TopStyle. TopStyle ima sve napredne opcije koje imaju moderni editori i još puno više. StyleWorx je sasvim zadovoljavajući za početnike, označava sintaksu, ima podršku za CSS1 i CSS2 stilove.

## Vrste selektora

Selektor može biti HTML tag što znači da će taj tag biti redefinisan u CSS-u i gde god se pojavljuje u dokumentu biće prikazan kao što je definisano CSS-om. Npr. sledećim kodom će svaki <h1> tag imati svojstva definisana unutar vitičastih zagrada:

h1 {

font:small-caps bold italic 2.5em Georgia, 'Times New Roman', times, serif;

color: red; }

Korišćenje klase kao selektora omogućava postavljanje nezavisnog stila koji se može primeniti na bilo koji HTML tag. Klasa, za razliku od HTML selektora koji automatski cilja specifični tag, daje jedinstveno ime koje se onda specificira u atributu bilo kojeg HTML taga (ili tagova) na koji se želi primeniti definisani stil. U sljedećem primeru definisana je klasa .nazivPoglavlja:

.nazivPoglavlja {

display: block;

margin-bottom: 8px;

font-size: smaller;

color:black; }

Klasa .nazivPoglavlja će redefinsati svaki tag HTML koda koji u svom atributu ima specificirano njeno ime. Primena ove klase može se videti na primeru <span> tagu:

<span class="nazivPoglavlja">

Sadržaj taga definiranog klasom "nazivPoglavlja"

</span>

ID selektor je jedinstveni identifikator koji se koristi za kreiranje stila nezavisno o tagu. Pomoću njega se uspostavlja struktura stranice. Kada HTML elementu dodelimo ID kreirali smo objekat, na taj način browser pristupa svojstvima tog elementa. Npr. ako div tagu dodelimo ID vrednosti "sadrzaj":

<div id="sadrzaj">

identificirali smo taj div kao jedinstveni element na ekranu kojim se kasnije može manipulisati kroz CSS na sledeći način:

#sadrzaj {

position: absolute;

top: 190px;

width: 480px; }

Naravno, "id" vrednost ne može se ponoviti nigde u celom dokumentu i tu je konceptualna razlika u odnosu na ostale vrste selektora.

Atribut taga takođe može biti korišćen kao selektor. Kod ovog selektora stil se može primeniti na različite načine:

* na sve tagove koji imaju specificirani atribut bez obzira na njegovu vrednost, npr.: [onclick] {...} biti će primenjen na sve tagove koji imaju onclick atribut,
* na sve tagove koji imaju specificirani atribut i egzaktnu vrednost tog atributa, npr.: [href=index.html] {...} bit će primenjen na sve tagove koji imaju navedeni atribut i vrednost
* na tagove koji sadrže specificiranu vrednost kao deo vrednosti atributa odvojenu razmacima od ostatka vrednosti, npr.: [title~=Maja] {...} primeniti će se na sve tagove koji u vrednosti atributa 'title' imaju reč ' Maja' kao deo vrednosti odvojen razmacima od ostatka, npr.: title="Hello from Maja".
* na tagove koji sadrže specificiranu vrednost kao početak vrednosti atributa odvojen crticom od ostatka vrednosti, npr.: [lang|=srb] {...} biti će primenjen na sve tagove kojima vrednost atributa 'lang' počinje sa 'srb-' ili je cela vrijednost samo srb', npr.: lang="srb-srbija". ili lang="srb".

CSS podržava svojevrsnu hijerarhiju stilova pa je dozvoljeno kombinovati različite selektore u koje se onda može uključiti i tzv. univerzalni selektor asterisk '\*', npr. '#sadrzaj \* i' biće primenjen na <i> tag koji je ugniježđen u bilo koji tag kojem je roditelj tag s atributom 'id=sadrzaj'.

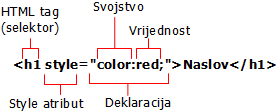
Zbog svega navedenog, model HTML elementa naziva se box-model:



### *Slika 4. Prikaz box modela*

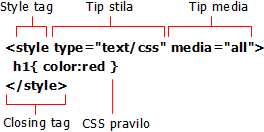
## Načini implementacije stilova

CSS možemo ugraditi u HTML na tri načina:

* U pojedinačni HTML tag unutar HTML koda, tako da utiče samo na taj konkretan tag u dokumentu (inline rule).

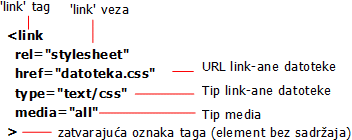
*Slika 5. Sintaksa inline implementacije*

* Unutar 'head' taga HTML-a, da utiče na pojedinačnu web stranicu (internal rule).



*Slika 6.. Sintaksa internal implementacije*

* Kao eksterni dokument tako da utičena celi web site (external rule). Eksterni dokument može biti ugrađen na dva načina:
  + pomoću 'link' taga unutar zaglavlja HTML dokumenta



*Slika 7. Sintaksa external implementacije putem linka*

* + ili importiran preko URL-a eksterne CSS datoteke.

Slika - Sintaksa 'external' implementacije putem importa

*Slika 8. Sintaksa external implementacije putem importa*

Import se može koristiti ne samo kao import u HTML, nego se jedna eksterna CSS datoteka može importovati u drugu.

## Prednosti upotrebe CSS-a u web dokumentima

Iako je CSS moguće koristiti samo parcijalno, mnoge su prednosti njegove pune primene od samog početka razvoja web projekata:

1. Potpuno odvajanje sadržaja od prezentacije. Spoljni CSS može sadržati stilove za sve elemente svih stranica web site-a. Time se postiže dvostruka korist: sam sadržaj ostaje rasterećen od prezentacijskih informacija, dok je s druge strane prezentacija web site-a definisana centralizovano, pa se svaka izmena aplicira na sve stranice web site-a.
2. Fleksibilno pozicioniranje elemenata. Bez upotrebe CSS-a, delove HTML dokumenta moguće je samo donekle pozicionirati upotrebom tablica, pokazalo se kao neprecizno i izrazito štetno s aspekta strukture dokumenta.
3. Ubrzano učitavanje stranica. Renderovanje tablica, naročito ugniježđenih, daleko je sporije od CSS-baziranog pozicioniranja, a osim toga HTML dokumenti su i svojom veličinom bitno manji. Zbog toga se CSS-bazirane stranice učitavaju daleko brže.
4. Učitavanje stranica bez JavaScripta. Deo korisnika interneta ima deaktiviran JavaScript u browser-u uglavnom radi zaštite kompjutera od virusa i/ili nametljivih reklama. To znači da npr. JavaScript navigacija neće biti prikazana, a pomoću CSS-a često je moguće postići isti ili barem sličan efekt.
5. Dostupnost na različitim medijima. CSS pravila za tipove media ('@media' rules) specificiraju kako će dokument biti prezentovan na različitim medijima: ekran, papir, braille-ov uređaj, handheld uređaji, itd. U zavisnosti od toga s kojeg uređaja dolazi zahtev za prikaz web stranice browser koristi odgovarajuću CSS datoteku. Neka CSS svojstva dizajnirana su samo za jedan određeni tip medija, (npr. 'page-break-before' samo za stranične medije), dok neka svojstva mogu deliti različiti mediji ali zahtijevaju i različite vrednosti (npr. 'font-size' svojstvo koristi se za 'print' i 'screan' mediji ali radi njihovih razlika za štampani dokument obično treba manja veličina fonta).
6. Brži razvoj dizajna. CSS omogućava brže dizajniranje stranica jer često korišćene stilove možemo spremiti u odvojenu datoteku i tako učiniti lako dostupnim za korištenje kod svakog novog dizajna. Osim toga, kod novog razvoja dizajna može se koristiti postojeća CSS datoteka i na njoj raditi modifikacije umjesto razvoja nove.

Koliko je CSS efikasan u separaciji dizajna od sadržaja najbolje pokazuje primer sa Zen Garden site-a (profesionalni site za grafičku umetnost CSS-a) gde je identičan web sadržaj, dakle ista HTML datoteka, stilizovana primenom dve različite CSS datoteke:

# ŠTA JE JAVASCRIPT

JavaScript je objektno orjentisan programski jezik namenjen je dinamičkoj izradi Web sajtova. Koristi se u milionima Web sajtova za poboljšanje funkcionalnosti , validaciju formi ,detekciju browsera, asihronu komunikaciju sa serverom i još mnogo toga. Razvijen je 1995. godine od strane Netscape-a i korsiti se kao client-side (klijent zasnovan) internet programski jezik. To znači da se njegova obrada vrši na korisnikovom računaru u samom browser-u čime se postiže veća brzina rada, web aplikacije odgovaraju brže na korsnikove akcije i samim tim se postiže veća efikasnost. Predstavlja skriptni jezik , što u prevodu znači lagan programski jezik.

1. JavaScript je dizajniran da bi se dodala interaktivnost HTML stranama

2. JavaScript je skript jezik (jednostavan programski jezik)

3. JavaScript se sastaji od linija koda koji može da se izvršava

4. JavaScript se obično direktno ugrađuje u HTML strane

5. JavaScript je interpreterski jezik (što znači da se izvršava bez prethodnog kompajliranja)

6. JavaScript nije licenciran jezik

## JAVASCRIPT UNUTAR WEB BROWSERA

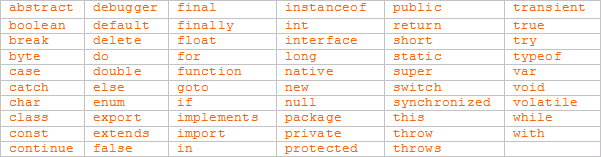
Kada je popularnost Netscape Navigatora počela naglo da raste, Nombas je razvio verziju CEnvi-ja prilagođenog izvršavanju unutar browser-a, drugim rečima unutar web-stranica. Ti rani eksperimenti su nazvani 'Espresso Pages', i predstavljaju prvi client-side skriptni programski jezik korišćen na World Wide Webu. U to vreme malo ko je mogao predvideti da će ideja potekla iz male kompanije Nombas imati toliki uticaj na kasniji razvoj interneta, i da će postati jedna od njegovih temeljnih komponenti.Kako je korišćenje interneta dobijalo na popularnosti, postepeno je rasla i potreba za standardnim client-side skriptnim jezikom, u najvećoj meri zbog jednostavne ali krucijalne potrebe za validacijom web-formi. Bez skriptnog jezika unutar web browser-a, validnost svake formekoju bi korsnik popunio morala bi se proveriti na serverskoj strani. To znači da bi korisnik nakon što je popunio formu i kliknuo na 'submit' button, bilo potrebno sledeće:

* slanje podataka iz forme na server
* provera validnosti unetih podataka
* u slučaju neispravnosti nekog od unetih podataka, ponovno učitavanje stranice sa obaveštenjem koja polja u formi nisu korektno popunjena
* ponovno slanje forme na server, itd.

Ako se uzme u obzir činjenica da je u to vreme najveći broj korisnika pristupao internetu preko modema tipičnom brzinom 28.8 kbps, i da je i serverski potencijal tada bio znatno manji nego danas, za samo jedan round-trip prilikom slanja forme bilo je potrebno i 30-ak sekundi, i to za običnu validaciju web-forme.

## KLJUČNE REČI I TIPOVI PODATAKA

JavaScript se sastoji od sledećih ključnih reči:



*Slika 9.Operatori izraza*

Neke od njih nisu u upotrebi, već su rezervisane za buduća proširenja. Bilo koja od ključnih reči se ne može koristiti za imenovanje varijabli ili funkcija.

JavasScript poznaje sledeće tipove podataka:

* numeričke (npr. 35 ili 12.341)
* logičke (jedna od vrednosti true ili false)
* stringove (npr. 'Dobar dan' ili 'poruka za korisnika') i
* specijalni tip podataka, a radi se o samo dve predefinisane vrednosti: null za null-vrednost, i undefined za vrednost varijable koja nije definisana

Ovako mali broj primitivnih tipova podataka pokazao se sasvim dostatanim u razvoju web aplikacija, s obzirom da za kompleksnije tipove podataka postoji mogućnost kreiranja struktura sačinjenih od primitivnih tipova i/ili drugih struktura.

## VARIJABLE I LITERALI

Varijable su, kao i u drugim programskim jezicima, simbolički nazivi vrednosti u aplikaciji, koji su podložni jednostavnim pravilima: naziv mora započeti alfa-karakterom (A-Z ili a-z), underscore-om (\_) ili dolar ($) karakterom, svi preostali karakteri u nazivu mogu dodatno sadržavati i numeričke karaktere. JavaScript je case-sensitive, pa tako mogu postojati dve različite varijable s nazivima 'a123' i 'A123'.

* Primeri sintaksno korektnih naziva varijabli: a, x1, $1\_abc, Razmak\_u\_imenu, \_Clan
* Primeri pogrešnih naziva varijabli: 1a, Razmak u imenu, +abc

JavaScript je jezik koji tipove podataka interpretira dinamički, što znači da se tip varijable ne specificira prilikom deklaracije, te da se podaci automatski konvertiraju prema potrebi tokom izvršavanja skripte. Varijable je potrebno dekilarisati pomoću ključne riječi var, npr:

|  |
| --- |
| var abc; |

Moguće ih je inicijalizirati istovremeno s deklaracijom:

|  |
| --- |
| var abc = 3; |

Takođe, varijable je dopušteno redeklarisati, tako da je sledeći kod dobar:

|  |
| --- |
| var abc = 3;  abc = abc + 1;  var abc = 'Irena'; |

Područje delovanja (scope) svake varijable može biti lokalno ili globalno. Lokalne varijable su one deklarisane unutar funkcije, i raspoložive su samo kodu unutar te funkcije, dok su globalne deklarisane izvan funkcije i dostupne su kodu u celoj skripti. Varijable se mogu deklarisati i bez ključne riječi var i u tom slučaju su globalne bez obzira na mesto deklaracije, ali taj je način izrazito nepoželjan sa staništa strukture koda, te u tom slučaju interpreter generiše 'warning' poruku.Konstante su podržane u JavaScriptu, i zapravo su samo specijalizacija varijabli od kojih se razlikuju u tačno tri osobine: deklarišu se ključnom rečju const (umjesto var), potrebno im je inicijalizovati vrednost prilikom deklaracije, i nemaju mogućnosti promene vrednosti putem pridruživanja druge vrednosti ili redeklaracije. Naravno, unutar istog scope-a ne mogu postojati varijabla i konstanta s jednakim nazivima.

|  |
| --- |
| function f()  {  var a;  a = 3;  const b = 4;  const a = 5; // sintaksno ispravno, ali logička greška; varijabla tog naziva postoji  b = b + 1; // greška, konstanta ne može menjati vrednost  a = a + 1; // greška, iako na početku varijabla, kasnije redeklarisano u const  } |

Osim varijabli primitivnih tipova podataka, JavaScript podržava objekte, koji su u generalnom smislu slični strukturama programskog jezika C, ali daleko fleksibilniji. U osnovi, služe za kreiranje kompleksnih tipova podataka.

Numerički celobrojni literali se mogu izraziti u decimalnom sastavu (bez vodećih nula) ili heksadecimalnom sastavu(s prefiksom '0x'). JavaScript je do verzije 1.5 podržavao i oktalni sistem(uz jednu vodeću nulu), ali je izbačen iz standarda pa ga treba izbegavati, iako ga neki browser-i još uvijek podržavaju. Predznak '-' označava negativnu vrednost, a predznak '+', koji može biti izostavljen, pozitivnu.

Numerički decimalni literali sastoje se od:

* celobrojnog dela (celobrojni numerik) sa ili bez predznaka; može se izostaviti, i u tom slučaju se podrazumeva 0
* decimalne tačke
* decimalnog dela (drugi celobrojni numerik)
* eksponenta (formata 'e' ili 'E' iza kojeg sliedi celobrojni numerik); eksponent se može izostaviti, i u tom slučaju se podrazumieva 1.
* Logički literali mogu imati samo dvie vrednosti: true ili false. Potrebno je obratiti pažnju na činjenicu da je JavaScript case-sensitive, tako da npr. True nije korektno izražena logička vrijednost.

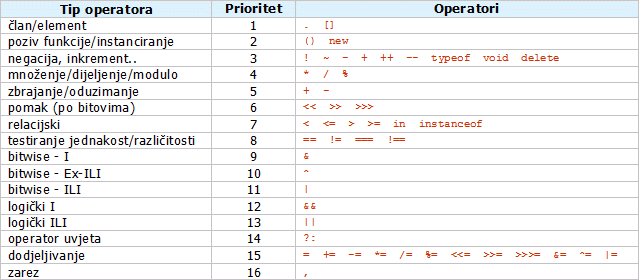
String literali su proizvoljno dug skup karaktera, sa jednostrukim (') ili dvostrukim (") navodnicima. Unutar stringa se, slično kao i u programskom jeziku C, mogu navesti specijalni karakteri kombinacijom backslash (\) karaktera i oznake za konkretni specijalni karakter: \f = from feed, \t = tab, \r = carriage return, \n = new line, \\ = backslash, \' = apostrof, \" = navodnici itd.

## OPERATORI I IZRAZI

Osnovni operatori u JavaScriptu slični su većini jezika, a većina ima potpuno isto značenje kao u programskom jeziku C. Tako je '=' operator pridruživanja, '+' operator sabiranja, '==' operator provere jednakosti itd. Izrazi su bilo koji skup literala, varijabli, operatora i podizraza koji ima jedinstvenom vrednost. JavaScript podržava tri osnovna tipa izraza, analogno primitivnim tipovima podataka: aritmetičke, logičke i stringovne. Konceptualno, postoje dve kategorije izraza:

* izrazi koji pridodeljuju vrednost varijabli; npr. a = 7 jest izraz kojim se varijabli a pridodjeljuje vrijednost 7; on sam (celi izraz) takođe ima vrednost 7. Takvi izrazi koriste operator pridruživanja (assignment operator)
* izrazi koji jednostavno sadrže rezultat kao vrednost; npr. 3 + 5 jest izraz koji jednostavno ima vrednost 8

Pri evaluiranju izraza, vrede više-manje uobičajena pravila važnosti (prednosti) operatora, tako da primer izraz 3 + 2 \* 6 rezultira s 15, a ne 30. Potpuna tablica prednosti opreatora u JavaScriptu:



## MOGUĆNOST I UPOTREBA JAVASCRIPT-A

JavaScript omogućava stvaranje dinamike prilikom izrade neke Web stranice, takođe pruža i veliki broj mogućnosti dizajnerima Web stranica neke od njih su sledeće:

**1. JavaScript daje HTML dizajnerima alat za programiranje -** dizajneri HTML stranica obično nisu programeri,ali JavaScript je skript jezik sa veoma jednostavnom sintaksom.

**2. JavaScript može da dinamički ubaci kod u HTML stranu –** sledeća JavaScript naredba:

document.write("<h1>" + name + "</h1>") može da napiše promenljiv tekst koji senalazi u promenljivoj name na HTML strani.

**3. JavaScript može da reaguje na događaje -** JavaScript može da se podesi tako da se izvrši kad se nešto desi, npr. kad se strana učita ili kad korisnik klikne na HTML element

**4. JavaScript može da pročita ili ispiše HTML elemente -** JavaScript može da pročita i da promenisadržaj HTML elementa.

**5. JavaScript može da se koristi za proveru ispravnosti unetih podataka -** JavaScript može da se koristi za proveru ispravnosti podataka unetih u formu, da proveri ispravnost podataka pre nego što se pošalju serveru.

**6. JavaScript može da se koristi za detektovanje *browser***-**a korisnika** - JavaScript može da se koristiza detektovanje *browser*-a iu zavisnosti od *browser*-a, učitavanje strane specijalno dizajnirane za taj *browser.*

**7. JavaScript može da se koristi za kreiranje *cookie*-ja** - JavaScript može da se koristi za čuvanje ivraćanje informacija o računaru posetioca.

## IMPLEMENTACIJA JAVASCRIPTA

JavaScript u sekciji body će biti izvršen dok se strana učitava. JavaScript u sekciji head će se izvršiti tek kad bude pozvan.Takođe JavaScript se može pozvati i u inline varijanti tj . to je slučaj kada određen poziv JavaScript funkcije ili kompletan JavaScript kod stavljamo na onload ,onclick.

## PRIMERI

Primer 1. Skript koji sadrži funkcije smešta se u *Head* sekciju dokumenta. Ovako smo sigurni da se script učitava pre poziva funkcije

|  |
| --- |
| <html>  <head>  <script type="text/javascript">  function message()  {  alert("This alert box was called with the onload event")  }  </script>  </head>  <body onload="message()">  </body>  </html> |

|  |
| --- |
| <html>  <head>  </head>  <body>  <script type="text/javascript">  document.write("This message is written when the page loads")  </script>  </body>  </html> |

|  |
| --- |
|  |

JavaScript se izvršava odmah kad *browser* učitava stranu. Ovo nije baš uvek poželjno, zato što nekad želimo da izvršimo neki skript tek kad korisnik izazove događaj. **Script u sekciji *head:*** Skriptovi koji treba da se izvrše kad su pozvani, odnosno kad se izazove neki događaj idu u sekciju *head*. Na ovaj način sigurni smo da je skript učitan pre nego što ga bilo ko pozove.

|  |
| --- |
| <html>  <head>  <script type="text/javascript">  ....  </script>  </head>  </html> |

Script u sekciji *body*: Skriptovi koji treba da se izvrše u trenutku učitavanja strane idu u sekciju *body*.

Ovakav skript generiše sadržaj strane.

|  |
| --- |
| <html>  <head>  </head>  <body>  <script type="text/javascript">  ....  </script>  </body>  </html> |

Script u obe sekcije, *body* i *head*: U dokumentu može da postiji neograničeni broj skriptova, tako da možete da ih stavite u obe sekcije.

|  |
| --- |
| <html>  <head>  <script type="text/javascript">  ....  </script>  </head>  <body>  <script type="text/javascript">  ....  </script>  </body>  </html> |

## KORIŠĆENJE SPOLJAŠNJEG JAVASRIPTA

Nekada želite da isti skript koristite na više stranica, a da ne morate da ga ponovo pišete na svakoj strani možete da napišete JavaScript u posebnom fajlu (sa .js ekstenzijom). **Napomena:** Eksterni skript ne može da sadrži tag <script>.

Da bi koristili eksterni skript, stavite .js fajl za vrednost atributa src taga <script>:

|  |
| --- |
| <html>  <head>  <script src="xxx.js"></script>  </head>  <body>  </body>  </html> |

# jQuery

jQuery je moćna JavaScript biblioteka koja vam omogućava da Web sajtu dodate funkcionalnosti koje sa čistim html/css-om ne biste mogli da uradite. Između ostalog, možete pristupati raznim delovima Web stranice kao i samom sadržaju strane i menjati ih, dodavati razne animacije na Web stranicu, reagovati na neki impuls od strane korisnika kao što je na primer klik ili prelazak mišem na određenom delu strane, vaditi informacije sa servera bez potrebe za ponovnim učitavanjem cele strane (ovo je poznato kao AJAX) itd.

Ono što jQuery izdvaja od drugih biblioteka su njegove veoma proste za korišćenje ali pri tom veoma moćne metode. jQuery metode su podeljene u više namenskih grupa i sveukupno čine njegov API.

## jQuery Jezgro

jQuery-jevo jezgro predstavlja njegov glavni jQuery objekat o kojem smo već pričali. On služi sa hvatanje elemenata iz DOM-a pomoću selektora. Može da se koristi na dva načina:

|  |
| --- |
| 1. // preko jQuery imena 2. jQuery();   3.  4. // ili preko spominjanog dolar znaka  5. $(); |

Svejedno je koji ćete pristup koristiti ali praksa pokazuje da je standardno da se koristi dolar znak zbog lakšeg raspoznavanja u kodu. Međutim ponekad je potrebno da na stranici imate još neku bibilioteku osim jQuery-ja. Pošto većina drugih biblioteka takodje koristi dolar znak kao svoj glavni objekat ovo bi moglo da predstavlja problem jer bi došlo do konflikta izmedju dve biblioteke.Tu na scenu stupa jQuery-jeva .noConflict() metoda, koja omogućava nesmetano korišćenje jQuery-ja sa bilo kojim drugim bibliotekama koje koriste dolar objekat. Pošto iz gore priloženog vidimo da jQuery može da se poziva na dva načina, ova metoda isključuje njegovo korišćenje dolar znaka. Drugim rečima, nakon upotrebe ove funkcije jQuery ćete pozivati preko njegovog imena dok će dolar biti rezervisan za druge biblioteke. Sve što je potrebno da uradite je da nadovežete .noConflict() meotodu na dolar pre bilo kakvog korišćenja od strane drugih biblioteka.

|  |
| --- |
| 1. $.noConflict(); 2. // posle ovoga možete da koristite dolar za druge biblioteke |

## jQuery Selektori

Browser-i svaku HTML stranicu učitavaju u memoriju kao skup medjusobno povezanih objekata sa odredjenim svojstvima. To se naziva DOM (Document Object Model). Svaka operacija u JavaScript-u počinje odabirom željenih elemenata sa kojima se želi raditi. Dakle, pristup objektu je osnova bilo kakve operacije nad njim.

Kao što smo spomenuli, jQuery koristi svoj dolar objekat za prosleđivanje HTML elemenata, na kojima onda izvršava svoje funkcije. To se vrši pomoću selektora. Podržavanjem svih CSS 1-3 selektora, kao i dodavanjem svojih, jQuery nudi veoma moćan skup alata za "hvatanje" elemenata u dokumentu. Pođimo od osnovnih elemenata. Sve što je potrebno da uradite da bi ste ugrabili određeni element (ili više njih), je da upišete njegovo ime između navodnika unutar dolar $() funkcije onako kako bi ste to uradili u CSS dokumentu:

|  |
| --- |
|  |

Dakle bilo koji CSS selektor da upotrebite, jQuery će pronaći i selektovati traženi element. Pored standardnih CSS selektora, jQuery je dodao i svoje selektore radi još lakšeg rada sa elementima:

* :checked

Selektuje sve zadate elemente koji su čekirani (takvi elementi mogu biti: checkbox i radio button).

|  |
| --- |
| 1 $("input[type="checkbox"]:checked"); |

Navedeni primer će selektovati svaki prvi <input> element koji je checkbox, koji je pri tom čekiran.

* :contains()

Selektuje sve zadate elemente koji sadrže traženi tekst.

|  |
| --- |
| 1. $("div:contains('John')"); |

Navedeni primer će selektovati svaki <div> element koji u sebi ili bilo gde u elementima koje sadrži tekst John.

* :empty

Selektuje sve zadate elemente koji nemaju nikakve elemente u sebi, uključujući i tekst.

|  |
| --- |
| 1. $("div:empty"); |

Navedeni primer će selektovati sve <div> elemente koji u sebi ne sadrže elemente ili tekst.

* :has()

Selektuje element koji u sebi sadrži makar jedan zadati element.

|  |
| --- |
| 1. $("div:has(p)"); |

Navedeni primer će selektovati sve <div> elemente koji imaju <p> element u sebi ili u bilo kojim elementima koje sadrže, ne samo kao child element.

* :not()

Selektuje elemente koji ne odgovaraju zadatom selektoru.

|  |
| --- |
| 1. $("div:not(.mojaklasa)"); |

Navedeni primer će selektovati sve <div> elemente koji nemaju klasu .mojaklasa

* :first-child

Selektuje elemente koji su prvi child njihovog parent-a.

|  |
| --- |
| 1. $("div span:first-child"); |

Navedeni primer će selektovati svaki prvi <span> element unutar svakog <div> elementa.

# Flash

Flash CS4 je aplikacija koju koristimo za razvoj multimedijalnog sadržaja koja se sastoji od niza alata koji nam omogućuju:

* rad sa vektorskom i rasterskom (bitmap) grafikom
* kreiranje textualnih polja
* animaciju i interaktivnost
* jednostavne i složene navigacijske šeme
* upotrebu zvuka i videa
* povezivanje i razmenu podataka sa drugim aplikacijama i skriptama
* upotreba gotovih komponenti u našim projektima

Flash kao proizvod nije samo orijentiran na grafičku stranu, on također koristi jedan jako moćan programski jezik koji se zove “ActionScript” . ActionScript je objektno orijentirani programski jezik koji omogućuje stvaranje kompleksnih Flash animacija, upravljanje i stvaranje dinamičkog sadržaja, dinamičku kontrolu zvuka i videa te još mnogo toga.

Flash sadržaji ili datoteke nalaze se u 2 formata a to su radna:

* (.fla)
* i završna(.swf) verzija.

(.fla) datoteke su radne datoteke koje možemo iznova i iznova uređivati dok (.swf) su završne ili “zapakovane” datoteke koje su gotove i njih se stavlja na web stranice. Kada smo gotovi sa projektom sve što trebamo je publicirati (publishing) da bi se projekt mogao pregledavati izvan Flash radne okoline. Za pregledavanje (.swf) sadržaja potreban je Flash Player ili Web browser s kojim ćemo pokretati željeni sadržaj.

Prednosti Flash tehnologije su velike, nabrojati ću samo neke od njih:

* Visoko stilizirani web projekti
* Podrška za XML i web servise
* Animacija, zvuk, video
* Nezavisnost o internetskim pretraživačima
* Brz razvoj multimedijalnih projekata
* Jednostavno povezivanje sa serverskim tehnologijama (PHP, ASP.NET…)
* Jednostavno povezivanje s bazama podataka upotrebom poslužiteljskih web tehnologija, poput Coldfusiona, PHP-a, ASP.NET-a, Pearl-a, JSP-a, Flash Remoting tehnologija i slično
* Tržišna zastupljenost Flash Playera
* Razvoj projekata na MS Windows i Apple MacOS sastavima

Glavni meni kao i većina Windows aplikacija pa tako i Flash ima glavni i meni u kojem se nalaze sve naredbe za upravljanje i stvaranje sadržaja u Flash radnoj okolini. Pa tako imamo karticu “File” koji nam daje opcije da kreiramo novi document “New”, da otvorimo postojeći dokument te da sačuvamo dokumente, da “uvozimo” i “izvozimo” simbole u biblioteku (library) i sl.

“Edit” ima uobičajene naredbe kao što su “Copy”, “Paste”, “Cut”, ovde se nalaze i specijalne naredbe kao što je “Paste In Place” te kopiranje i premeštanje sličica animacije (Cut/Copy/Paste Frames).

“View“ nam daje opcije kakav pogled želimo na sceni, da li želimo “Grid” tj. mreže koje su vidljive nama samo radi preciznijeg crtanja, ili “Ruler” tj. lenjir za precizno smeštanje elemenata na sceni i sl.

“Insert” nam pruža opciju dodavanja layera, kadrova, scena, time line efekata, stvaranja simbola.

“Text” najobičnija opcija za obradu texta.

“Commands” nam pruža snimanje i automatizaciju često korištenih naredbi.

“Control” nam daje opcije za kontrolu animacija pomoću osnovnih naredbi “Stop”, “Play”, “Rewind”, “Forward”, te testiranje animacije pre objave.

“Debug” omogućava Flash programerima ispravljanje grešaka u delovima filma pisanim ActionScript programskim jezikom.

“Window” otvaranje pojedinih panela i biblioteka simbola koji dolaze zajedno sa Flash aplikacijom.

“Help” sastav za pomoć u radu sa aplikacijama.

## Scena

Scena je mesto gde sve što radimo stavljamo na nju, ona ja na početku veliko belo platno u sredini same aplikacije, nemoguće je promašiti. Na njoj se izvodi sve ono što želimo da bude prikazano, dok možemo postavljati stvari i iza platna ili sa strane no te stvari neće biti vidljive osim ako se ne napravi animacija da sa te “nevidljive” strane dođu u vidljivi deo platna.

Također postoje “Scene” koje služe ako postoji animacija koja će biti presložena ili više složenijih aplikacija jedna scena nam neće biti dovoljna zato uključujemo dodatnu panelu koja se zove “Scene”, a uljučuje se tako da odemo u “Window -> Other panels -> Scene” ili kraticom “Shift + F2”.Panela “Scene” izgleda ovako:

## Vremenska Linija

Vremenska linija (timeline), vrlo je važan dio Flash radne okoline. To je mesto gde animiramo sadržaj animacije, mesto za kontrolu i pregled animiranog sadržaja te mesto rada sa slojevima (layers). Vremenska linija, zapravo, omogućava stvaranje animacije menjajući sadržaj i izgled scene u nekom vremenskom period. Glavni delovi vremenske linije su slojevi i kadrovi te klizač (playhead). Slojevi drže sadržaj animacije segmentalnim. Desno od slojeva nalaze se kadrovi koji pripadaju tom sloju. Kadrovi u Flash animaciji predstavljaju vreme i događaje, dok klizač pokazuje koji se kadar trenutačno prikazuje na sceni (određeni deo animacije u vremenu). Evo kako izgleda “Vremenska linija”:

|  |
| --- |
|  |

*Slika 10. Vremenska linija.*

## Paneli

Na panelima su grupisane kontrole za upravljanje različitim opcijama Flasha (boje, scene, transformacije, pozicioniranje i sl.).

* “Info” panel nam pokazuje širinu, visinu, koordinate, RGB boju objekta i trenutačnu poziciju miša

|  |
| --- |
|  |

*Slika 11. Paneli.*

* “Transform” precizno menja veličine, rotiranja objekata (skew).

|  |
| --- |
|  |

*Slika 12. kartica Transform.*

* “Align” poravnava objekte jedan prema drugome (u odnosu na scenu).

|  |
| --- |
|  |

*Slika 13. kartica Align.*

* “Color” rad s bojama.

|  |
| --- |
|  |

*Slika 14. kartica Color.*

“Swatches” služi za definisanje prozirnih boja i dodavanje boja u paletu.

* “Properties” je jako bitan panel zato što od svakog objekta, animacije, slike, i bilo čega što je označeno bi će karakteristike toga koje se mogu uređivati.

|  |
| --- |
|  |

*Slika 15. kartica Properties.*

## 

## Alatke u Flash-u

Koje služe za rastersko i vektorsku grafiku su:

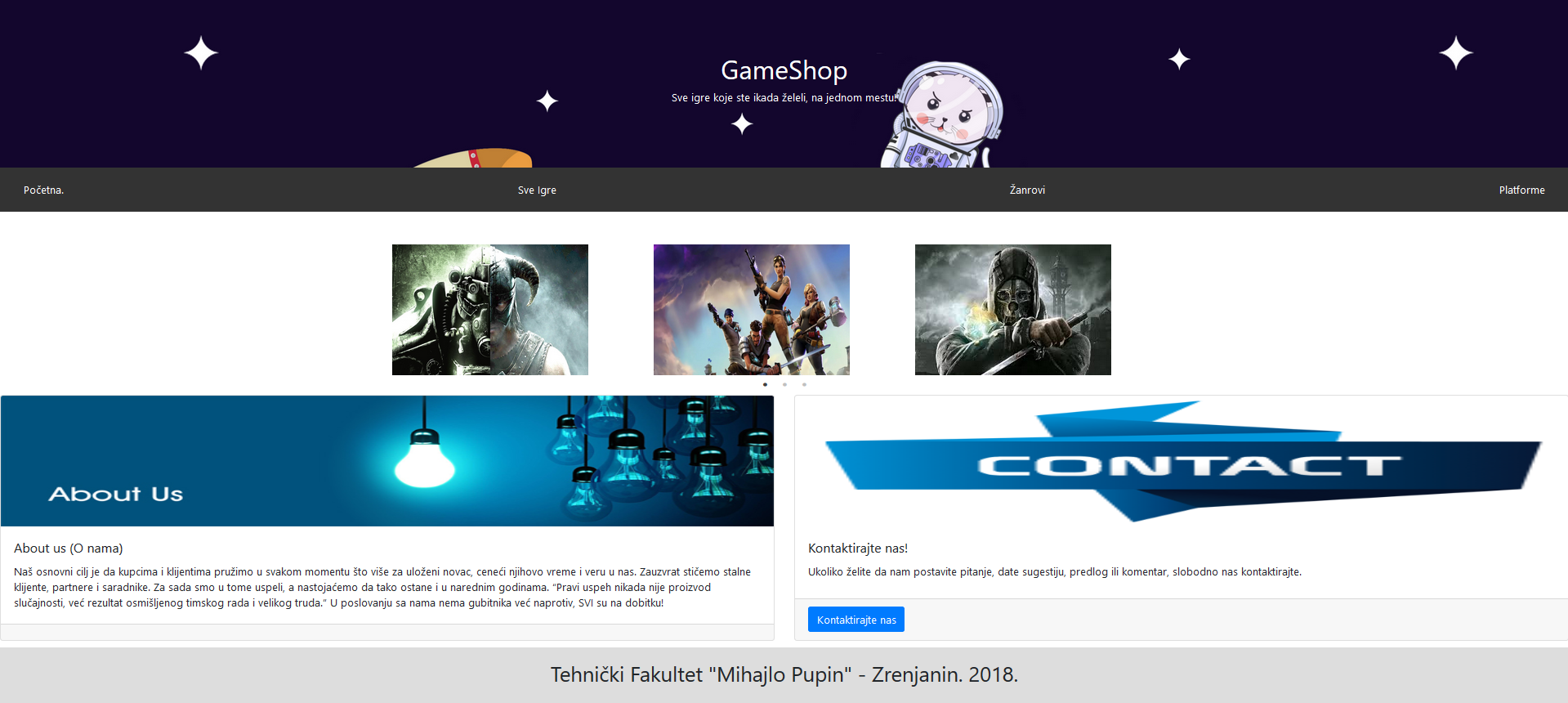
|  |
| --- |
| “SELECTION” označavanje/modificikovanje objekta na radnoj površini. |
| “SUBSELECTION” služi za menjanje označenih tačaka |
| “FREE TRANSFORM” transformisanje objekta (veličina/rotiranje) |
| “GRADIENT TOOL” transformacija ispona |
| “LASSO” označavanje nepravilnih/pravilnih oblika |
| “LINE” crtanje pravih linija |
| “PEN” crtanje krivih |
| “TEXT” polja za unos teksta |
| “SHAPE TOOLS” alati za crtanje oblika poput elipsi, kvadrata, poligona i sl. |
| “PENCIL”olovka za crtanje linija |
| “BRUSH” četkica crtanje ispona |
| “DECO TOOL” alat crtanje vektorskih uzoraka |
| “BONE TOOL” alat za inverznu kinematiku odnosno izradu koštane armature za animaciju, dostupan isključivo u radu s AS3.0 datotekama. |
| “INK BUCKET” menjanje/dodavanje boja i vrste obruba (stroke) |
| “PAINT BUCKET” menjanje/dodavanje boja i vrsta uspona (fill) |
| “EYEDROPPER”prenos boje/vrste postojeće linije/ispona s jednog na drugi efekt. |
| “ERASER” briše linije i/ili ispone |
| “HAND” služi za pomicanje po uvećanoj sceni |
| “ZOOM” služi za uvećanje/umanjenje prikaza scene |
| “SELECTION” označavanje/modifikovanja objekta na radnoj površini. |

# Kreiranje Web Sajta

Za izradu Web sajta koristio sam sledeće tehnologije: Adobe Photoshop CC 2018, HTML, Javascript, CSS, jQuery, Bootstrap, Slick.

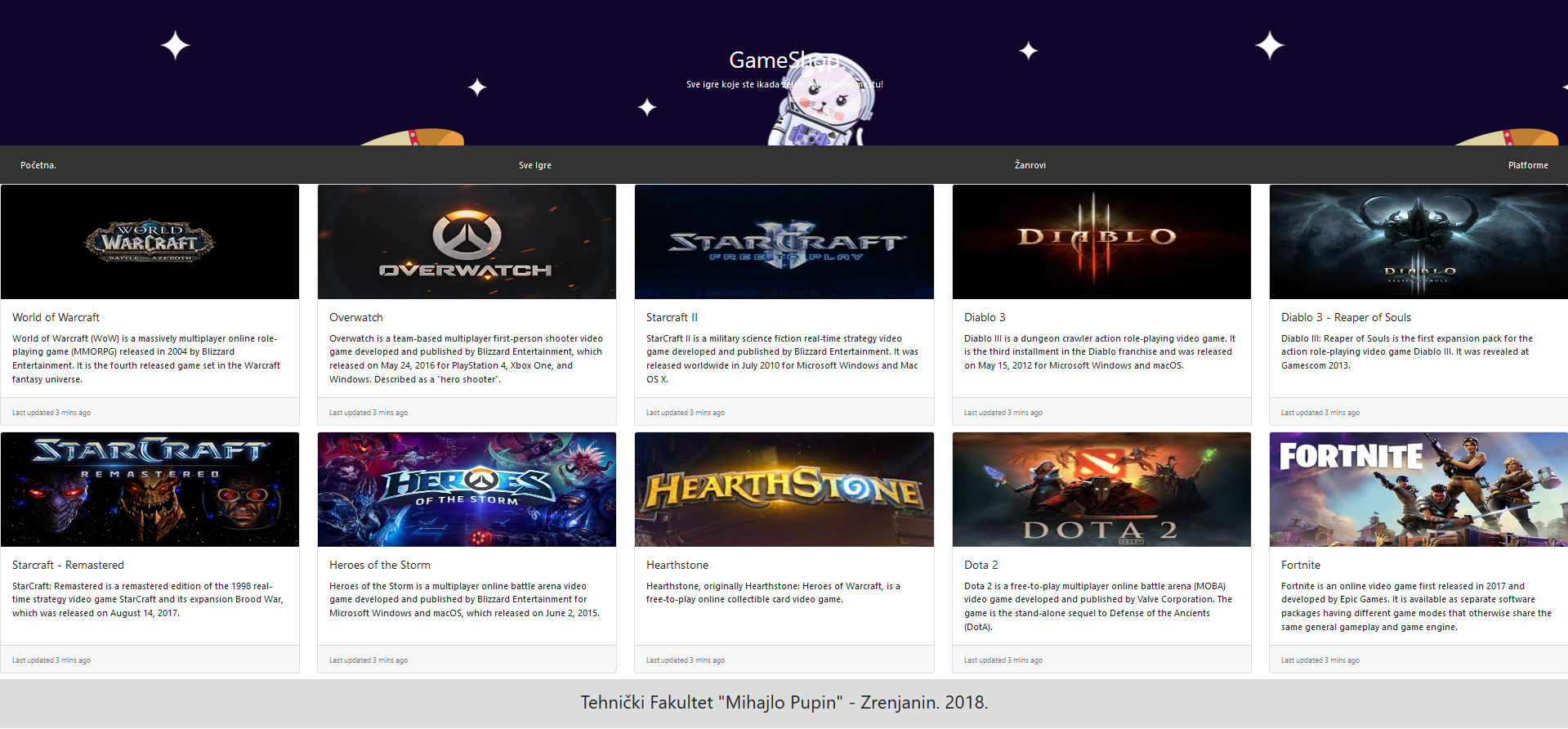
## Kreiranje početne strane

Prilikom kreiranja početne stranice, trudio sam se da ispoštujem neka osnovna pravila (strukture) početne stranice. Dodao sam navigaciju, “header”, informacije o kompaniji I kontakt, slajder sa slikama nekih trenutno popularnih igara I footer u kom su ostavljene informacije o fakultetu.



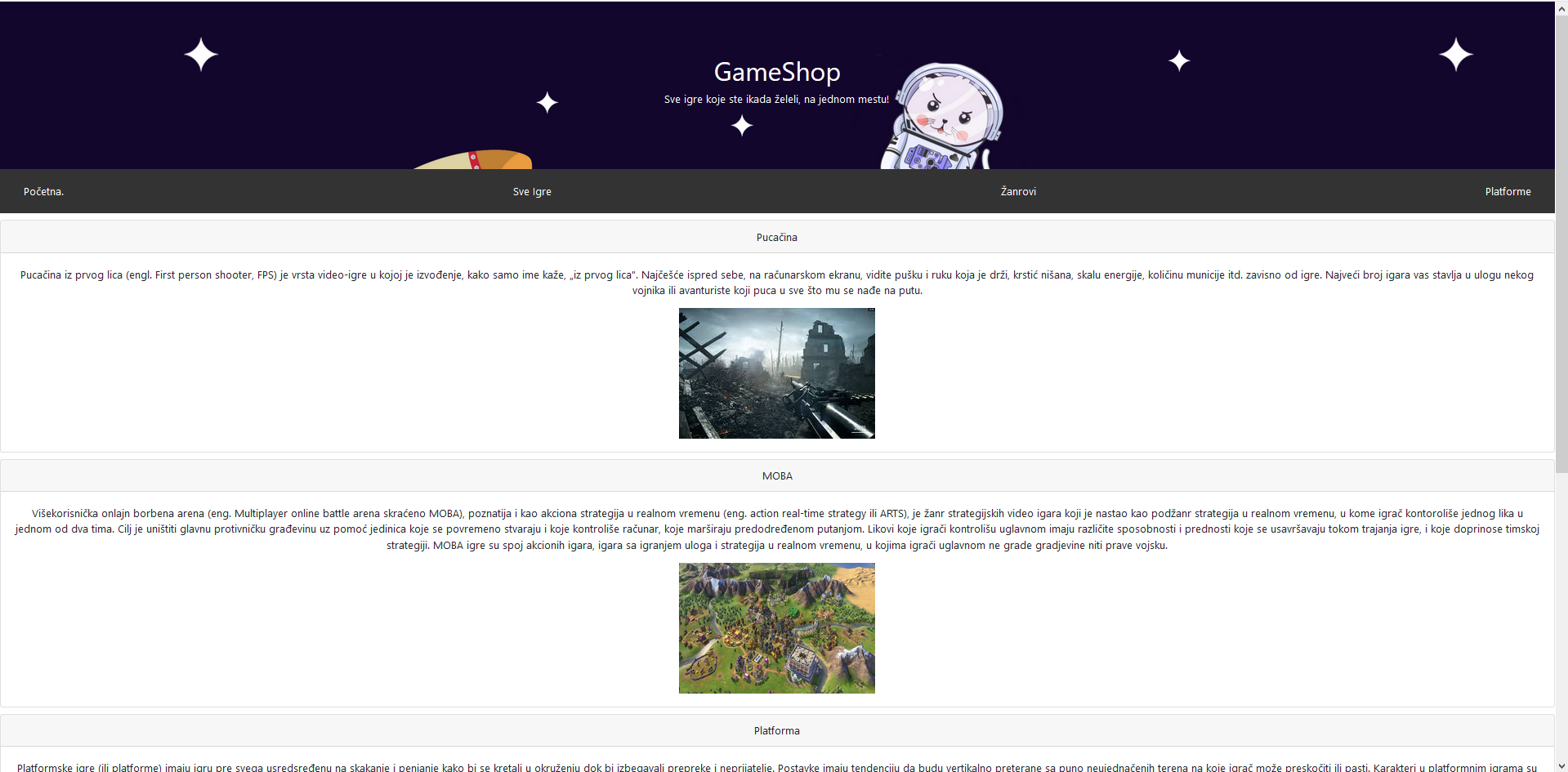
## Kreiranje stranice “Sve Igre”

Stranica “Sve Igre” je slična početnoj stranici, samo što se na ovoj stranici nalaze dodatne informacije o svim igrama. Dodate su takozvane “Kartice” a svaka kartica sadrži informacije o pojedinačnoj igri.



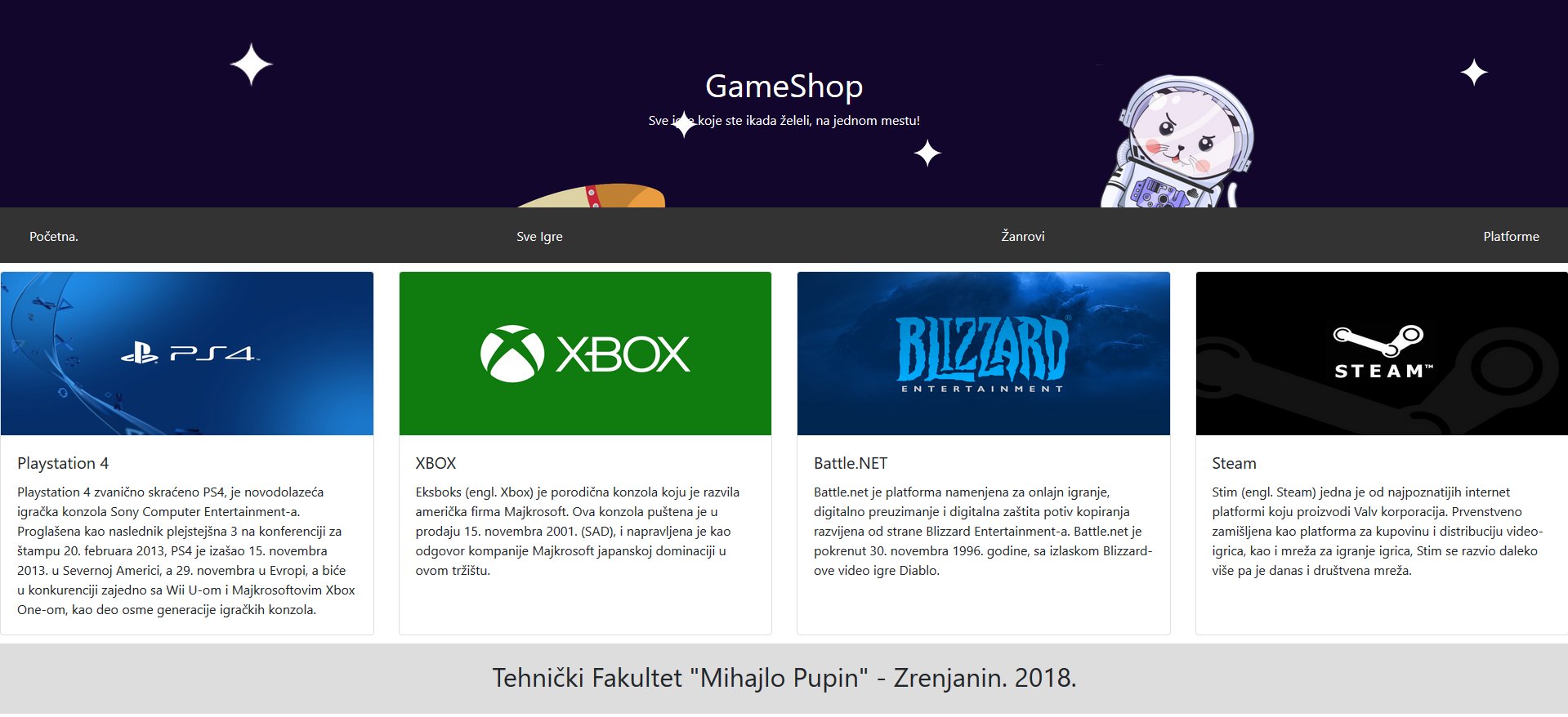
## Kreiranje stranice “Žanrovi”

Stranica “Žanrovi” sadrži informacije o nekoliko žanrova. Sada ne koristimo kartice, već žanrove ređamo jedan iznad drugog.



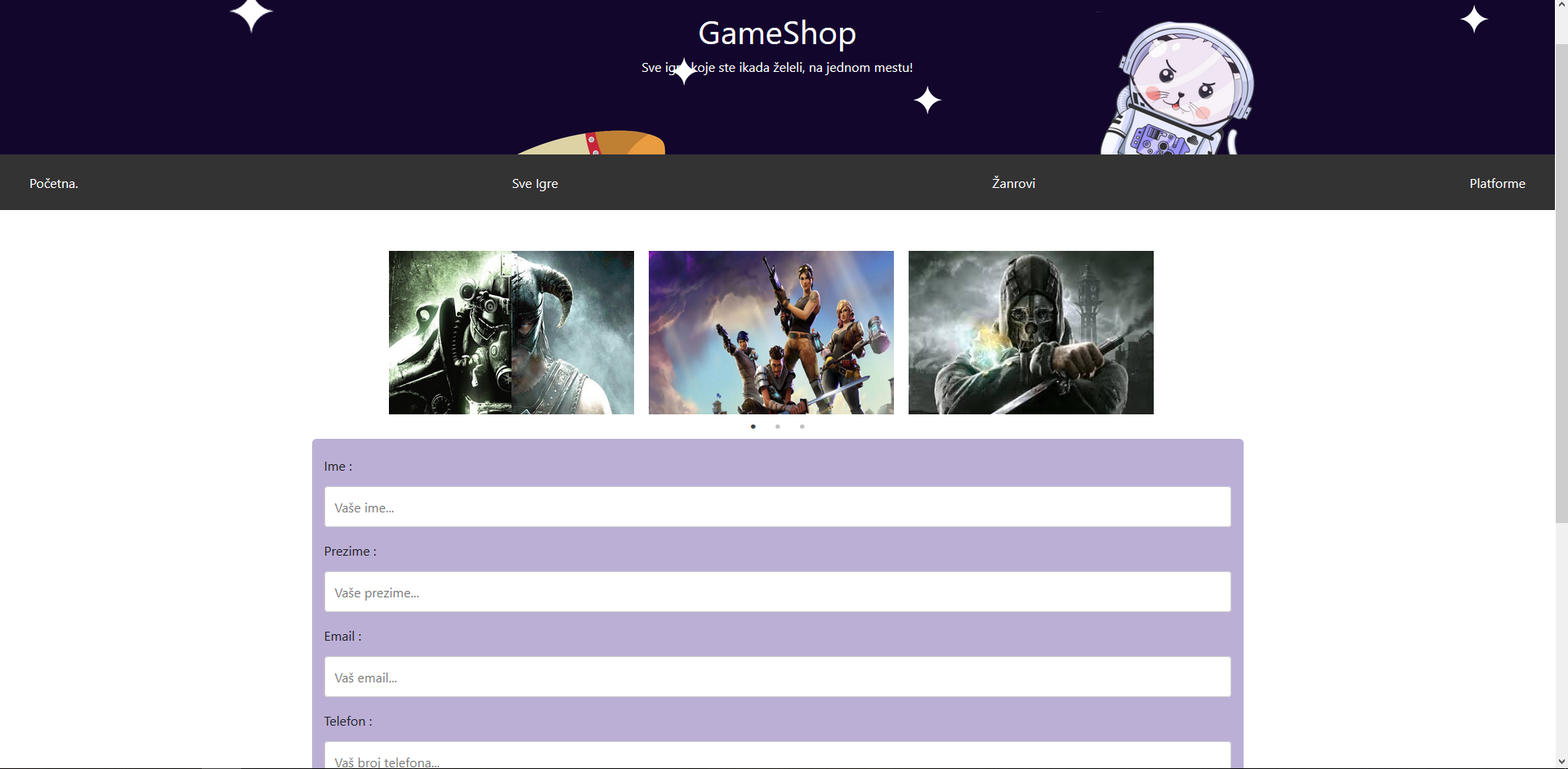
## Kreiranje stranice “Platforme”

Stranica “Platforme” je slična stranici “Sve Igre”, ali sadrži informacije o trenutno najpopularnijim platformama za trgovinu I igranje igara.



## Kreiranje stranice “Kontakt”

Stranica “Kontakt” sadrži formu preko koje korisnici mogu da nas kontaktiraju.



# Zaključak

Sa novijim dobima dobijamo sve više opcija za stvaranje web sajtova I aplikacija. Ovaj seminarski rad je samo jedan primer od mnogih kako se može napraviti statički sajt korišćenjem HTML, CSS, Javascript I nekoliko biblioteka.

# Literatura

[1] Krusha, Edmond; Barbir, Ines; Kardum, Mladen; Haraminčić, Dalibor; Kralj, Damir: " [Tim Berners-Lee 'Surfing the network'](http://www.procesne-mreze.com/racunala/tim-berners-lee-surfing-the-network-2.html)"  
[2] Berners-Lee &Tim: "[Information Management: A Proposal](http://www.w3.org/History/1989/proposal.html)"  
[3] Jacobs & Ian: "[About the World Wide Web Consortium (W3C)](http://www.w3.org/Consortium/Overview.html)"  
[4] W3Schools: "[HTML Tutorial](http://www.w3schools.com/htmL/html_elements.asp)"  
[5] Zakas &Nicholas C.: "[History and evolution of Javascript](http://www.java-samples.com/showtutorial.php?tutorialid=814)"  
[6] W3C: "[Web Content Accessibility Guidelines 1.0](http://www.w3.org/TR/WCAG10)"  
[7] Flanagan, David: "[JavaScript: The Definitive Guide, 4th Edition](http://docstore.mik.ua/orelly/webprog/jscript/ch12_01.htm)"  
[8] W3C: "[Document Object Model (DOM) Requirements](http://www.w3.org/TR/2000/WD-DOM-Requirements-20000208/)"  
[9] Huang &Vivian: "[The DOM Event Model Basic](http://www.lingihuang.com/viviancrap/archives/the-dom-event-model-basic/)"  
[10] OpenAjax Alliance: "[The characteristics of Ajax applications](http://www.openajax.org/member/wiki/images/8/89/NexawebAjaxCharacteristics.pdf)"   
[11]Michael Morrison(2007). Head First JavaScript.  
[12]Ryan Benedetti, Ronan Cranley(2011). Head First jQuery.  
[13]Lesa Snider. (2012). Photoshop CS. O'Reilly Media / Pogue Press.  
[14]Adobe Creative Team. (2012). Adobe Photoshop CS6 Classroom in a Book. Adobe  
[15] Huang, Vivian: "The DOM Event Model Basic"  
[16]OpenAjax Alliance: "Introducing Ajax and OpenAjax   
[17] Wei, Coach K.: "AJAX: Asynchronous Java + XML?"  
[18] W3Schools: "[CSS Tutorial](http://www.w3schools.com/htmL/html_elements.asp)"

# Prilog

Index.html:

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <title>GameShop</title>  <link rel="stylesheet" href="Resources/css/index.css" />  <script src="https://code.jquery.com/jquery-3.3.1.js" integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  crossorigin="anonymous"> </script>  <link rel="stylesheet" type="text/css" href="Resources/slick-1.8.1/slick/slick.css" />  <link rel="stylesheet" type="text/css" href="Resources/slick-1.8.1/slick/slick-theme.css" />  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  crossorigin="anonymous"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"  crossorigin="anonymous"></script>  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"  crossorigin="anonymous"></script>  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"  crossorigin="anonymous">  </head>  <body>  <header role="banner" class="header">  <h1>GameShop</h1>  <p>Sve igre koje ste ikada želeli, na jednom mestu!</p>  </header>  <div class="navbar">  <a href="index.html">Početna.</a>  <a href="Pages/sve\_igre.html">Sve Igre</a>  <a href="Pages/zanrovi.html">Žanrovi</a>  <a href="Pages/platforme.html">Platforme</a>  </div>  <div class="slider">  <div>  <img src="Media/Images/g1.jpg" />  </div>  <div>  <img src="Media/Images/g2.jpg" />  </div>  <div>  <img src="Media/Images/g3.jpg" />  </div>  <div>  <img src="Media/Images/g4.jpg" />  </div>  <div>  <img src="Media/Images/g5.jpg" />  </div>  <div>  <img src="Media/Images/g6.jpg" />  </div>  <div>  <img src="Media/Images/g7.jpg" />  </div>  <div>  <img src="Media/Images/g8.jpg" />  </div>  <div>  <img src="Media/Images/g9.jpg" />  </div>  </div>  <div class="card-deck">  <div class="card">  <img class="card-img-top" src="Media/Images/7.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">About us (O nama)</h5>  <p class="card-text">Naš osnovni cilj je da kupcima i klijentima pružimo u svakom momentu što više za  uloženi novac, ceneći njihovo vreme i veru u nas. Zauzvrat stičemo stalne klijente, partnere i  saradnike. Za sada smo u tome uspeli, a nastojaćemo da tako ostane i u narednim godinama.  “Pravi uspeh nikada nije proizvod slučajnosti, već rezultat osmišljenog timskog rada i velikog  truda.”  U poslovanju sa nama nema gubitnika već naprotiv, SVI su na dobitku!</p>  </div>  <div class="card-footer">  </div>  </div>  <div class="card">  <img class="card-img-top" src="Media/Images/6.png" style="background-size: contain;" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Kontaktirajte nas!</h5>  <p class="card-text">Ukoliko želite da nam postavite pitanje, date sugestiju, predlog ili komentar,  slobodno nas kontaktirajte.</p>  </div>  <div class="card-footer">  <a href="Pages/kontakt.html" class="btn btn-primary">Kontaktirajte nas</a>  </div>  </div>  </div>  <div class="footer">  <h2>Tehnički Fakultet "Mihajlo Pupin" - Zrenjanin. 2018.</h2>  </div>  <script type="text/javascript" src="Resources/slick-1.8.1/slick/slick.min.js"></script>  <script type="text/javascript">  $(document).ready(function () {  $('.slider').slick({  slidesToShow: 3,  slidesToScroll: 3,  dots: true,  infinite: true,  cssEase: 'linear'  });  });  </script>  </body>  </html> |

Zanrovi.html

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <title>GameShop</title>  <link rel="stylesheet" href="../Resources/css/index.css" />  <script src="https://code.jquery.com/jquery-3.3.1.js" integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  crossorigin="anonymous"> </script>  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick.css" />  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick-theme.css" />  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  crossorigin="anonymous"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"  crossorigin="anonymous"></script>  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"  crossorigin="anonymous"></script>  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"  crossorigin="anonymous">  </head>  <body>  <header role="banner" class="header">  <h1>GameShop</h1>  <p>Sve igre koje ste ikada želeli, na jednom mestu!</p>  </header>  <div class="navbar">  <a href="../index.html">Početna.</a>  <a href="../Pages/sve\_igre.html">Sve Igre</a>  <a href="../Pages/zanrovi.html">Žanrovi</a>  <a href="../Pages/platforme.html">Platforme</a>  </div>  <div class="card text-center cardMarginTop10">  <div class="card-header">  Pucačina  </div>  <div class="card-body">  <p class="card-text">Pucačina iz prvog lica (engl. First person shooter, FPS) je vrsta video-igre u kojoj  je izvođenje, kako samo ime kaže, „iz prvog lica". Najčešće ispred sebe, na računarskom ekranu, vidite  pušku i ruku koja je drži, krstić nišana, skalu energije, količinu municije itd. zavisno od igre.  Najveći broj igara vas stavlja u ulogu nekog vojnika ili avanturiste koji puca u sve što mu se nađe na  putu.</p>  <img src="../Media/Images/zanr1.jpg" />  </div>  </div>  <div class="card text-center cardMarginTop10">  <div class="card-header">  MOBA  </div>  <div class="card-body">  <p class="card-text">Višekorisnička onlajn borbena arena (eng. Multiplayer online battle arena skraćeno  MOBA), poznatija i kao akciona strategija u realnom vremenu (eng. action real-time strategy ili ARTS),  je žanr strategijskih video igara koji je nastao kao podžanr strategija u realnom vremenu, u kome igrač  kontoroliše jednog lika u jednom od dva tima. Cilj je uništiti glavnu protivničku građevinu uz pomoć  jedinica koje se povremeno stvaraju i koje kontroliše računar, koje marširaju predodređenom putanjom.  Likovi koje igrači kontrolišu uglavnom imaju različite sposobnosti i prednosti koje se usavršavaju  tokom trajanja igre, i koje doprinose timskoj strategiji. MOBA igre su spoj akcionih igara, igara sa  igranjem uloga i strategija u realnom vremenu, u kojima igrači uglavnom ne grade gradjevine niti prave  vojsku. </p>  <img src="../Media/Images/zanr2.jpg" />  </div>  </div>  <div class="card text-center cardMarginTop10">  <div class="card-header">  Platforma  </div>  <div class="card-body">  <p class="card-text">Platformske igre (ili platforme) imaju igru pre svega usredsređenu na skakanje i  penjanje kako bi se kretali u okruženju dok bi izbegavali prepreke i neprijatelje. Postavke imaju  tendenciju da budu vertikalno preterane sa puno neujednačenih terena na koje igrač može preskočiti ili  pasti. Karakteri u platformnim igrama su obično u stanju da skoče mnogo puta svoje sopstvene visine, a  igračima je ponuđena neka kontrola nad njihovim kretanjem u vazduhu, kao i visinu i rastojanje njihovih  skokova</p>  <img src="../Media/Images/zanr3.png" />  </div>  </div>  <div class="card text-center cardMarginTop10">  <div class="card-header">  Borilačka  </div>  <div class="card-body">  <p class="card-text">Borba je obično jedan na jedan ili protiv malog broja jednako moćnih protivnika,  često uključujući nasilne i preterane nenadmašive napade. Većina borbenih igara karakteriše veliki  broj likova za reprodukciju i konkurentski multiplayer mod.</p>  <img src="../Media/Images/zanr4.jpg" />  </div>  </div>  <div class="footer">  <h2>Tehnički Fakultet "Mihajlo Pupin" - Zrenjanin. 2018.</h2>  </div>  </body>  </html> |

Kontakt.html

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <title>GameShop</title>  <link rel="stylesheet" href="../Resources/css/index.css" />  <link rel="stylesheet" href="../Resources/css/kontakt.css" />  <script src="https://code.jquery.com/jquery-3.3.1.js" integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  crossorigin="anonymous"> </script>  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick.css" />  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick-theme.css" />  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  crossorigin="anonymous"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"  crossorigin="anonymous"></script>  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"  crossorigin="anonymous"></script>  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"  crossorigin="anonymous">  </head>  <body>  <header role="banner" class="header">  <h1>GameShop</h1>  <p>Sve igre koje ste ikada želeli, na jednom mestu!</p>  </header>  <div class="navbar">  <a href="../index.html">Početna.</a>  <a href="../Pages/sve\_igre.html">Sve Igre</a>  <a href="../Pages/zanrovi.html">Žanrovi</a>  <a href="../Pages/platforme.html">Platforme</a>  </div>  <div class="slider">  <div>  <img src="../Media/Images/g1.jpg" />  </div>  <div>  <img src="../Media/Images/g2.jpg" />  </div>  <div>  <img src="../Media/Images/g3.jpg" />  </div>  <div>  <img src="../Media/Images/g4.jpg" />  </div>  <div>  <img src="../Media/Images/g5.jpg" />  </div>  <div>  <img src="../Media/Images/g6.jpg" />  </div>  <div>  <img src="../Media/Images/g7.jpg" />  </div>  <div>  <img src="../Media/Images/g8.jpg" />  </div>  <div>  <img src="../Media/Images/g9.jpg" />  </div>  </div>  <div class="container">  <form action="action\_page.php">  <label for="fname">Ime : </label>  <input type="text" id="fname" name="firstname" placeholder="Vaše ime...">  <label for="lname">Prezime : </label>  <input type="text" id="lname" name="lastname" placeholder="Vaše prezime...">  <label for="lname">Email : </label>  <input type="text" id="email" name="email" placeholder="Vaš email...">  <label for="lname">Telefon : </label>  <input type="text" id="telefon" name="telefon" placeholder="Vaš broj telefona...">  <label for="subject">Poruka</label>  <textarea id="subject" name="subject" placeholder="Napišite nešto..." style="height:200px"></textarea>  <input type="submit" value="Pošalji">  </form>  </div>  <div class="footer">  <h2>Tehnički Fakultet "Mihajlo Pupin" - Zrenjanin. 2018.</h2>  </div>  <script type="text/javascript" src="../Resources/slick-1.8.1/slick/slick.min.js"></script>  <script type="text/javascript">  $(document).ready(function () {  $('.slider').slick({  slidesToShow: 3,  slidesToScroll: 3,  dots: true,  infinite: true,  cssEase: 'linear'  });  });  </script>  </body>  </html> |

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <meta http-equiv="X-UA-Compatible" content="ie=edge">  <title>GameShop</title>  <link rel="stylesheet" href="../Resources/css/index.css" />  <link rel="stylesheet" href="../Resources/css/kontakt.css" />  <script src="https://code.jquery.com/jquery-3.3.1.js" integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  crossorigin="anonymous"> </script>  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick.css" />  <link rel="stylesheet" type="text/css" href="../Resources/slick-1.8.1/slick/slick-theme.css" />  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"  crossorigin="anonymous"></script>  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js" integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"  crossorigin="anonymous"></script>  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js" integrity="sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"  crossorigin="anonymous"></script>  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"  crossorigin="anonymous">  </head>  <body>  <header role="banner" class="header">  <h1>GameShop</h1>  <p>Sve igre koje ste ikada želeli, na jednom mestu!</p>  </header>  <div class="navbar">  <a href="../index.html">Početna.</a>  <a href="../Pages/sve\_igre.html">Sve Igre</a>  <a href="../Pages/zanrovi.html">Žanrovi</a>  <a href="../Pages/platforme.html">Platforme</a>  </div>  <div class="card-deck">  <div class="card">  <img class="card-img-top" src="../Media/Images/igra1.png" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">World of Warcraft</h5>  <p class="card-text">World of Warcraft (WoW) is a massively multiplayer online role-playing game  (MMORPG) released in 2004 by Blizzard Entertainment. It is the fourth released game set in the  Warcraft fantasy universe.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra2.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Overwatch</h5>  <p class="card-text">Overwatch is a team-based multiplayer first-person shooter video game developed  and published by Blizzard Entertainment, which released on May 24, 2016 for PlayStation 4, Xbox  One, and Windows. Described as a "hero shooter".</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra3.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Starcraft II</h5>  <p class="card-text">StarCraft II is a military science fiction real-time strategy  video game developed and published by Blizzard Entertainment. It was released worldwide in July  2010 for Microsoft Windows and Mac OS X.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra5.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Diablo 3</h5>  <p class="card-text">Diablo III is a dungeon crawler action role-playing video game. It is the third  installment in the Diablo franchise and was  released on May 15, 2012 for Microsoft Windows and macOS.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra4.jpeg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Diablo 3 - Reaper of Souls</h5>  <p class="card-text">Diablo III: Reaper of Souls is the first expansion pack for the action  role-playing video game Diablo III. It was revealed at Gamescom 2013.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  </div>  <div class="card-deck" style="margin-top:10px;">  <div class="card">  <img class="card-img-top" src="../Media/Images/igra6.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Starcraft - Remastered</h5>  <p class="card-text">StarCraft: Remastered is a remastered edition of the 1998 real-time strategy video  game StarCraft and its expansion Brood War, which was released on August 14, 2017.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra7.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Heroes of the Storm</h5>  <p class="card-text">Heroes of the Storm is a multiplayer online battle arena video game developed and  published by Blizzard Entertainment for Microsoft Windows and macOS, which released on June 2,  2015.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra8.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Hearthstone</h5>  <p class="card-text">Hearthstone, originally Hearthstone: Heroes of Warcraft, is a free-to-play online  collectible card video game.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra9.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Dota 2</h5>  <p class="card-text">Dota 2 is a free-to-play multiplayer online battle arena (MOBA) video game  developed and published by Valve Corporation. The game is the stand-alone sequel to Defense of the  Ancients (DotA).</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  <div class="card">  <img class="card-img-top" src="../Media/Images/igra10.jpg" alt="Card image cap">  <div class="card-body">  <h5 class="card-title">Fortnite</h5>  <p class="card-text">Fortnite is an online video game first released in 2017 and developed by Epic  Games. It is available as separate software packages having different game modes that otherwise  share the same general gameplay and game engine.</p>  </div>  <div class="card-footer">  <small class="text-muted">Last updated 3 mins ago</small>  </div>  </div>  </div>  <div class="footer">  <h2>Tehnički Fakultet "Mihajlo Pupin" - Zrenjanin. 2018.</h2>  </div>  <script type="text/javascript" src="../Resources/slick-1.8.1/slick/slick.min.js"></script>  <script type="text/javascript">  $(document).ready(function () {  $('.slider').slick({  slidesToShow: 3,  slidesToScroll: 3,  dots: true,  infinite: true,  cssEase: 'linear'  });  });  </script>  </body>  </html> |

Index.css

|  |
| --- |
| body {  font-family: Arial, Helvetica, sans-serif;  }  .header {  padding: 80px;  text-align: center;  background-image: url("../../Media/Images/1.png");  color: white;  }  .header h1 {  font-size: 40px;  }  .navbar {  overflow: hidden;  background-color: #333;  }  .navbar a {  float: left;  display: block;  color: white;  text-align: center;  padding: 14px 20px;  text-decoration: none;  }  .navbar a.right {  float: right;  }  .navbar a:hover {  background-color: #ddd;  color: black;  }  .row {  display: flex;  flex-wrap: wrap;  }  .side {  flex: 30%;  background-color: #f1f1f1;  padding: 20px;  }  .main {  flex: 70%;  background-color: white;  padding: 20px;  }  @media screen and (max-width: 700px) {  .row {  flex-direction: column;  }  }  @media screen and (max-width: 400px) {  .navbar a {  float: none;  width: 100%;  }  }  .footer {  padding: 20px;  text-align: center;  background: #ddd;  }  .slider {  width: 50%;  margin: 0 auto;  margin-top: 50px;  }  div img {  width: 300px;  height: 200px;  }  .footer {  margin-top: 10px;  }  .cardMarginTop10 {  margin-top: 10px;  } |